

Solutions Companion

C++

**AND ALGORITHMIC THINKING
FOR THE COMPLETE BEGINNER**

3rd Revised Edition

Aristides S. Bouras

Copyright ©

Copyright © by Aristides S. Bouras
<https://www.bouraspage.com>

RCode: 240301

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, mechanical or electronic, including photocopying, recording, or by any information storage and retrieval system, without written permission from the authors.

Warning and Disclaimer

This book is designed to provide the answers to all of the review questions, as well as the solutions to all review exercises of the book "C++ AND ALGORITHMIC THINKING FOR THE COMPLETE BEGINNER – Third Edition". Every effort has been taken to make this book compatible with all releases of C++, and it is almost certain to be compatible with any future releases of it.

The information is provided on an "as is" basis. The authors shall have neither liability nor responsibility to any person or entity with respect to any loss or damages arising from the information contained in this book or from the use of the files that may accompany it.

Table of Contents

How to Report Errata.....	8
If you Like the Book.....	8
Chapter 1.....	9
1.7 Review Questions: True/False.....	9
1.8 Review Questions: Multiple Choice.....	9
Review in “Introductory Knowledge”	10
Review Crossword Puzzles	10
Chapter 4.....	12
4.17 Review Questions: True/False	12
4.18 Review Questions: Multiple Choice.....	12
Chapter 5.....	13
5.8 Review Questions: True/False.....	13
5.9 Review Questions: Multiple Choice.....	13
5.10 Review Exercises	13
Chapter 6.....	14
6.4 Review Questions: True/False.....	14
6.5 Review Questions: Multiple Choice.....	14
Chapter 7.....	15
7.7 Review Questions: True/False.....	15
7.8 Review Questions: Multiple Choice.....	15
7.9 Review Exercises	15
Chapter 8.....	16
8.2 Review Questions: True/False.....	16
8.3 Review Exercises	16
Chapter 9.....	18
9.3 Review Exercises	18
Review in “Getting Started with C#”	20
Review Crossword Puzzles	20
Chapter 10	21
10.2 Review Exercises	21
Chapter 11	29
11.3 Review Questions: True/False	29
11.4 Review Questions: Multiple Choice.....	29

11.5 Review Exercises	29
Chapter 12	32
12.2 Review Exercises	32
Chapter 13	35
13.2 Review Exercises	35
Chapter 14	38
14.4 Review Questions: True/False	38
14.5 Review Questions: Multiple Choice	38
14.6 Review Exercises	39
Review in “Sequence Control Structures”	42
Review Crossword Puzzle	42
Chapter 15	43
15.9 Review Questions: True/False	43
15.10 Review Questions: Multiple Choice	43
15.11 Review Exercises	43
Chapter 16	45
16.2 Review Questions: True/False	45
16.3 Review Questions: Multiple Choice	45
16.4 Review Exercises	45
Chapter 17	52
17.2 Review Questions: True/False	52
17.3 Review Questions: Multiple Choice	52
17.4 Review Exercises	52
Chapter 18	61
18.2 Review Questions: True/False	61
18.3 Review Exercises	61
Chapter 19	73
19.2 Review Questions: True/False	73
19.3 Review Exercises	73
Chapter 20	81
20.2 Review Questions: True/False	81
20.3 Review Exercises	81
Chapter 21	90
21.4 Review Exercises	90
Chapter 22	97

22.9 Review Questions: True/False	97
22.10 Review Questions: Multiple Choice	97
22.11 Review Exercises.....	98
Chapter 23	102
23.7 Review Exercises	102
Review in “Decision Control Structures”	125
Review Crossword Puzzle	125
Chapter 24	126
24.3 Review Questions: True/False	126
Chapter 25	127
25.4 Review Questions: True/False	127
25.5 Review Questions: Multiple Choice.....	127
25.6 Review Exercises	127
Chapter 26.....	140
26.3 Review Questions: True/False	140
26.4 Review Questions: Multiple Choice.....	140
26.5 Review Exercises	140
Chapter 27	157
27.3 Review Questions: True/False	157
27.4 Review Questions: Multiple Choice.....	157
27.5 Review Exercises	157
Chapter 28	166
28.4 Review Exercises	166
Chapter 29	175
29.8 Review Questions: True/False	175
29.9 Review Questions: Multiple Choice.....	175
29.10 Review Exercises.....	175
Chapter 30	181
30.7 Review Questions: True/False	181
30.8 Review Exercises	181
Review in “Loop Control Structures”	210
Review Crossword Puzzle	210
Chapter 31	211
31.13 Review Questions: True/False	211
31.14 Review Questions: Multiple Choice	211

31.15 Review Exercises.....	211
Chapter 32.....	227
32.7 Review Questions: True/False.....	227
32.8 Review Questions: Multiple Choice.....	227
32.9 Review Exercises.....	228
Chapter 33.....	241
33.8 Review Questions: True/False.....	241
33.9 Review Questions: Multiple Choice.....	241
33.10 Review Exercises.....	241
Chapter 34.....	252
34.7 Review Questions: True/False.....	252
34.8 Review Exercises.....	252
Review in “Data Structures in C#”.....	292
Review Crossword Puzzle.....	292
Chapter 35.....	293
35.4 Review Questions: True/False.....	293
Chapter 36.....	294
36.8 Review Questions: True/False.....	294
36.9 Review Exercises.....	294
Chapter 37.....	308
37.9 Review Questions: True/False.....	308
37.10 Review Exercises.....	308
Chapter 38.....	319
38.3 Review Exercises.....	319
Review in “Subprograms”.....	333
Review Crossword Puzzle.....	333
Chapter 39.....	334
39.8 Review Questions: True/False.....	334
39.9 Review Exercises.....	334
Review in “Object Oriented Programming”.....	351
Review Crossword Puzzle.....	351
Chapter 40.....	352
40.8 Review Questions: True/False.....	352
40.9 Review Exercises.....	352
Chapter 41.....	357

41.2 Review Exercises	357
Review in "Files"	370
Review Crossword Puzzle	370
Some Final Words from the Author	371
Some of my Books	372

How to Report Errata

Although I have taken great care to ensure the accuracy of the content in this book, mistakes can still occur. If you come across any errors, either in the text or the code, I highly encourage you to report them to me. By doing so, you'll not only assist in saving other readers from potential confusion and frustration but also contribute to enhancing the quality of the next release. If you discover any errors, please report them by visiting one of the following addresses:

- <https://tinyurl.com/28nwh2nf>
- <https://www.bouraspape.com/report-errata>



Once I verify your reported error(s), your submission will be accepted. The errata will then be uploaded to my website and added to any existing list of corrections.

If you Like the Book

If you find the book valuable, please consider visiting the web store where you purchased it, as well as goodreads.com, to show your appreciation by writing a positive review and awarding as many stars as you think appropriate. By doing so, you will motivate me to keep writing and, of course, you'll be assisting other readers in discovering my work.

Chapter 1

1.7 Review Questions: True/False

1. true
2. false
3. true
4. false
5. false
6. true
7. true
8. false
9. false
10. false
11. true
12. true
13. false
14. false
15. false
16. true
17. true
18. false
19. false
20. true
21. false
22. false
23. true

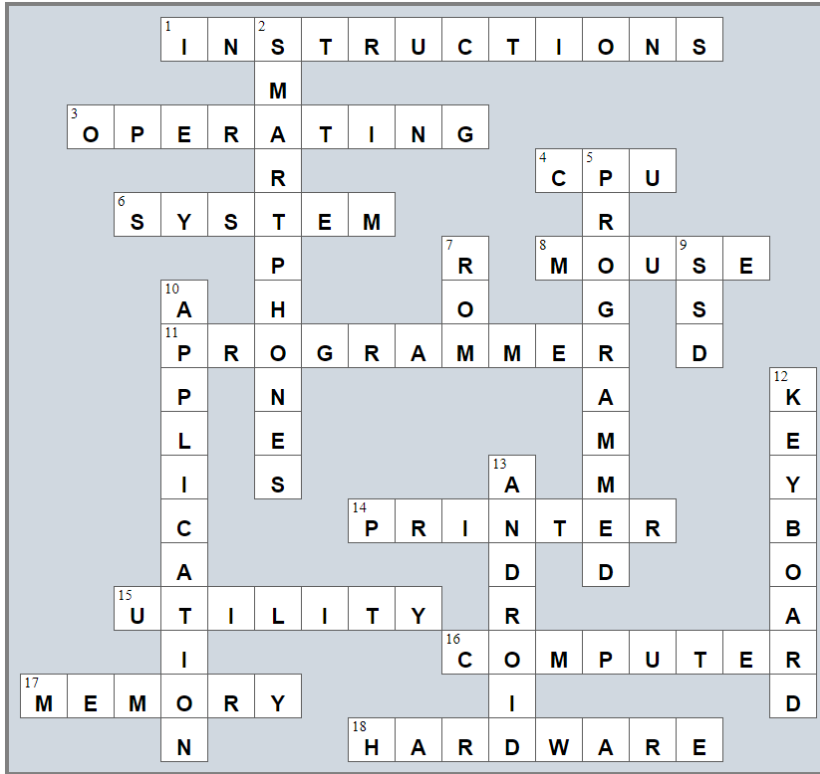
1.8 Review Questions: Multiple Choice

1. b
2. d
3. d
4. c
5. f
6. d
7. c
8. b
9. c
10. b
11. a

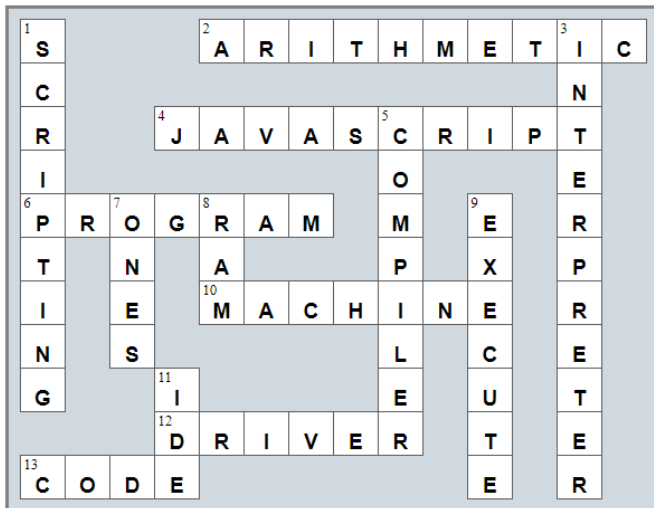
Review in "Introductory Knowledge"

Review Crossword Puzzles

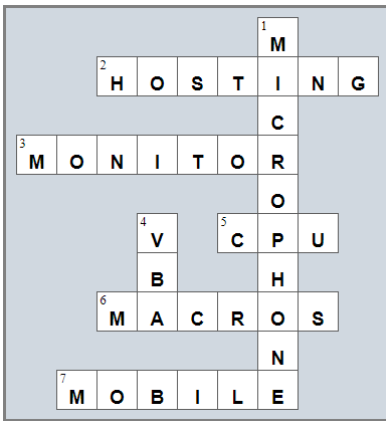
1.



2.



3.



Chapter 4

4.17 Review Questions: True/False

1. true
2. false
3. false
4. false
5. false
6. true
7. false
8. true
9. true
10. true
11. false
12. false
13. true
14. true
15. false
16. true
17. false
18. false
19. false
20. true
21. true
22. true
23. false
24. false
25. false
26. false
27. true
28. true
29. false
30. false
31. false
32. false
33. false
34. true
35. false
36. false
37. false
38. false
39. true
40. true
41. false

4.18 Review Questions: Multiple Choice

1. c
2. b
3. c
4. a
5. a
6. a
7. b
8. d
9. a
10. d

Chapter 5

5.8 Review Questions: True/False

- | | |
|----------|-----------|
| 1. false | 10. false |
| 2. false | 11. true |
| 3. true | 12. false |
| 4. false | 13. true |
| 5. false | 14. true |
| 6. true | 15. true |
| 7. false | 16. true |
| 8. false | 17. false |
| 9. true | |

5.9 Review Questions: Multiple Choice

- | | |
|------|------|
| 1. e | 5. c |
| 2. a | 6. c |
| 3. d | 7. d |
| 4. b | 8. a |

5.10 Review Exercises

- 1 - c, 2 - d, 3 - a, 4 - b
- 1 - d, 2 - c, 3 - b, 4 - a
-

Value	Data Type	Declaration and Initialization
The name of my friend	String	<code>string name = "Mark";</code>
My address	String	<code>string address = "254 Lookout Rd. Wilson, NY 27893";</code>
The average daily temperature	Float	<code>double average = 70.3;</code>
A telephone number	String	<code>string phoneNumber = "1-891-764-2410";</code>
My Social Security Number (SSN)	String	<code>string ssn = "123-45-6789";</code>
The speed of a car	Float	<code>double speed = 90.5;</code>
The number of children in a family	Integer	<code>int children = 3;</code>

Chapter 6

6.4 Review Questions: True/False

1. true
2. true
3. true
4. false
5. false

6.5 Review Questions: Multiple Choice

1. c
2. a
3. b
4. b

Chapter 7

7.7 Review Questions: True/False

1. false
2. true
3. false
4. false
5. false
6. false
7. false
8. false
9. true
10. false
11. false
12. true
13. false
14. false
15. false
16. true
17. false
18. true
19. false
20. false
21. false
22. true
23. false
24. false

7.8 Review Questions: Multiple Choice

1. c
2. c
3. b
4. d
5. b
6. d
7. d
8. c

7.9 Review Exercises

1. ii, iv, v, ix, x
2. i. String, ii. Boolean, iii. String, iv. String, v. Float, vi. Integer
3. i. d, ii. f, iii. c, iv. e
4. i. 26, ii. 28
5. i. 5, ii. 6
6. i. 1, ii. 0, iii. 1, iv. 1, v. 0, vi. 1
7. i. $2 * 3$, ii. 4
8. i. 2, ii. 0, iii. 1, iv. 0, v. 0, vi. 0
9. i. 2, ii. 5
10. My name is George Malkovich
11. i. (-3), ii. 1
12. California California California

Chapter 8

8.2 Review Questions: True/False

1. false
2. true
3. false
4. false

8.3 Review Exercises

1. Solution

Step	Statement	Notes	a	b	c	d
1	cin >> a	User enters value 3	3.0	?	?	?
2	b = a + 10		3.0	13.0	?	?
3	a = b * (a - 3)		0.0	13.0	?	?
4	c = 3 * b / 6		0.0	13.0	6.5	?
5	d = c * c		0.0	13.0	6.5	42.25
6	d--		0.0	13.0	6.5	41.25
7	cout << d << endl	It displays: 41.25				

2. Solution

For the input value of 3

Step	Statement	a	b	c	d
1	cin >> a	3	?	?	?
2	a = (a + 1) * (a + 1) + 6 / 3 * 2 + 20	40	?	?	?
3	b = a % 13	40	1	?	?
4	c = b % 7	40	1	1	?
5	d = a * b * c	40	1	1	40
6	cout << a << ", " << b << ", " << c << ", " << d << endl	It displays: 40, 1, 1, 40			

For the input value of 4

Step	Statement	a	b	c	d
1	cin >> a	4	?	?	?
2	a = (a + 1) * (a + 1) + 6 / 3 * 2 + 20	49	?	?	?
3	b = a % 13	49	10	?	?
4	c = b % 7	49	10	3	?
5	d = a * b * c	49	10	3	1470
6	cout << a << ", " << b << ", " << c << ", " << d << endl	It displays: 49, 10, 3, 1470			

For the input value of 1

Step	Statement	a	b	c	d
1	cin >> a	1	?	?	?

2	<code>a = (a + 1) * (a + 1) + 6 / 3 * 2 + 20</code>	28	?	?	?
3	<code>b = a % 13</code>	28	2	?	?
4	<code>c = b % 7</code>	28	2	2	?
5	<code>d = a * b * c</code>	28	2	2	112
6	<code>cout << a << ", " << b << ", " << c << ", " << d << endl</code>	It displays: 28, 2, 2, 112			

3. Solution

For the input values of 8, 4

Step	Statement	a	b	c	d	e
1	<code>cin >> a</code>	8	?	?	?	?
2	<code>cin >> b</code>	8	4	?	?	?
3	<code>c = a + b</code>	8	4	12	?	?
4	<code>d = 1 + a / b * c + 2</code>	8	4	12	27	?
5	<code>e = c + d</code>	8	4	12	27	39
6	<code>c += d + e</code>	8	4	78	27	39
7	<code>e--</code>	8	4	78	27	38
8	<code>d -= c + d % c</code>	8	4	78	-78	38
9	<code>cout << c << ", " << d << ", " << e << endl</code>	It displays: 78, -78, 38				

For the input values of 4, 4

Step	Statement	a	b	c	d	e
1	<code>cin >> a</code>	4	?	?	?	?
2	<code>cin >> b</code>	4	4	?	?	?
3	<code>c = a + b</code>	4	4	8	?	?
4	<code>d = 1 + a / b * c + 2</code>	4	4	8	11	?
5	<code>e = c + d</code>	4	4	8	11	19
6	<code>c += d + e</code>	4	4	38	11	19
7	<code>e--</code>	4	4	38	11	18
8	<code>d -= c + d % c</code>	4	4	38	-38	18
9	<code>cout << c << ", " << d << ", " << e << endl</code>	It displays: 38, -38, 18				

Chapter 9

9.3 Review Exercises

1. Solution

The statement $S = S1 + S3 + SS$ is wrong. It must be $S = S1 + S3 + S5$

2. Solution

For the input values of 5, 5

Step	Statement	a	b	c	d	e
1	<code>cin >> a</code>	5	?	?	?	?
2	<code>cin >> b</code>	5	5	?	?	?
3	<code>c = a + b</code>	5	5	10	?	?
4	<code>d = 5 + a / b * c + 2</code>	5	5	10	17	?
5	<code>e = c - d</code>	5	5	10	17	-7
6	<code>c += d + c</code>	5	5	37	17	-7
7	<code>e--</code>	5	5	37	17	-8
8	<code>d += e + c % b</code>	5	5	37	11	-8
9	<code>cout << c << ", " << d << ", " << e << endl</code>	It displays: 37, 11, -8				

For the input values of 4, 2

Step	Statement	a	b	c	d	e
1	<code>cin >> a</code>	4	?	?	?	?
2	<code>cin >> b</code>	4	2	?	?	?
3	<code>c = a + b</code>	4	2	6	?	?
4	<code>d = 5 + a / b * c + 2</code>	4	2	6	19	?
5	<code>e = c - d</code>	4	2	6	19	-13
6	<code>c += d + c</code>	4	2	31	19	-13
7	<code>e--</code>	4	2	31	19	-14
8	<code>d += e + c % b</code>	4	2	31	6	-14
9	<code>cout << c << ", " << d << ", " << e << endl</code>	It displays: 31, 6, -14				

3. Solution

For the input value of 5

Step	Statement	a	b	c
1	<code>cin >> b</code>	?	5	?
2	<code>c = 5</code>	?	5	5
3	<code>c = c * b</code>	?	5	25
4	<code>a = 3 * c % 10</code>	5	5	25

5	<code>cout << a << endl</code>	It displays: 5
----------	--	----------------

For the input value of 4

Step	Statement	a	b	c
1	<code>cin >> b</code>	?	4	?
2	<code>c = 5</code>	?	4	5
3	<code>c = c * b</code>	?	4	20
4	<code>a = 3 * c % 10</code>	0	4	20
5	<code>cout << a << endl</code>	It displays: 0		

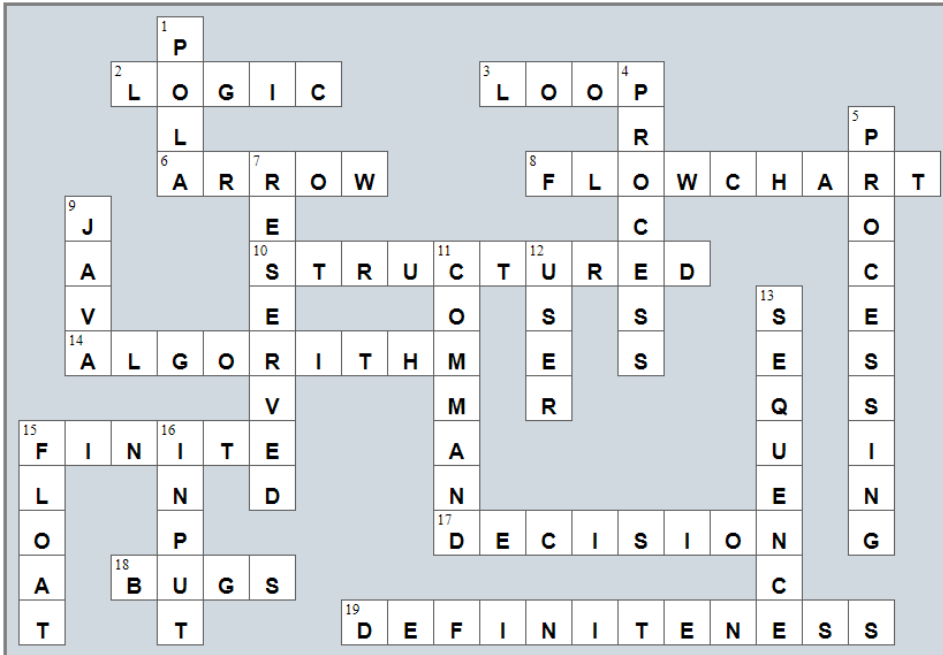
For the input value of 15

Step	Statement	a	b	c
1	<code>cin >> b</code>	?	15	?
2	<code>c = 5</code>	?	15	5
3	<code>c = c * b</code>	?	15	75
4	<code>a = 3 * c % 10</code>	5	15	75
5	<code>cout << a << endl</code>	It displays: 5		

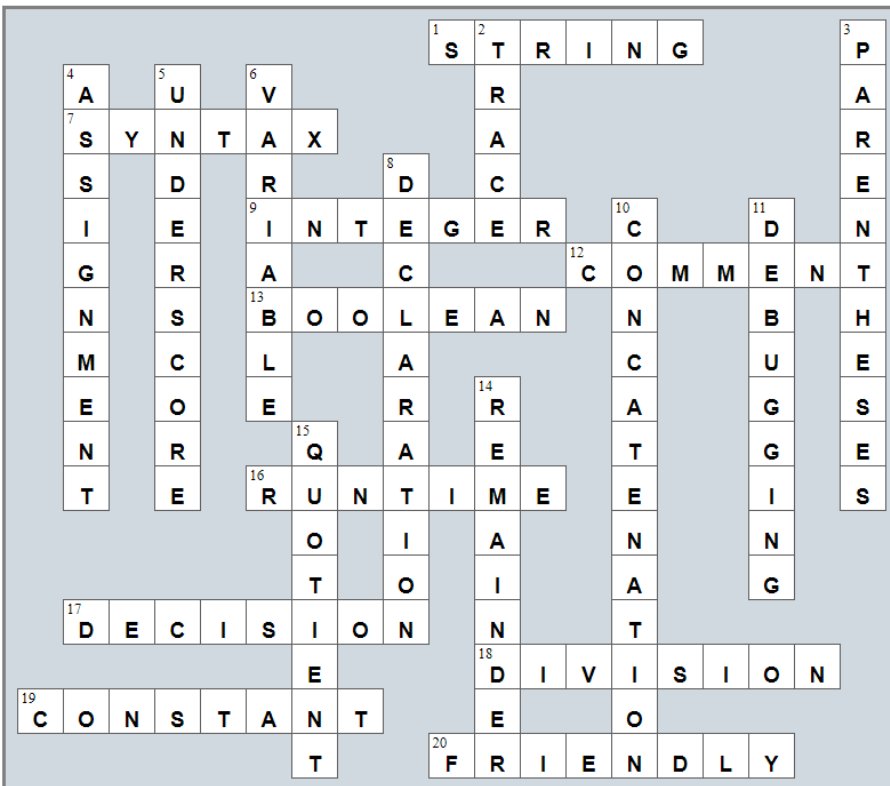
Review in "Getting Started with C#"

Review Crossword Puzzles

1.



2.



Chapter 10

10.2 Review Exercises

1. Solution

```
#include <iostream>
using namespace std;
int main() {
    double gallons, milesDriven, mpg;

    cout << "Enter miles driven: ";
    cin >> milesDriven;
    cout << "Enter gallons of gas used: ";
    cin >> gallons;

    mpg = milesDriven / gallons;

    cout << "Your car's MPG is: " << mpg << endl;
    return 0;
}
```

2. Solution

```
#include <iostream>
using namespace std;
int main() {
    double b, h, area;

    cout << "Enter base: ";
    cin >> b;
    cout << "Enter height: ";
    cin >> h;

    area = 0.5 * b * h;

    cout << area << endl;
    return 0;
}
```

3. Solution

```
#include <iostream>
using namespace std;
int main() {
    double angle1, angle2, angle3;

    cout << "Enter 1st angle: ";
    cin >> angle1;
    cout << "Enter 2nd angle: ";
    cin >> angle2;

    angle3 = 180 - angle1 - angle2;
}
```

```
    cout << angle3 << endl;
    return 0;
}
```

4. Solution

```
#include <iostream>
using namespace std;
int main() {
    int g1, g2, g3, g4;
    double average;

    cout << "Enter 1st grade: ";
    cin >> g1;
    cout << "Enter 2nd grade: ";
    cin >> g2;
    cout << "Enter 3rd grade: ";
    cin >> g3;
    cout << "Enter 4th grade: ";
    cin >> g4;

    average = (g1 + g2 + g3 + g4) / 4.0;

    cout << average << endl;
    return 0;
}
```

5. Solution

```
#include <iostream>
using namespace std;
const double PI = 3.14159;

int main() {
    double r, perimeter;

    cout << "Enter radius: ";
    cin >> r;

    perimeter = 2 * PI * r;

    cout << perimeter << endl;
    return 0;
}
```

6. Solution

```
#include <iostream>
using namespace std;
const double PI = 3.14159;

int main() {
    double d, radius, volume;
```

```

    cout << "Enter diameter (in meters): ";
    cin >> d;

    radius = d / 2;
    volume = 4 / 3 * PI * radius * radius * radius;

    cout << volume << endl;
    return 0;
}

```

7. Solution

Only a), e) and g) are syntactically correct. The latter is more user friendly.

8. Solution

```

#include <iostream>
using namespace std;
int main() {
    string firstName, lastName, middleName, title;

    cout << "First name: ";
    cin >> firstName;
    cout << "Middle name: ";
    cin >> middleName;
    cout << "Last name: ";
    cin >> lastName;
    cout << "Title: ";
    cin >> title;

    cout << title << " " << firstName << " " << middleName << " " << lastName << endl;
    cout << firstName << " " << middleName << " " << lastName << endl;
    cout << lastName << ", " << firstName << endl;
    cout << lastName << ", " << firstName << " " << middleName << endl;
    cout << lastName << ", " << firstName << " " << middleName << ", " << title << endl;
    cout << firstName << " " << lastName << endl;
    return 0;
}

```

9. Solution

```

#include <iostream>
using namespace std;
const double PI = 3.14159;

int main() {
    double d, radius, perimeter, area, volume;

    cout << "Enter diameter: ";
    cin >> d;

    radius = d / 2;
    perimeter = 2 * PI * radius;

```

```
    area = PI * radius * radius;
    volume = 4 / 3 * PI * radius * radius * radius;

    cout << radius << " " << perimeter << " " << area << " " << volume << endl;
    return 0;
}
```

10. Solution

```
#include <iostream>
using namespace std;
int main() {
    double charge, tip, tax, total;

    cout << "Enter charge for a meal: ";
    cin >> charge;

    tip = charge * 10 / 100;
    tax = charge * 7 / 100;

    total = charge + tip + tax;

    cout << total << endl;
    return 0;
}
```

11. Solution

```
#include <iostream>
using namespace std;
int main() {
    int minutes, seconds, totalSeconds;
    double s, a;

    cout << "Enter the distance traveled (in meters): ";
    cin >> s;
    cout << "Enter the minutes traveled: ";
    cin >> minutes;
    cout << "Enter the seconds traveled: ";
    cin >> seconds;

    totalSeconds = minutes * 60 + seconds;

    a = 2 * s / (totalSeconds * totalSeconds);

    cout << a << endl;
    return 0;
}
```

12. Solution

```
#include <iostream>
using namespace std;
int main() {
```



```
double f, c;

cout << "Enter temperature in Fahrenheit: ";
cin >> f;

c = 5 / 9 * (f - 32);

cout << c << endl;
return 0;
}
```

13. Solution

```
#include <iostream>
using namespace std;
int main() {
    int w, h;
    double bmi;

    cout << "Enter weight in pounds: ";
    cin >> w;
    cout << "Enter height in inches: ";
    cin >> h;

    bmi = w * 703.0 / (h * h);

    cout << bmi << endl;
    return 0;
}
```

14. Solution

```
#include <iostream>
using namespace std;
int main() {
    double sTotal, tip, total;
    int gRate;

    cout << "Enter subtotal: ";
    cin >> sTotal;
    cout << "Enter gratuity rate (0 - 100): ";
    cin >> gRate;

    tip = sTotal * gRate / 100;

    total = sTotal + tip;

    cout << "Tip is $" << tip << " and total is $" << total << endl;
    return 0;
}
```

15. Solution

```
#include <iostream>
```

```
using namespace std;
const double VAT = 0.20;

int main() {
    double btaxPrice1, btaxPrice2, btaxPrice3, ataxPrice1, ataxPrice2, ataxPrice3, avg;

    cout << "Enter before-tax price 1: ";
    cin >> btaxPrice1;
    cout << "Enter before-tax price 2: ";
    cin >> btaxPrice2;
    cout << "Enter before-tax price 3: ";
    cin >> btaxPrice3;

    ataxPrice1 = btaxPrice1 + btaxPrice1 * VAT;
    ataxPrice2 = btaxPrice2 + btaxPrice2 * VAT;
    ataxPrice3 = btaxPrice3 + btaxPrice3 * VAT;

    avg = (ataxPrice1 + ataxPrice2 + ataxPrice3) / 3;

    cout << avg << endl;
    return 0;
}
```

16. Solution

```
#include <iostream>
using namespace std;
const int VAT = 0.20;

int main() {
    double ataxPrice, btaxPrice;

    cout << "Enter after-tax price: ";
    cin >> ataxPrice;

    btaxPrice = ataxPrice / (1 + VAT);

    cout << btaxPrice << endl;
    return 0;
}
```

17. Solution

```
#include <iostream>
using namespace std;
int main() {
    double iPrice, fPrice, saved;
    int discount;

    cout << "Enter price: ";
    cin >> iPrice;
    cout << "Enter discount (0 - 100): ";
    cin >> discount;
```

```
fPrice = iPrice - iPrice * discount / 100;
saved = iPrice - fPrice;

cout << fPrice << " " << saved << endl;
return 0;
}
```

18. Solution

```
#include <iostream>
using namespace std;
const int VAT = 0.20;

int main() {
    int iKWh, fKWh, kWhConsumed;
    double cost;

    cout << "Enter kWh at the beginning of the month: ";
    cin >> iKWh;
    cout << "Enter kWh at the end of the month: ";
    cin >> fKWh;

    kWhConsumed = fKWh - iKWh;

    cost = kWhConsumed * 0.06;
    cost += cost * VAT;

    cout << kWhConsumed << " " << cost << endl;
    return 0;
}
```

19. Solution

```
#include <iostream>
using namespace std;
int main() {
    int soldYachts;
    double yachtsCost, insuranceCost, totalCost, totalEarnings;

    cout << "Enter number of yachts sold: ";
    cin >> soldYachts;

    yachtsCost = soldYachts * 1000000;
    insuranceCost = 250000 * 12;
    totalCost = yachtsCost + insuranceCost;
    totalEarnings = soldYachts * 1500000;

    cout << totalEarnings - totalCost << endl;
    return 0;
}
```

20. Solution

```
#include <iostream>
```

```
using namespace std;
int main() {
    int day, month, daysPassed;

    cout << "Enter current month: ";
    cin >> month;
    cout << "Enter current day: ";
    cin >> day;

    daysPassed = (month - 1) * 30 + day;

    cout << daysPassed << endl;
    return 0;
}
```

21. Solution

```
#include <iostream>
using namespace std;
int main() {
    int day, month, daysPassed, daysLeft;

    cout << "Enter current month: ";
    cin >> month;
    cout << "Enter current day: ";
    cin >> day;

    daysPassed = (month - 1) * 30 + day;
    daysLeft = 360 - daysPassed;

    cout << daysLeft << endl;
    return 0;
}
```

Chapter 11

11.3 Review Questions: True/False

- | | | |
|----------|-----------|-----------|
| 1. true | 7. false | 13. true |
| 2. false | 8. true | 14. true |
| 3. false | 9. false | 15. true |
| 4. false | 10. false | 16. true |
| 5. false | 11. false | 17. false |
| 6. true | 12. false | 18. false |

11.4 Review Questions: Multiple Choice

- | | | |
|------|------|------|
| 1. d | 3. b | 5. a |
| 2. d | 4. c | 6. b |

11.5 Review Exercises

1. Solution

For the input value of 9

Step	Statement	a	b	c
1	cin >> a	9.0	?	?
2	a += 6 / sqrt(a) * 2 + 20.4	33.4	?	?
3	b = round(a) % 4	33.4	1.0	?
4	c = b % 3	33.4	1.0	1.0
5	cout << a << ", " << b << ", " << c << endl	It displays: 33.4, 1, 1		

For the input value of 4

Step	Statement	a	b	c
1	cin >> a	4.0	?	?
2	a += 6 / sqrt(a) * 2 + 20.4	30.4	?	?
3	b = round(a) % 4	30.4	2.0	?
4	c = b % 3	30.4	2.0	2.0
5	cout << a << ", " << b << ", " << c << endl	It displays: 30.4, 2, 2		

2. Solution

For the input value of -2

Step	Statement	a	b	c
1	cin >> a	-2	?	?
2	b = abs(a) % 4 + pow(a, 4)	-2	18	?
3	c = b % 5	-2	18	3
4	cout << b << ", " << c << endl	It displays: 18, 3		

For the input value of -3

Step	Statement	a	b	c
1	cin >> a	-3	?	?
2	b = abs(a) % 4 + pow(a, 4)	-3	84	?
3	c = b % 5	-3	84	4
4	cout << b << ", " << c << endl	It displays: 84, 4		

3. Solution

```
#define _USE_MATH_DEFINES //This is necessary for Visual Studio Community IDE
#include <iostream>
#include <cmath>
using namespace std;
int main() {
    double degrees, radians;

    cout << "Enter angle in radians: ";
    cin >> radians;

    degrees = radians * 180 / M_PI;

    cout << degrees << endl;
    return 0;
}
```

4. Solution

```
#include <iostream>
#include <cmath>
using namespace std;
int main() {
    double a, b, hypotenuse;

    cout << "Enter right angle side A of a right-angled triangle: ";
    cin >> a;
    cout << "Enter right angle side B of a right-angled triangle: ";
    cin >> b;

    hypotenuse = sqrt(pow(a, 2) + pow(b, 2));

    cout << hypotenuse << endl;
    return 0;
}
```

5. Solution

```
#define _USE_MATH_DEFINES //This is necessary for Visual Studio Community IDE
#include <iostream>
#include <cmath>
using namespace std;
int main() {
```

```
double adjacent, opposite, th;

cout << "Enter angle  $\theta$  (in degrees) of a right-angled triangle: ";
cin >> th;
cout << "Enter length of adjacent side: ";
cin >> adjacent;

opposite = tan(th * M_PI / 180) * adjacent;

cout << opposite << endl;
return 0;
}
```

Chapter 12

12.2 Review Exercises

1. Solution

- i. a, e, g, h
- ii. c, f

2. Solution

- i. $y = \text{pow}(x + 3, 5 * w) / (7 * (x - 4))$
- ii. $y = \text{pow}(3 * \text{pow}(x, 2) - \text{pow}(x, 3) / 4, 1 / 5.0)$
- iii. $y = \text{sqrt}(\text{pow}(x, 4) - 2 * \text{pow}(x, 3) - 7 * x * x + x) / \text{pow}(4 * (7 * \text{pow}(x, 4) - 3 / 4.0 * \text{pow}(x, 3)) * (7 * x * x + x), 1 / 3.0)$
- iv. $y = x / (x - 3 * (x - 1)) + x * \text{pow}(x - 1, 1 / 5.0) / ((\text{pow}(x, 3) - 2) * \text{pow}(x - 1, 3))$
- v. $y = \text{pow}(\sin(M_PI / 3) - \cos(M_PI / 2 * w), 2)$
- vi. $y = \text{pow}(\sin(M_PI / 2 * x) + \cos(3 * M_PI / 2 * w), 3) / \text{pow}(\tan(2 * M_PI / 3 * w) - \sin(M_PI / 2 * x), 0.5) + 6$

3. Solution

```
#include <iostream>
#include <cmath>
using namespace std;
int main() {
    double x, y;

    cout << "Enter value for x: ";
    cin >> x;

    y = sqrt(x * x + 1) * (pow(x, 3) + pow(x, 2));

    cout << y << endl;
    return 0;
}
```

4. Solution

```
#include <iostream>
using namespace std;
int main() {
    double x, y;

    cout << "Enter value for x: ";
    cin >> x;

    y = 7 * x / (2 * x + 4 * (x * x + 4));

    cout << y << endl;
    return 0;
}
```


5. Solution

```
#include <iostream>
#include <cmath>
using namespace std;
int main() {
    double w, x, y;

    cout << "Enter value for x: ";
    cin >> x;
    cout << "Enter value for w: ";
    cin >> w;

    y = pow(x, x + 1) / pow(tan(2 * w / 3 + 5) + tan(x / 2 + 1), 3);

    cout << y << endl;
    return 0;
}
```

6. Solution

```
#include <iostream>
#include <cmath>
using namespace std;
int main() {
    double w, x, y;

    cout << "Enter value for x: ";
    cin >> x;
    cout << "Enter value for w: ";
    cin >> w;

    y = (3 + w) / (6 * x + 7 * (x + 4)) + x * pow(3 * w + 1, 1 / 5) * (5 * x + 4) / ((pow(x, 3) + 3) *
pow(x - 1, 6));

    cout << y << endl;
    return 0;
}
```

7. Solution

```
#include <iostream>
#include <cmath>
using namespace std;
int main() {
    double w, x, y;

    cout << "Enter value for x: ";
    cin >> x;
    cout << "Enter value for w: ";
    cin >> w;
```

```
y = pow(x, x) / pow(sin(2 * w / 3 + 5) - x, 2) + pow(sin(3 * x) + w, x + 1) / pow(sqrt(7 * w), 3 / 2);  
  
cout << y << endl;  
return 0;  
}
```

8. Solution

```
#include <iostream>  
#include <cmath>  
using namespace std;  
int main() {  
    double a, b, c, area, semi;  
  
    cout << "Enter length A: ";  
    cin >> a;  
    cout << "Enter length B: ";  
    cin >> b;  
    cout << "Enter length C: ";  
    cin >> c;  
  
    semi = (a + b + c) / 2;  
    area = sqrt(semi * (semi - a) * (semi - b) * (semi - c));  
  
    cout << area << endl;  
    return 0;  
}
```

Chapter 13

13.2 Review Exercises

1. Solution

```
#include <iostream>
using namespace std;
int main() {
    int lastDigit, n, result;

    cout << "Enter an integer: ";
    cin >> n;

    lastDigit = n % 10;
    result = lastDigit * 8;

    cout << result << endl;
    return 0;
}
```

2. Solution

```
#include <iostream>
using namespace std;
int main() {
    int digit1, digit2, digit3, digit4, digit5, number, r, reversedNumber;

    cout << "Enter a five-digit integer: ";
    cin >> number;

    digit5 = number % 10;
    r = (int)(number / 10);

    digit4 = r % 10;
    r = (int)(r / 10);

    digit3 = r % 10;
    r = (int)(r / 10);

    digit2 = r % 10;
    digit1 = (int)(r / 10);

    reversedNumber = digit5 * 10000 + digit4 * 1000 + digit3 * 100 + digit2 * 10 + digit1;
    cout << number << " + " << reversedNumber << " = " << number + reversedNumber << endl;
    return 0;
}
```

3. Solution

```
#include <iostream>
using namespace std;
int main() {
```

```
int n, result;

cout << "Enter an integer: ";
cin >> n;

result = n % 2;

cout << result << endl;
return 0;
}
```

4. Solution

```
#include <iostream>
using namespace std;
int main() {
    int n, result;

    cout << "Enter an integer: ";
    cin >> n;

    result = 1 - n % 2;

    cout << result << endl;
    return 0;
}
```

5. Solution

```
#include <iostream>
using namespace std;
int main() {
    int days, hours, minutes, number, r, seconds, weeks;

    cout << "Enter an elapsed time in seconds: ";
    cin >> number;

    weeks = (int)(number / 604800); // 60 * 60 * 24 * 7 = 604800
    r = number % 604800;

    days = (int)(r / 86400); // 60 * 60 * 24 = 86400
    r = r % 86400;

    hours = (int)(r / 3600);
    r = r % 3600;

    minutes = (int)(r / 60);
    seconds = r % 60;

    cout << weeks << " week(s) " << days << " day(s) " << hours << " hour(s) ";
    cout << minutes << " minute(s) and " << seconds << " second(s)" << endl;
    return 0;
}
```

6. Solution

```
#include <iostream>
using namespace std;
int main() {
    int amount, r, usd1, usd10, usd20, usd5;

    cout << "Enter amount of money to withdraw: ";
    cin >> amount;

    usd20 = (int)(amount / 20);
    r = amount % 20;

    usd10 = (int)(r / 10);
    r = r % 10;

    usd5 = (int)(r / 5);
    usd1 = r % 5;

    cout << usd20 << " note(s) of $20 " << usd10 << " note(s) of $10 ";
    cout << usd5 << " note(s) of $5 and " << usd1 << " note(s) of $1" << endl;
    return 0;
}
```

7. Solution

```
#include <iostream>
using namespace std;
int main() {
    int distance, feet, inches, miles, r, steps, yards;

    cout << "Enter number of steps: ";
    cin >> steps;

    distance = steps * 25;

    miles = (int)(distance / 63360);
    r = distance % 63360;

    yards = (int)(r / 36);
    r = r % 36;

    feet = (int)(r / 12);
    inches = r % 12;

    cout << miles << " mile(s) " << yards << " yard(s) ";
    cout << feet << " foot/feet and " << inches << " inch(es)" << endl;
    return 0;
}
```

Chapter 14

14.4 Review Questions: True/False

- | | | |
|----------|-----------|-----------|
| 1. true | 7. true | 13. false |
| 2. false | 8. false | 14. true |
| 3. false | 9. true | 15. true |
| 4. true | 10. false | 16. true |
| 5. true | 11. false | 17. true |
| 6. false | 12. true | |

14.5 Review Questions: Multiple Choice

1. d
2. b
3. b
4. d
5. b
6. b
7. c
8. a
9. c
10. a

14.6 Review Exercises

1. Solution

```
#include <iostream>
#include <ctime>
#include <cstdlib>
#include <boost/algorithm/string.hpp>
using namespace boost::algorithm;
using namespace std;
int main() {
    string alphabetLower, alphabetUpper;

    srand(time(NULL));

    alphabetLower = "abcdefghijklmnopqrstuvwxy";
    alphabetUpper = to_upper_copy(alphabetLower);

    cout << alphabetUpper[rand() % 26] <<
         alphabetLower[rand() % 26] <<
         alphabetLower[rand() % 26] <<
         alphabetLower[rand() % 26] <<
         alphabetLower[rand() % 26];
    return 0;
}
```

2. Solution

```
#include <iostream>
#include <ctime>
#include <cstdlib>
#include <boost/algorithm/string.hpp>
using namespace boost::algorithm;
using namespace std;
int main() {
    string name, x;
    srand(time(NULL));

    cout << "Enter name: ";
    cin >> name;

    x = replace_all_copy(to_lower_copy(name), " ", "");

    cout << x[rand() % x.length()] <<
         x[rand() % x.length()] <<
         x[rand() % x.length()] <<
         (1000 + rand() % (9999 - 1000 + 1));

    return 0;
}
```

3. Solution

First approach

```
#include <iostream>
using namespace std;
int main() {
    int number, reversedNumber;
    string sNumber, digit1, digit2, digit3;

    cout << "Enter a three-digit integer: ";
    cin >> number;

    sNumber = to_string(number);

    digit1 = sNumber[0];
    digit2 = sNumber[1];
    digit3 = sNumber[2];

    reversedNumber = 100 * stoi(digit3) + 10 * stoi(digit2) + stoi(digit1);

    cout << reversedNumber << endl;
    return 0;
}
```

Second approach

```
#include <iostream>
using namespace std;
int main() {
    int number, reversedNumber;
    string sNumber, digit1, digit2, digit3;

    cout << "Enter a three-digit integer: ";
    cin >> number;

    sNumber = to_string(number);

    digit1 = sNumber[0];
    digit2 = sNumber[1];
    digit3 = sNumber[2];

    reversedNumber = stoi(digit3 + digit2 + digit1);

    cout << reversedNumber << endl;
    return 0;
}
```

4. Solution

```
#include <iostream>
#include <boost/algorithm/string.hpp>

using namespace boost::algorithm;
using namespace std;
int main() {
```



```
string firstName, lastName, middleName;

cout << "First name: ";
cin >> firstName;
cout << "Middle name: ";
cin >> middleName;
cout << "Last name: ";
cin >> lastName;

firstName = to_upper_copy(firstName.substr(0, 1)) + to_lower_copy(firstName.substr(1));
middleName = to_upper_copy(middleName.substr(0, 1)) + to_lower_copy(middleName.substr(1));
lastName = to_upper_copy(lastName.substr(0, 1)) + to_lower_copy(lastName.substr(1));

cout << firstName << " " << middleName << " " << lastName << endl;
cout << firstName << " " << middleName[0] << ". " << lastName << endl;
cout << lastName << " " << firstName[0] << "." << endl;
return 0;
}
```

5. Solution

```
#include <iostream>
using namespace std;
int main() {
    string word, abbreviation;

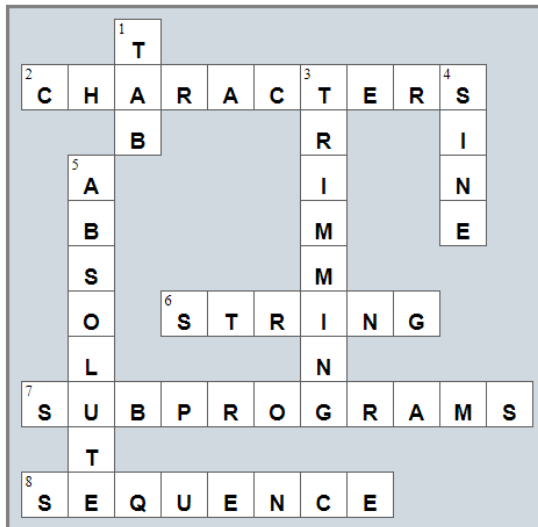
    cout << "Enter a long word: ";
    cin >> word;

    abbreviation = word[0] + to_string(word.length() - 2) + word[word.length() - 1];
    cout << abbreviation << endl;
    return 0;
}
```

Review in "Sequence Control Structures"

Review Crossword Puzzle

1.



Chapter 15

15.9 Review Questions: True/False

- | | | |
|----------|-----------|-----------|
| 1. true | 9. true | 17. false |
| 2. false | 10. true | 18. true |
| 3. false | 11. true | 19. true |
| 4. false | 12. true | 20. false |
| 5. false | 13. true | 21. true |
| 6. false | 14. true | 22. true |
| 7. false | 15. false | 23. true |
| 8. true | 16. false | |

15.10 Review Questions: Multiple Choice

- | | | |
|------|------|------|
| 1. b | 3. a | 5. c |
| 2. a | 4. a | 6. d |

15.11 Review Exercises

1. Solution

- i. c, e, g
- ii. a, j
- iii. d, f
- iv. b, h, i

2. Solution

a	b	c	a != 1	b > a	c / 2 > 2 * a
3	-5	8	true	false	false
1	10	20	false	true	true
-4	-2	-9	true	true	true

3. Solution

BE1 (Boolean Expression 1)	BE2 (Boolean Expression 2)	BE1 BE2	BE1 && BE2	!(BE2)
false	false	false	false	true
false	true	true	false	false
true	false	true	false	true
true	true	true	true	false

4. Solution

a	b	c	a > 3 c > b && c > 1	a > 3 && c > b c > 1
4	-6	2	true	true

-3	2	-4	false	false
2	5	5	false	true

5. Solution

Expression	Value
<code>pow(x + y, 3)</code>	8
<code>(x + y) / (pow(x, 2) - 14)</code>	1
<code>x - 1 == y + 5</code>	true
<code>x > 2 && y == 1</code>	false
<code>x == 1 y == -2 && !(flag == false)</code>	true
<code>!(x >= 3) && (x % 2 > 1)</code>	false

6. Solution

- i. false
- ii. true

7. Solution

- i. `age < 12 && age != 8`
- ii. `age >= 6 && age <= 9 || age == 11`
- iii. `age > 7 && age != 10 && age != 12`
- iv. `age == 6 || age == 9 || age == 11`
- v. `age >= 6 && age <= 12 && age != 8`
- vi. `age != 7 && age != 10`

8. Solution

- i. `x != 4 || y == 3`
- ii. `x + 4 > 0`
- iii. `!(x <= 5) && y != 4`
- iv. `x == false`
- v. `!(x < 4 && z <= 4)`
- vi. `x == 2 || x < -5`

9. Solution

- i. `!(x < 4 || y == 10)`
- ii. `!(x - 2 < 9)`
- iii. `!(!(x < 2) && y == 4)`
- iv. `!(x == false && y != 3)`
- v. **First approach:** `!(!(x < 2 || y < 2))`
Second approach: `x < 2 || y < 2`
- vi. `!(x == -2 || x > 2)`

Chapter 16

16.2 Review Questions: True/False

- | | |
|----------|----------|
| 1. false | 5. false |
| 2. false | 6. false |
| 3. true | 7. true |
| 4. false | 8. false |

16.3 Review Questions: Multiple Choice

- | | |
|------|------|
| 1. b | 4. d |
| 2. c | 5. c |
| 3. d | |

16.4 Review Exercises

1. Solution

The corrections/additions are in red

```
#include <iostream>
using namespace std;
int main() {
    double x, y;

    cin >> x;

    y = -5;
    if (x * y / 2 > 20)
        y *= 2;
        x += 4 * x * x;
}

cout << x << y << endl;
return 0;
}
```

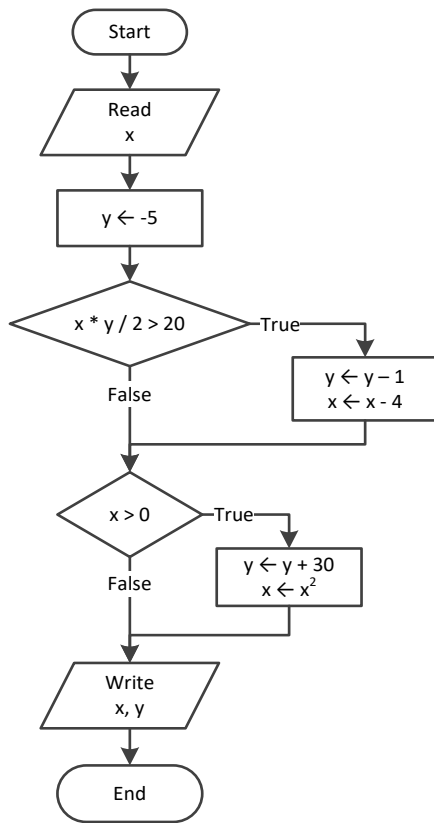
2. Solution

For the input value of 10

Step	Statement	x	y
1	cin >> x	10.0	?
2	y = -5	10.0	-5.0
3	if (x * y / 2 > 20)	false	
4	if (x > 0)	true	
5	y += 30	10.0	25.0
6	x = pow(x, 2)	100.0	25.0
7	cout << x << ", " << y << endl	It displays: 100, 25	

For the input value of -10

Step	Statement	x	y
1	cin >> x	-10.0	?
2	y = -5	-10.0	-5.0
3	if (x * y / 2 > 20)	true	
4	y--	-10.0	-6.0
5	x -= 4	-14.0	-6.0
6	if (x > 0)	false	
7	cout << x << ", " << y << endl	It displays: -14, -6	



3. Solution

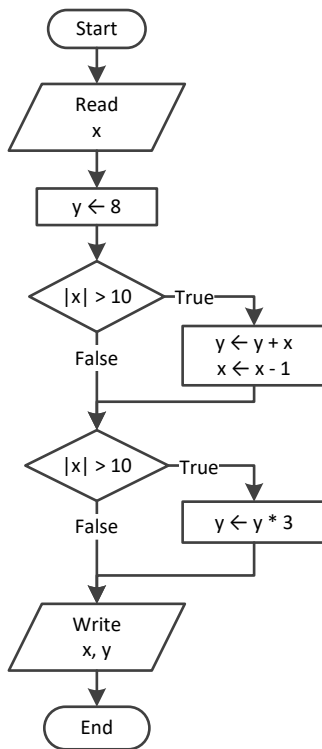
For the input value of -11

Step	Statement	x	y
1	cin >> x	-11	?
2	y = 8	-11	8
3	if (abs(x) > 10)	true	
4	y += x	-11	-3
5	x--	-12	-3
6	if (abs(x) > 10)	true	
7	y *= 3	-12	-9

8	<code>cout << x << ", " << y << endl</code>	It displays: -12, -9
----------	---	----------------------

For the input value of 11

Step	Statement	x	y
1	<code>cin >> x</code>	11	?
2	<code>y = 8</code>	11	8
3	<code>if (abs(x) > 10)</code>	true	
4	<code>y += x</code>	11	19
5	<code>x--</code>	10	19
6	<code>if (abs(x) > 10)</code>	false	
7	<code>cout << x << ", " << y << endl</code>	It displays: 10, 19	



4. Solution

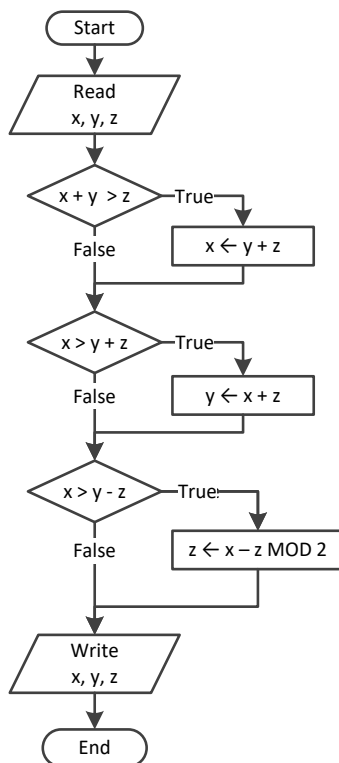
For input values of 1, 2 and 3

Step	Statement	x	y	z
1	<code>cin >> x</code>	1	?	?
2	<code>cin >> y</code>	1	2	?
3	<code>cin >> z</code>	1	2	3
4	<code>if (x + y > z)</code>	false		
5	<code>if (x > y + z)</code>	false		
6	<code>if (x > y - z)</code>	true		
7	<code>z = x - z % 2</code>	1	2	0

8	<code>cout << x << ", " << y << ", " << z << endl</code>	It displays: 1, 2, 0
----------	--	----------------------

For input values of 4, 2 and 1

Step	Statement	x	y	z
1	<code>cin >> x</code>	4	?	?
2	<code>cin >> y</code>	4	2	?
3	<code>cin >> z</code>	4	2	1
4	<code>if (x + y > z)</code>		true	
5	<code>x = y + z</code>	3	2	1
6	<code>if (x > y + z)</code>		false	
7	<code>if (x > y - z)</code>		true	
8	<code>z = x - z % 2</code>	3	2	2
9	<code>cout << x << ", " << y << ", " << z << endl</code>	It displays: 3, 2, 2		



5. Solution

```

#include <iostream>
using namespace std;
int main() {
    double x;

    cout << "Enter a number: ";
    cin >> x;

    if (x > 0) {
        cout << "Positive" << endl;
    }
}
  
```



```
    }  
    return 0;  
}
```

6. Solution

```
#include <iostream>  
using namespace std;  
int main() {  
    double x, y;  
  
    cout << "Enter a number: ";  
    cin >> x;  
    cout << "Enter a second number";  
    cin >> y;  
  
    if (x > 0 && y > 0) {  
        cout << "Both Positives" << endl;  
    }  
    return 0;  
}
```

7. Solution

```
#include <iostream>  
using namespace std;  
int main() {  
    double x, y;  
  
    cout << "Enter your age: ";  
    cin >> x;  
  
    if (x > 14) {  
        cout << "You can drive a car in Kansas (USA)" << endl;  
    }  
    return 0;  
}
```

8. Solution

```
#include <iostream>  
#include <boost/algorithm/string.hpp>  
using namespace boost::algorithm;  
using namespace std;  
int main() {  
    string str;  
  
    cout << "Enter a string: ";  
    cin >> str;  
  
    if (str == to_upper_copy(str)) {  
        cout << "Uppercase" << endl;  
    }  
}
```

```
    return 0;
}
```

9. Solution

```
#include <iostream>
using namespace std;
int main() {
    string str;

    cout << "Enter a string: ";
    cin >> str;

    if (str.length() > 20) {
        cout << "Many characters" << endl;
    }
    return 0;
}
```

10. Solution

```
#include <iostream>
using namespace std;
int main() {
    double n1, n2, n3, n4;

    cout << "Enter 1st number: ";
    cin >> n1;
    cout << "Enter 2nd number: ";
    cin >> n2;
    cout << "Enter 3rd number: ";
    cin >> n3;
    cout << "Enter 4th number: ";
    cin >> n4;

    if (n1 < 0 || n2 < 0 || n3 < 0 || n4 < 0) {
        cout << "Among the provided numbers, there is a negative one!" << endl;
    }
    return 0;
}
```

11. Solution

```
#include <iostream>
using namespace std;
int main() {
    double a, b, c;

    cout << "Enter 1st number: ";
    cin >> a;
    cout << "Enter 2nd number: ";
    cin >> b;
```

```
    if (a > b) {
        c = a;
        a = b;
        b = c;
    }

    cout << a << ", " << b << endl;
    return 0;
}
```

12. Solution

```
#include <iostream>
using namespace std;
int main() {
    double average, t1, t2, t3;

    cout << "Enter 1st temperature: ";
    cin >> t1;
    cout << "Enter 2nd temperature: ";
    cin >> t2;
    cout << "Enter 3rd temperature: ";
    cin >> t3;

    average = (t1 + t2 + t3) / 3;

    if (average > 60) {
        cout << "Heat Wave" << endl;
    }
    return 0;
}
```

Chapter 17

17.2 Review Questions: True/False

1. false
2. true
3. true
4. false
5. false
6. false

17.3 Review Questions: Multiple Choice

1. b
2. c
3. c

17.4 Review Exercises

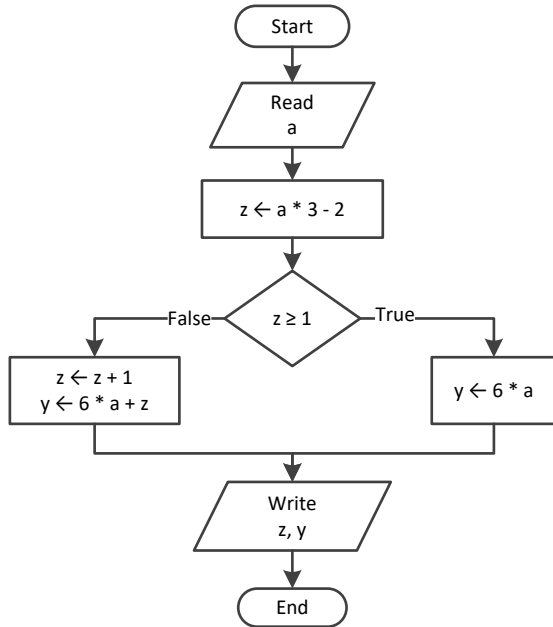
1. Solution

For input value of 3

Step	Statement	a	y	z
1	<code>cin >> a</code>	3.0	?	?
2	<code>z = a * 3 - 2</code>	3.0	?	7.0
3	<code>if (z >= 1)</code>	true		
4	<code>y = 6 * a</code>	3.0	18.0	7.0
5	<code>cout << z << ", " << y << endl</code>	It displays: 7 18		

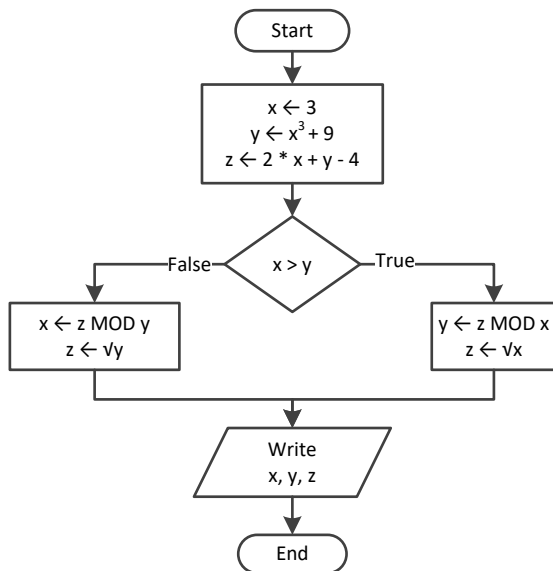
For input value of 0.5

Step	Statement	a	y	z
1	<code>cin >> a</code>	0.5	?	?
2	<code>z = a * 3 - 2</code>	0.5	?	-0.5
3	<code>if (z >= 1)</code>	false		
4	<code>z++</code>	0.5	?	0.5
5	<code>y = 6 * a + z</code>	0.5	3.5	0.5
6	<code>cout << z << ", " << y << endl</code>	It displays: 0.5, 3.5		



2. Solution

Step	Statement	x	y	z
1	$x = 3$	3.0	?	?
2	$y = \text{pow}(x, 3) + 9$	3.0	36.0	?
3	$z = 2 * x + y - 4$	3.0	36.0	38.0
4	if ($x > y$)	false		
5	$x = z \% y$	2.0	36.0	38.0
6	$z = \text{sqrt}(y)$	2.0	36.0	6.0
7	cout << x << ", " << y << ", " << z << endl	It displays: 2, 36, 6		



3. Solution

```
#include <iostream>
using namespace std;
int main() {
    double w, x, y, z;

    cin >> x;
    w = x * 3 - 15;
    z = (w + 7) * (x + 4) - 10;
    if (w > x && z > x) {
        x++;
        y = x / 2 + 4;
    }
    else {
        y = x / 4 + 2;
    }
    cout << y << endl;
    return 0;
}
```

For input value of 10

Step	Statement	x	y	w	z
1	cin >> x	10.0	?	?	?
2	w = x * 3 - 15	10.0	?	15.0	?
3	z = (w + 7) * (x + 4) - 10	10.0	?	15.0	298.0
4	if (w > x && z > x)	true			
5	x++	11.0	?	15.0	298.0
6	y = x / 2 + 4	11.0	9.5	15.0	298.0
7	cout << y << endl	It displays: 9.5			

For input value of 2

Step	Statement	x	y	w	z
1	cin >> x	2.0	?	?	?
2	w = x * 3 - 15	2.0	?	-9.0	?
3	z = (w + 7) * (x + 4) - 10	2.0	?	-9.0	-22.0
4	if (w > x && z > x)	false			
5	y = x / 4 + 2	2.0	2.5	-9.0	-22.0
6	cout << y << endl	It displays: 2.5			

4. Solution

```
#include <iostream>
using namespace std;
int main() {
    double num;
```

```
cout << "Enter a number: ";
cin >> num;

if (num > 100) {
    cout << "Provided number is greater than 100" << endl;
}
else {
    cout << "Provided number is less than or equal to 100" << endl;
}
return 0;
}
```

5. Solution

```
#include <iostream>
using namespace std;
int main() {
    double num;

    cout << "Enter a number: ";
    cin >> num;

    if (num >= 0 && num <= 100) {
        cout << "Provided number is between 0 and 100" << endl;
    }
    else {
        cout << "Provided number is not between 0 and 100" << endl;
    }
    return 0;
}
```

6. Solution

```
#include <iostream>
using namespace std;
int main() {
    string name1, name2;
    int goals1, goals2;

    cout << "Enter team name 1: ";
    cin >> name1;
    cout << "Enter team name 2: ";
    cin >> name2;

    cout << "Enter goals " << name1 << " scored: ";
    cin >> goals1;
    cout << "Enter goals " << name2 << " scored: ";
    cin >> goals2;

    if (goals1 > goals2) {
        cout << "Winner: " << name1 << endl;
    }
}
```

```
    else {  
        cout << "Winner: " << name2 << endl;  
    }  
    return 0;  
}
```

7. Solution

```
#include <iostream>  
using namespace std;  
int main() {  
    int x;  
  
    cin >> x;  
    if (x % 6 == 0) {  
        cout << x << " is a multiple of 6" << endl;  
    }  
    else {  
        cout << x << " is not a multiple of 6" << endl;  
    }  
    return 0;  
}
```

8. Solution

```
#include <iostream>  
using namespace std;  
int main() {  
    int x;  
  
    cin >> x;  
    if (x % 6 == 0 || x % 7 == 0) {  
        cout << x << " is a multiple of 6 or a multiple of 7" << endl;  
    }  
    else {  
        cout << x << " is neither a multiple of 6 nor a multiple of 7" << endl;  
    }  
    return 0;  
}
```

9. Solution

```
#include <iostream>  
using namespace std;  
int main() {  
    int x, y;  
  
    cin >> x;  
  
    y = x % 4;  
    if (y == 0) {  
        cout << x << " is a multiple of 4" << endl;  
    }  
}
```



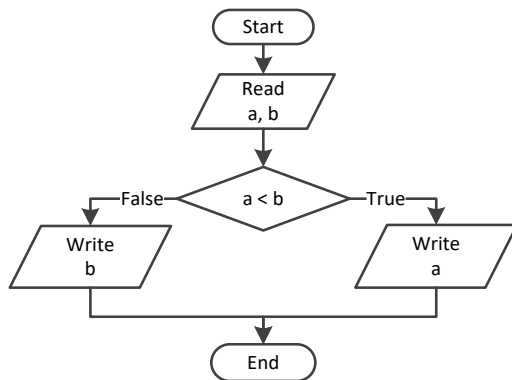
```

else {
    cout << x << " is not a multiple of 4" << endl;
}

cout << "The structure is: " << x << " = " << (int)(x / 4) << " x 4 + " << y << endl;
return 0;
}

```

10. Solution



```

#include <iostream>
using namespace std;
int main() {
    double a, b;

    cin >> a >> b;

    if (a < b) {
        cout << a << endl;
    }
    else {
        cout << b << endl;
    }
    return 0;
}

```

11. Solution

```

#include <iostream>
using namespace std;
int main() {
    double a, b, c;

    cin >> a >> b >> c;

    if (a < b + c && b < a + c && c < a + b) {
        cout << "Provided numbers can be lengths of the three sides of a triangle" << endl;
    }
    else {
        cout << "Provided numbers cannot be lengths of the three sides of a triangle" << endl;
    }
}

```

```
    return 0;
}
```

12. Solution

```
#include <iostream>
#include <cmath>
using namespace std;
int main() {
    double a, b, c;

    cin >> a >> b >> c;

    if (pow(a, 2) == pow(b, 2) + pow(c, 2) ||
        pow(b, 2) == pow(a, 2) + pow(c, 2) ||
        pow(c, 2) == pow(a, 2) + pow(b, 2)) {
        cout << "Provided numbers can be lengths of the three sides of a right triangle" << endl;
    }
    else {
        cout << "Provided numbers cannot be lengths of the three sides of a right triangle" << endl;
    }
    return 0;
}
```

13. Solution

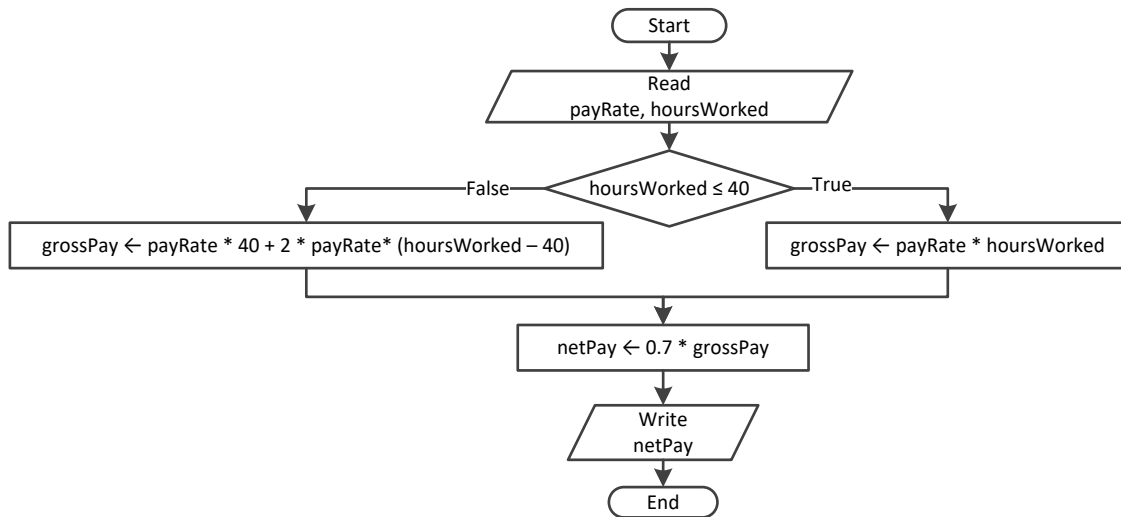
```
#include <iostream>
using namespace std;
int main() {
    double a, average, b, c;

    cout << "Enter 1st jump in meters: ";
    cin >> a;
    cout << "Enter 2nd jump in meters: ";
    cin >> b;
    cout << "Enter 3rd jump in meters: ";
    cin >> c;

    average = (a + b + c) / 3;

    if (average >= 8) {
        cout << "Qualified" << endl;
    }
    else {
        cout << "Disqualified" << endl;
    }
    return 0;
}
```

14. Solution



```

#include <iostream>
using namespace std;
int main() {
    double grossPay, netPay, payRate;
    int hoursWorked;

    cin >> payRate >> hoursWorked;

    if (hoursWorked <= 40) {
        grossPay = payRate * hoursWorked;
    }
    else {
        grossPay = payRate * 40 + 2 * payRate * (hoursWorked - 40);
    }

    netPay = 0.7 * grossPay;
    cout << netPay << endl;
    return 0;
}

```

15. Solution

```

#include <iostream>
using namespace std;
int main() {
    int miles, milesLeft, r;

    cout << "Enter miles traveled: ";
    cin >> miles;

    r = miles % 12000;

    if (r > 6000) {
        milesLeft = 12000 - r;
        cout << "Your car needs a major service in " << milesLeft << " miles" << endl;
    }
}

```

```
}  
else {  
    milesLeft = 6000 - r;  
    cout << "Your car needs a minor service in " << milesLeft << " miles" << endl;  
}  
return 0;  
}
```

16. Solution

```
#include <iostream>  
#include <cmath>  
using namespace std;  
int main() {  
    double a1, a2, s1, s2, t;  
  
    cout << "Enter the time the two cars traveled: ";  
    cin >> t;  
    cout << "Enter the acceleration for car A: ";  
    cin >> a1;  
    cout << "Enter the acceleration for car B: ";  
    cin >> a2;  
  
    s1 = 0.5 * a1 * pow(t, 2);  
    s2 = 0.5 * a2 * pow(t, 2);  
  
    cout << "Distance between them: " << abs(s1 - s2) << " meters" << endl;  
  
    if (s1 > s2) {  
        cout << "Car A is first" << endl;  
    }  
    else {  
        cout << "Car B is first" << endl;  
    }  
    return 0;  
}
```

Chapter 18

18.2 Review Questions: True/False

- | | |
|----------|----------|
| 1. true | 5. false |
| 2. false | 6. true |
| 3. false | 7. false |
| 4. false | 8. true |

18.3 Review Exercises

1. Solution

For input value of 5

Step	Statement	q	b
1	cin >> q	5	?
2	if (q > 0 && q <= 50)	true	
3	b = 1	5	1
4	cout << b << endl	It displays: 1	

For input value of 150

Step	Statement	q	b
1	cin >> q	150	?
2	if (q > 0 && q <= 50)	false	
3	else if (q > 50 && q <= 100)	false	
4	else if (q > 100 && q <= 200)	true	
5	b = 3	150	3
6	cout << b << endl	It displays: 3	

For input value of 250

Step	Statement	q	b
1	cin >> q	250	?
2	if (q > 0 && q <= 50)	false	
3	else if (q > 50 && q <= 100)	false	
4	else if (q > 100 && q <= 200)	false	
5	b = 4	250	4
6	cout << b << endl	It displays: 4	

For input value of -1

Step	Statement	q	b
1	cin >> q	-1	?
2	if (q > 0 && q <= 50)	false	
3	else if (q > 50 && q <= 100)	false	
4	else if (q > 100 && q <= 200)	false	

5	b = 4	-1	4
6	cout << b << endl	It displays: 4	

2. Solution

For input value of 5

Step	Statement	amount	discount	payment
1	cin >> amount	5.0	?	?
2	discount = 0	5.0	0.0	?
3	if (amount < 20)	true		
4	discount = 0	5.0	0.0	?
5	payment = amount - amount * discount / 100	5.0	0.0	5.0
6	cout << discount << ", " << payment << endl	It displays: 0, 5		

For input value of 150

Step	Statement	amount	discount	payment
1	cin >> amount	150.0	?	?
2	discount = 0	150.0	0.0	?
3	if (amount < 20)	false		
4	else if (amount >=20 && amount < 60)	false		
5	else if (amount >= 60 && amount < 100)	false		
6	else if (amount >= 100)	true		
7	discount = 15	150.0	15.0	?
8	payment = amount - amount * discount / 100	150.0	15.0	127.5
9	cout << discount << ", " << payment << endl	It displays: 15, 127.5		

For input value of -1

Step	Statement	amount	discount	payment
1	cin >> amount	-1.0	?	?
2	discount = 0	-1.0	0.0	?
3	if (amount < 20)	true		
4	discount = 0	-1.0	0.0	?
5	payment = amount - amount * discount / 100	-1.0	0.0	-1.0
6	cout << discount << ", " << payment << endl	It displays: 0, -1		

3. Solution

```
#include <iostream>
using namespace std;
int main() {
    double a, y;

    cin >> a;
```

```
    if (a < 1) {
        y = 5 + a;
        cout << y << endl;
    }
    else if (a < 5) {
        y = 23 / a;
        cout << y << endl;
    }
    else if (a < 10) {
        y = 5 * a;
        cout << y << endl;
    }
    else {
        cout << "Error!" << endl;
    }
    return 0;
}
```

4. Solution

```
#include <iostream>
using namespace std;
int main() {
    int n1, n2;

    cout << "Enter an integer: ";
    cin >> n1;
    cout << "Enter a second integer: ";
    cin >> n2;

    if (n1 % 2 == 0 && n2 % 2 == 0) {
        cout << "Both numbers are evens" << endl;
    }
    else if (n1 % 2 != 0 && n2 % 2 != 0) {
        cout << "Both numbers are odds" << endl;
    }
    else {
        cout << "Nothing special!" << endl;
    }
    return 0;
}
```

5. Solution

```
#include <iostream>
using namespace std;
int main() {
    string name1, name2;
    int goals1, goals2;

    cout << "Enter team name 1: ";
```

```

cin >> name1;
cout << "Enter team name 2: ";
cin >> name2;

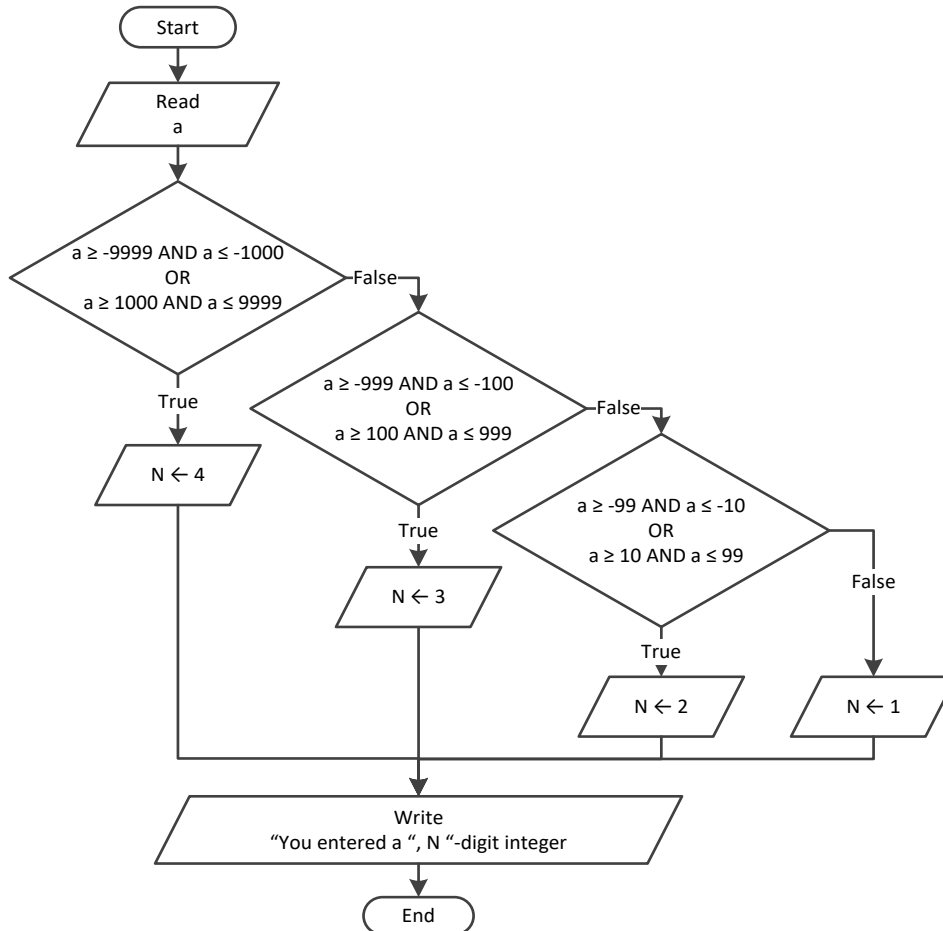
cout << "Enter goals " << name1 << " scored: ";
cin >> goals1;
cout << "Enter goals " << name2 << " scored: ";
cin >> goals2;

if (goals1 > goals2) {
    cout << "Winner: " << name1 << endl;
}
else if (goals2 > goals1) {
    cout << "Winner: " << name2 << endl;
}
else {
    cout << "It's a tie!" << endl;
}
return 0;
}

```

6. Solution

First approach




```
#include <iostream>
using namespace std;
int main() {
    int a, n;

    cin >> a;

    if (a >= -9999 && a <= -1000 || a >= 1000 && a <= 9999) {
        n = 4;
    }
    else if (a >= -999 && a <= -100 || a >= 100 && a <= 999) {
        n = 3;
    }
    else if (a >= -99 && a <= -10 || a >= 10 && a <= 99) {
        n = 2;
    }
    else {
        n = 1;
    }

    cout << "You entered a " << n << "-digit integer" << endl;
    return 0;
}
```

Second approach

```
#include <iostream>
using namespace std;
int main() {
    int a, n;

    cin >> a;

    a = abs(a);

    if (a >= 1000 && a <= 9999) {
        n = 4;
    }
    else if (a >= 100 && a <= 999) {
        n = 3;
    }
    else if (a >= 10 && a <= 99) {
        n = 2;
    }
    else {
        n = 1;
    }

    cout << "You entered a " << n << "-digit integer" << endl;
    return 0;
}
```

Third approach

```
#include <iostream>
using namespace std;
```

```

int main() {
    int a;
    string aString;

    cin >> a;

    aString = to_string(abs(a));
    cout << "You entered a " << aString.length() << "-digit integer" << endl;
    return 0;
}

```

7. Solution

First approach

```

#include <iostream>
using namespace std;
int main() {
    int a, n;

    cin >> a;

    if (a >= -9999 && a <= -1000 || a >= 1000 && a <= 9999) {
        cout << "You entered a 4-digit integer" << endl;
    }
    else if (a >= -999 && a <= -100 || a >= 100 && a <= 999) {
        cout << "You entered a 3-digit integer" << endl;
    }
    else if (a >= -99 && a <= -10 || a >= 10 && a <= 99) {
        cout << "You entered a 2-digit integer" << endl;
    }
    else if (a >= -9 && a <= 9) { //Include the value of zero
        cout << "You entered a 1-digit integer" << endl;
    }
    else {
        cout << "Error: Invalid value!" << endl;
    }
    return 0;
}

```

Second approach

```

#include <iostream>
using namespace std;
int main() {
    int a, n;

    cin >> a;

    a = abs(a);

    if (a >= 1000 && a <= 9999) {
        cout << "You entered a 4-digit integer" << endl;
    }
    else if (a >= 100 && a <= 999) {

```

```

    cout << "You entered a 3-digit integer" << endl;
}
else if (a >= 10 && a <= 99) {
    cout << "You entered a 2-digit integer" << endl;
}
else if (a >= 0 && a <= 9) {
    cout << "You entered a 1-digit integer" << endl;
}
else {
    cout << "Error: Invalid value!" << endl;
}
return 0;
}

```

Third approach

```

#include <iostream>
using namespace std;
int main() {
    int a;
    string aString;

    cin >> a;

    if (a >= -9999 && a <= 9999) {
        aString = to_string(abs(a));
        cout << "You entered a " << aString.length() << "-digit integer" << endl;
    }
    else {
        cout << "Error: Invalid value!" << endl;
    }
    return 0;
}

```

8. Solution

```

#include <iostream>
using namespace std;
int main() {
    double cad, eur, gbp, jpy, usd;
    int ch;

    cout << "1. Convert USD to Euro (EUR)" << endl;
    cout << "2. Convert USD to British Pound Sterling (GBP)" << endl;
    cout << "3. Convert USD to Japanese Yen (JPY)" << endl;
    cout << "4. Convert USD to Canadian Dollar (CAD)" << endl;

    cout << "Enter a choice: ";
    cin >> ch;

    cout << "Enter an amount in US dollars: ";
    cin >> usd;

    if (ch == 1) {

```

```
    eur = usd * 0.94;
    cout << "$" << usd << " = " << eur << " EUR" << endl;
}
else if (ch == 2) {
    gbp = usd * 0.81;
    cout << "$" << usd << " = " << gbp << " GBP" << endl;
}
else if (ch == 3) {
    jpy = usd * 149.11;
    cout << "$" << usd << " = " << jpy << " JPY" << endl;
}
else {
    cad = usd * 1.36;
    cout << "$" << usd << " = " << cad << " CAD" << endl;
}
return 0;
}
```

9. Solution

```
#include <iostream>
using namespace std;
int main() {
    int m;

    cout << "Enter the number of a month between 1 and 12: ";
    cin >> m;

    if (m <= 2 || m == 12) {
        cout << "Winter" << endl;
    }
    else if (m <= 5) {
        cout << "Spring" << endl;
    }
    else if (m <= 8) {
        cout << "Summer" << endl;
    }
    else {
        cout << "Fall (Autumn)" << endl;
    }
    return 0;
}
```

10. Solution

```
#include <iostream>
using namespace std;
int main() {
    int m;

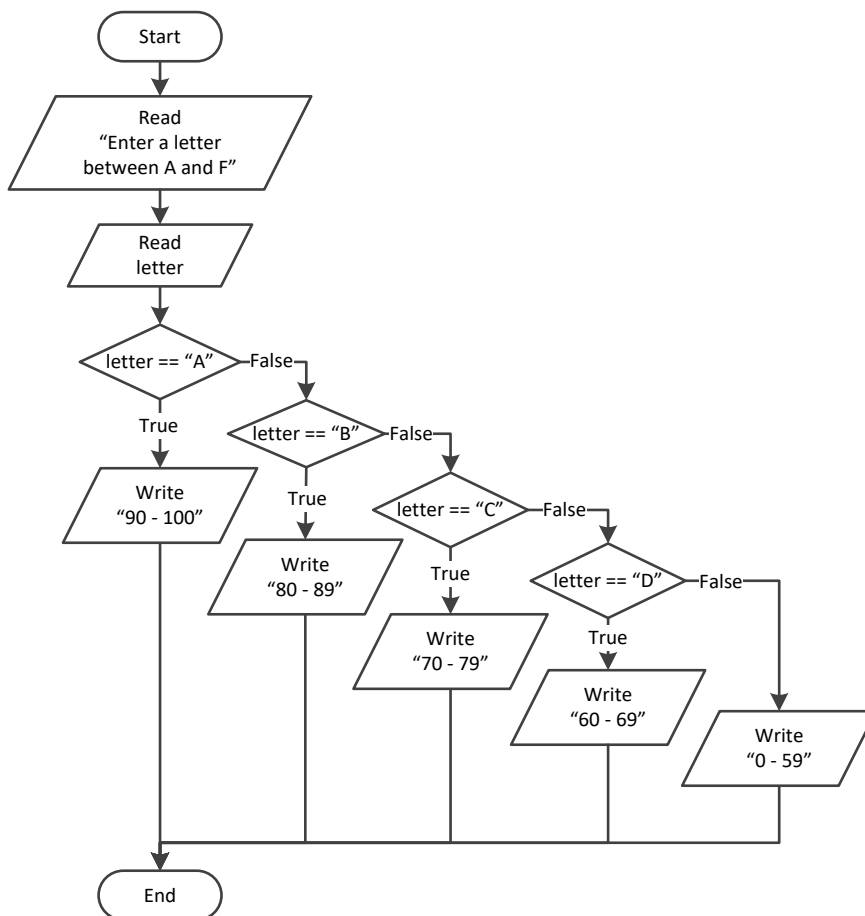
    cout << "Enter the number of a month between 1 and 12: ";
    cin >> m;
```

```

if (m < 1 || m > 12) {
    cout << "Error: Invalid value!" << endl;
else if (m <= 2 || m == 12) {
    cout << "Winter" << endl;
}
else if (m <= 5) {
    cout << "Spring" << endl;
}
else if (m <= 8) {
    cout << "Summer" << endl;
}
else {
    cout << "Fall (Autumn)" << endl;
}
return 0;
}

```

11. Solution



```

#include <iostream>
using namespace std;
int main() {
    string letter;

```

```
cout << "Enter a letter between A and F: ";
cin >> letter;

if (letter == "A") {
    cout << "90 - 100" << endl;
}
else if (letter == "B") {
    cout << "80 - 89" << endl;
}
else if (letter == "C") {
    cout << "70 - 79" << endl;
}
else if (letter == "D") {
    cout << "60 - 69" << endl;
}
else {
    cout << "0 - 59" << endl;
}
}
```

12. Solution

```
#include <iostream>
using namespace std;
int main() {
    double n;
    int x, y;
    string number;

    cout << "Enter a number between 0.0 and 9.9: ";
    cin >> n;

    x = (int)(n);
    y = (int)(n * 10) % 10;

    number = "";

    if (x == 1) {
        number += "One";
    }
    else if (x == 2) {
        number += "Two";
    }
    else if (x == 3) {
        number += "Three";
    }
    else if (x == 4) {
        number += "Four";
    }
    else if (x == 5) {
        number += "five";
    }
}
```

```
    else if (x == 6) {
        number += "six";
    }
    else if (x == 7) {
        number += "seven";
    }
    else if (x == 8) {
        number += "eight";
    }
    else if (x == 9) {
        number += "nine";
    }
    else if (x == 0) {
        number += "zero";
    }

    number += " point ";

    if (y == 1) {
        number += "one";
    }
    else if (y == 2) {
        number += "two";
    }
    else if (y == 3) {
        number += "three";
    }
    else if (y == 4) {
        number += "four";
    }
    else if (y == 5) {
        number += "five";
    }
    else if (y == 6) {
        number += "six";
    }
    else if (y == 7) {
        number += "seven";
    }
    else if (y == 8) {
        number += "eight";
    }
    else if (y == 9) {
        number += "nine";
    }
    else if (y == 0) {
        number += "zero";
    }

    cout << number;
    return 0;
}
```


Chapter 19

19.2 Review Questions: True/False

1. true
2. false
3. true
4. false
5. true
6. false
7. true

19.3 Review Exercises

1. Solution

For input value of 1

Step	Statement	a	x	y
1	cin >> a	1	?	?
2	x = 0	1	0	?
3	y = 0	1	0	0
4	case a == 1	true		
5	x = x + 5	1	5	0
6	y = y + 5	1	5	5
7	cout << x << ", " << y << endl	It displays: 5, 5		

For input value of 3

Step	Statement	a	x	y
1	cin >> a	3	?	?
2	x = 0	3	0	?
3	y = 0	3	0	0
4	case a == 1	false		
5	case a == 2	false		
6	case a == 3	true		
7	x = x - 9	3	-9	0
8	y = y + 3	3	-9	3
9	cout << x << ", " << y << endl	It displays: -9, 3		

For input value of 250

Step	Statement	a	x	y
1	cin >> a	250	?	?
2	x = 0	250	0	?
3	y = 0	250	0	0
4	case a == 1	false		
5	case a == 2	false		
6	case a == 3	false		

7	<code>x = x + 3</code>	250	3	0
8	<code>y++</code>	250	3	1
9	<code>cout << x << ", " << y << endl</code>	It displays: 3, 1		

2. Solution

For input values of 10, 2, 5

Step	Statement	a	x	y
1	<code>cin >> a</code>	10	?	?
2	<code>cin >> x</code>	10	2	?
3	<code>cin >> y</code>	10	2	5.0
4	<code>case a == 10</code>	true		
5	<code>x = x % 2</code>	10	0	5.0
6	<code>y = pow(y, 2)</code>	10	0	25.0
7	<code>cout << x << ", " << y << endl</code>	It displays: 0, 25		

For input values of 5, 2, 3

Step	Statement	a	x	y
1	<code>cin >> a</code>	5	?	?
2	<code>cin >> x</code>	5	2	?
3	<code>cin >> y</code>	5	2	3.0
4	<code>case a == 10</code>	false		
5	<code>case a == 3</code>	false		
6	<code>case a == 5</code>	true		
7	<code>x = x + 4</code>	5	6	3.0
8	<code>y += 7</code>	5	6	10.0
9	<code>cout << x << ", " << y << endl</code>	It displays: 6, 10		

For input values of 4, 6, 2

Step	Statement	a	x	y
1	<code>cin >> a</code>	4	?	?
2	<code>cin >> x</code>	4	6	?
3	<code>cin >> y</code>	4	6	2.0
4	<code>case a == 10</code>	false		
5	<code>case a == 3</code>	false		
6	<code>case a == 5</code>	false		
7	<code>x -= 3</code>	4	3	2.0
8	<code>y++</code>	4	3	3.0
9	<code>cout << x << ", " << y << endl</code>	It displays: 3, 3		

3. Solution

```
#include <iostream>
using namespace std;
int main() {
    int number;

    cout << "Enter the number of a month: ";
    cin >> number;

    switch (number) {
        case 1:
            cout << "JANUARY" << endl;
            break;
        case 2:
            cout << "FEBRUARY" << endl;
            break;
        case 3:
            cout << "MARCH" << endl;
            break;
        case 4:
            cout << "APRIL" << endl;
            break;
        case 5:
            cout << "MAY" << endl;
            break;
        case 6:
            cout << "JUNE" << endl;
            break;
        case 7:
            cout << "JULY" << endl;
            break;
        case 8:
            cout << "AUGUST" << endl;
            break;
        case 9:
            cout << "SEPTEMBER" << endl;
            break;
        case 10:
            cout << "OCTOBER" << endl;
            break;
        case 11:
            cout << "NOVEMBER" << endl;
            break;
        case 12:
            cout << "DECEMBER" << endl;
            break;
        default:
            cout << "Error" << endl;
    }
    return 0;
}
```

```
    }
```

4. Solution

```
#include <iostream>
using namespace std;
int main() {
    int choice;
    double feet, inches, miles, yards;

    cout << "1. Convert Miles to Yards" << endl;
    cout << "2. Convert Miles to Feet" << endl;
    cout << "3. Convert Miles to Inches" << endl;

    cout << "Enter a choice: ";
    cin >> choice;

    switch (choice) {
        case 1:
            cout << "Enter miles: ";
            cin >> miles;
            yards = miles * 1760;
            cout << miles << " miles = " << yards << " yards" << endl;
            break;
        case 2:
            cout << "Enter miles: ";
            cin >> miles;
            feet = miles * 5280;
            cout << miles << " miles = " << feet << " feet" << endl;
            break;
        case 3:
            cout << "Enter miles: ";
            cin >> miles;
            inches = miles * 63360;
            cout << miles << " miles = " << inches << " inches" << endl;
            break;
        default:
            cout << "Invalid choice!" << endl;
    }
    return 0;
}
```

5. Solution

```
#include <iostream>
using namespace std;
int main() {
    int number;

    cout << "Enter a number between 1 and 10: ";
    cin >> number;

    switch (number) {
```

```
    case 1:
        cout << "I" << endl;
        break;
    case 2:
        cout << "II" << endl;
        break;
    case 3:
        cout << "III" << endl;
        break;
    case 4:
        cout << "IV" << endl;
        break;
    case 5:
        cout << "V" << endl;
        break;
    case 6:
        cout << "VI" << endl;
        break;
    case 7:
        cout << "VII" << endl;
        break;
    case 8:
        cout << "VIII" << endl;
        break;
    case 9:
        cout << "IX" << endl;
        break;
    case 10:
        cout << "X" << endl;
        break;
    default:
        cout << "Error" << endl;
}
return 0;
}
```

6. Solution

```
#include <iostream>
using namespace std;
int main() {
    int bottles;

    cout << "Enter the total number of wine bottles purchased in a month: ";
    cin >> bottles;

    switch (bottles) {
        case 1:
            cout << "You are awarded 3 points" << endl;
            break;
        case 2:
            cout << "You are awarded 10 points" << endl;
```

```
        break;
    case 3:
        cout << "You are awarded 20 points" << endl;
        break;
    default:
        cout << "You are awarded 45 points" << endl;
    }
    return 0;
}
```

7. Solution

```
#include <iostream>
#include <ctime>
#include <cstdlib>
using namespace std;
int main() {
    int i;
    string name;

    srand(time(NULL));

    cout << "Enter your name: ";
    cin >> name;

    i = rand() % 3;

    switch (i) {
        case 0:
            cout << "Hello " << name << "!" << endl;
            break;
        case 1:
            cout << "Hi " << name << "!" << endl;
            break;
        case 2:
            cout << "What's up " << name << "!" << endl;
            break;
    }
    return 0;
}
```

8. Solution

```
#include <iostream>
using namespace std;
int main() {
    int num;

    cin >> num;

    switch (num) {
        case 0:
            cout << "zero" << endl;
    }
}
```

```
        break;
    case 1:
        cout << "one" << endl;
        break;
    case 2:
        cout << "two" << endl;
        break;
    case 3:
        cout << "three" << endl;
        break;
    case 4:
        cout << "four" << endl;
        break;
    case 5:
        cout << "five" << endl;
        break;
    case 6:
        cout << "six" << endl;
        break;
    case 7:
        cout << "seven" << endl;
        break;
    case 8:
        cout << "eight" << endl;
        break;
    case 9:
        cout << "nine" << endl;
        break;
    default:
        cout << "I don't know this number!" << endl;
}
return 0;
}
```

9. Solution

```
#include <iostream>
using namespace std;
int main() {
    int b;

    cout << "Enter Beaufort number: ";
    cin >> b;

    switch (b) {
        case 0:
            cout << "Calm" << endl;
            break;
        case 1:
            cout << "Light Air" << endl;
            break;
        case 2:
```

```
    cout << "Light breeze" << endl;
    break;
case 3:
    cout << "Gentle breeze" << endl;
    break;
case 4:
    cout << "Moderate breeze" << endl;
    break;
case 5:
    cout << "Fresh breeze" << endl;
    break;
case 6:
    cout << "Strong breeze" << endl;
    break;
case 7:
    cout << "Moderate gale" << endl;
    break;
case 8:
    cout << "Gale" << endl;
    break;
case 9:
    cout << "Strong gale" << endl;
    break;
case 10:
    cout << "Storm" << endl;
    break;
case 11:
    cout << "Violent storm" << endl;
    break;
case 12:
    cout << "Hurricane force" << endl;
    break;
default:
    cout << "Invalid Beaufort number!" << endl;
}
return 0;
}
```


Chapter 20

20.2 Review Questions: True/False

1. true
2. true
3. false
4. false

20.3 Review Exercises

1. Solution

For input values of 20, 1

Step	Statement	x	y
1	cin >> x	20	?
2	cin >> y	20	1
3	if (x < 30)	true	
4	case y == 1	true	
5	x = x % 3	2	1
6	y = 5	2	5
7	cout << x << ", " << y << endl	It displays: 2, 5	

For input values of 20, 3

Step	Statement	x	y
1	cin >> x	20	?
2	cin >> y	20	3
3	if (x < 30)	true	
4	case y == 1	false	
5	case y == 2	false	
6	case y == 3	true	
7	x = x + 5	25	3
8	y += 3	25	6
9	cout << x << ", " << y << endl	It displays: 25, 6	

For input values of 12, 8

Step	Statement	x	y
1	cin >> x	12	?
2	cin >> y	12	8
3	if (x < 30)	true	
4	case y == 1	false	
5	case y == 2	false	
6	case y == 3	false	
7	x -= 2	10	8

8	y++	10	9
9	cout << x << ", " << y << endl	It displays: 10, 9	

For input values of 50, 0

Step	Statement	x	y
1	cin >> x	50	?
2	cin >> y	50	0
3	if (x < 30)	false	
4	y++	50	1
5	cout << x << ", " << y << endl	It displays: 50, 1	

2. Solution

For input values of 60, 25

Step	Statement	x	y
1	cin >> x	60	?
2	cin >> y	60	25
3	if ((x + y) / 2 <= 20)	false	
4	if (y < 15)	false	
5	else if (y < 23)	false	
6	x = 2 * x + 5	125	25
7	y += 1	125	26
8	cout << x << ", " << y << endl	It displays: 125, 26	

For input values of 50, 8

Step	Statement	x	y
1	cin >> x	50	?
2	cin >> y	50	8
3	if ((x + y) / 2 <= 20)	false	
4	if (y < 15)	true	
5	x = x % 4	2	8
6	y = 2	2	2
7	cout << x << ", " << y << endl	It displays: 2, 2	

For input values of 20, 15

Step	Statement	x	y
1	cin >> x	20	?
2	cin >> y	20	15
3	if ((x + y) / 2 <= 20)	true	
4	if (y < 10)	false	
5	else if (y < 20)	true	

6	<code>x = x * 5</code>	100	15
7	<code>y += 2</code>	100	17
8	<code>cout << x << ", " << y << endl</code>	It displays: 100, 17	

For input values of 10, 30

Step	Statement	x	y
1	<code>cin >> x</code>	10	?
2	<code>cin >> y</code>	10	30
3	<code>if ((x + y) / 2 <= 20)</code>	true	
4	<code>if (y < 10)</code>	false	
5	<code>else if (y < 20)</code>	false	
6	<code>x = x - 2</code>	8	30
7	<code>y += 3</code>	8	33
8	<code>cout << x << ", " << y << endl</code>	It displays: 8, 33	

3. Solution

```
#include <iostream>
using namespace std;
int main() {
    int a;

    cin >> a;

    if (a > 1000)
        cout << "Big Positive" << endl;
    else {
        if (a > 0)
            cout << "Positive" << endl;
        else {
            if (a < -1000)
                cout << "Big Negative" << endl;
            else {
                if (a < 0)
                    cout << "Negative" << endl;
                else
                    cout << "Zero" << endl;
            }
        }
    }
    return 0;
}
```

4. Solution

First approach

```
#include <iostream>
using namespace std;
```

```
int main() {
    int age;

    cout << "Enter your age: ";
    cin >> age;

    if (age < 0) {
        cout << "Error: Invalid age!" << endl;
    }
    else {
        if (age < 16) {
            cout << "You cannot drive either a small scooter or a car" << endl;
        }
        else {
            if (age < 18) {
                cout << "You can drive a small scooter" << endl;
            }
            else {
                cout << "You can drive a car and a small scooter" << endl;
            }
        }
    }
    return 0;
}
```

Second approach

```
#include <iostream>
using namespace std;
int main() {
    int age;

    cout << "Enter your age: ";
    cin >> age;

    if (age < 0) {
        cout << "Error: Invalid age!" << endl;
    }
    else {
        if (age < 16) {
            cout << "You cannot drive either a small scooter or a car" << endl;
        }
        else if (age < 18) {
            cout << "You can drive a small scooter" << endl;
        }
        else {
            cout << "You can drive a car and a small scooter" << endl;
        }
    }
    return 0;
}
```

Third approach

```
#include <iostream>
```

```
using namespace std;
int main() {
    int age;

    cout << "Enter your age: ";
    cin >> age;

    if (age < 0) {
        cout << "Error: Invalid age!" << endl;
    }
    else if (age < 16) {
        cout << "You cannot drive either a small scooter or a car" << endl;
    }
    else if (age < 18) {
        cout << "You can drive a small scooter" << endl;
    }
    else {
        cout << "You can drive a car and a small scooter" << endl;
    }
    return 0;
}
```

5. Solution

```
#include <iostream>
using namespace std;
int main() {
    int soldHoverboards, employeesNum;
    double hoverboardsCost, insuranceCost, totalCost;
    double totalEarnings, profitLoss;

    cout << "Enter number of hoverboards sold: ";
    cin >> soldHoverboards;
    cout << "Enter number of employees: ";
    cin >> employeesNum;

    if (soldHoverboards < 0 || employeesNum <= 0) {
        cout << "Wrong value(s) entered" << endl;
    }
    else {
        hoverboardsCost = soldHoverboards * 150;
        insuranceCost = employeesNum * 1000;
        totalCost = hoverboardsCost + insuranceCost;

        totalEarnings = soldHoverboards * 250;
        profitLoss = totalEarnings - totalCost;

        if (profitLoss > 0) {
            cout << "Profit" << endl;
        }
        else if (profitLoss < 0) {
            cout << "Loss" << endl;
        }
    }
}
```

```
    else {
        cout << "Broke even" << endl;
    }
}
return 0;
}
```

6. Solution

First approach: Using nested decision structures

```
#include <iostream>
#include <ctime>
#include <cstdlib>
using namespace std;
int main() {
    int hour;
    string name;

    srand(time(NULL));

    cout << "Enter your name: ";
    cin >> name;

    hour = 1 + rand() % 24;
    cout << "The hour is " << hour << ":00" << endl;

    if (hour >= 5 && hour <= 11) {
        cout << "Good Morning " << name << "!" << endl;
    }
    else {
        if (hour >= 12 && hour <= 18) {
            cout << "Good Afternoon " << name << "!" << endl;
        }
        else {
            if (hour >= 19 && hour <= 22) {
                cout << "Good Evening " << name << "!" << endl;
            }
            else {
                cout << "Good Night " << name << "!" << endl;
            }
        }
    }
    return 0;
}
```

Second approach: Using a multiple-alternative decision structure

```
#include <iostream>
#include <ctime>
#include <cstdlib>
using namespace std;
int main() {
    int hour;
    string name;
```

```
srand(time(NULL));

cout << "Enter your name: ";
cin >> name;

hour = 1 + rand() % 24;
cout << "The hour is " << hour << ":00" << endl;

if (hour >= 5 && hour <= 11) {
    cout << "Good Morning " << name << "!" << endl;
}
else if (hour >= 12 && hour <= 18) {
    cout << "Good Afternoon " << name << "!" << endl;
}
else if (hour >= 19 && hour <= 22) {
    cout << "Good Evening " << name << "!" << endl;
}
else {
    cout << "Good Night " << name << "!" << endl;
}
return 0;
}
```

7. Solution

```
#include <iostream>
#include <cmath>
using namespace std;
int main() {
    double a, b, c;

    cout << "Enter the three sides of a triangle: ";
    cin >> a >> b >> c;

    if (a >= b + c || b >= a + c || c >= a + b) {
        cout << "Provided numbers cannot be lengths of the three sides of a triangle" << endl;
    }
    else {
        if (a == b && b == c) {
            cout << "Equilateral" << endl;
        }
        else if (pow(a, 2) == pow(b, 2) + pow(c, 2) ||
                pow(b, 2) == pow(a, 2) + pow(c, 2) ||
                pow(c, 2) == pow(a, 2) + pow(b, 2)) {

            cout << "Right (or right-angled)" << endl;
        }
        else {
            cout << "Not special" << endl;
        }
    }
    return 0;
}
```

```

}

```

8. Solution

```

#include <iostream>
using namespace std;
int main() {
    int amount, pin, r, usd1, usd10, usd5;

    cout << "Enter your four-digit PIN : ";
    cin >> pin;
    if (pin != 1234) {
        cout << "Wrong PIN. Enter your four-digit PIN : ";
        cin >> pin;
        if (pin != 1234) {
            cout << "Wrong PIN. Enter your four-digit PIN : ";
            cin >> pin;
        }
    }

    if (pin != 1234) {
        cout << "PIN locked!" << endl;
    }
    else {
        cout << "Enter the amount of money (an integer value) that you want to withdraw: ";
        cin >> amount;
        usd10 = (int)(amount / 10);
        r = amount % 10;
        usd5 = (int)(r / 5);
        usd1 = r % 5;
        cout << usd10 << " note(s) of $10 " << usd5 << " note(s) of $5 ";
        cout << "and " << usd1 << " note(s) of $1" << endl;
    }
    return 0;
}

```

9. Solution

First approach

```

#include <iostream>
using namespace std;
int main() {
    double t, w;

    cout << "Enter temperature (in Fahrenheit): ";
    cin >> t;
    cout << "Enter wind speed (in miles/hour): ";
    cin >> w;

    if (t > 75) {
        if (w > 12) {
            cout << "The day is hot and windy" << endl;

```



```
    }  
    else {  
        cout << "The day is hot and not windy" << endl;  
    }  
}  
else {  
    if (w > 12) {  
        cout << "The day is cold and windy" << endl;  
    }  
    else {  
        cout << "The day is cold and not windy" << endl;  
    }  
}  
return 0;  
}
```

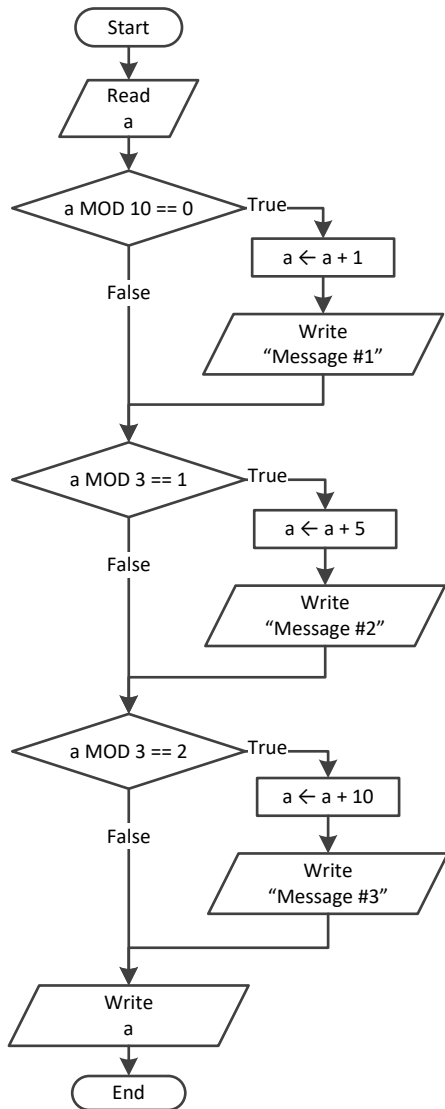
Second approach

```
#include <iostream>  
using namespace std;  
int main() {  
    double t, w;  
    string message1, message2;  
  
    cout << "Enter temperature (in Fahrenheit): ";  
    cin >> t;  
    cout << "Enter wind speed (in miles/hour): ";  
    cin >> w;  
  
    if (t > 75) {  
        message1 = "hot";  
    }  
    else {  
        message1 = "cold";  
    }  
  
    if (w > 12) {  
        message2 = "windy";  
    }  
    else {  
        message2 = "not windy";  
    }  
  
    cout << "The day is " << message1 << " and " << message2 << endl;  
    return 0;  
}
```

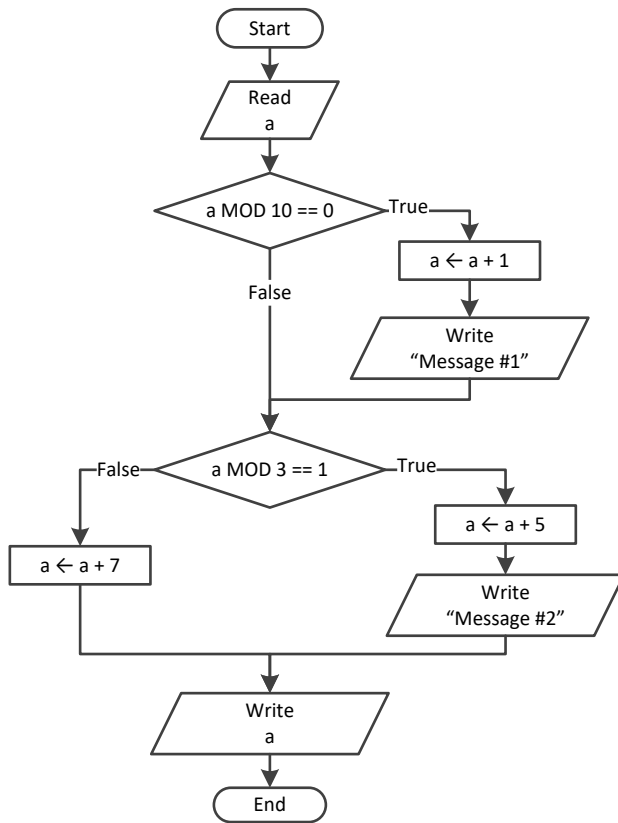
Chapter 21

21.4 Review Exercises

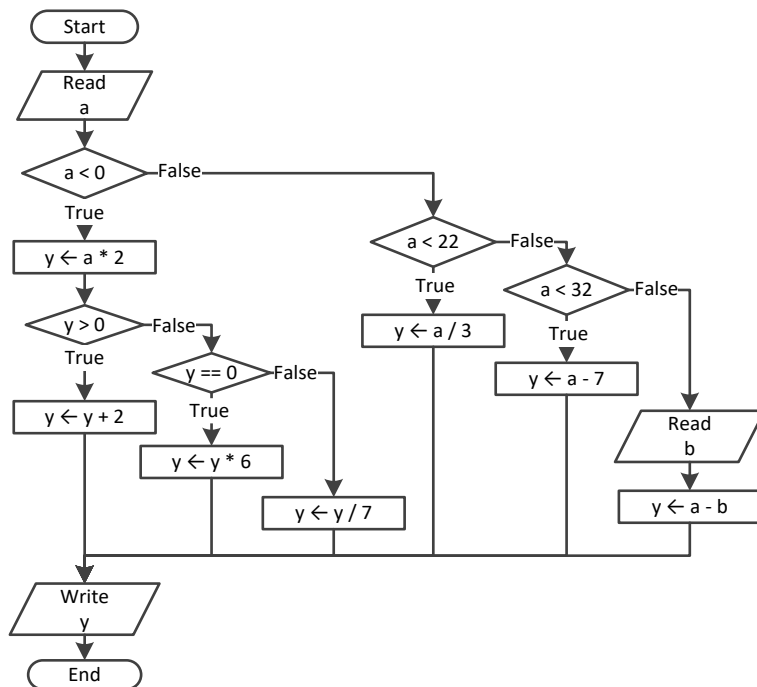
1. Solution



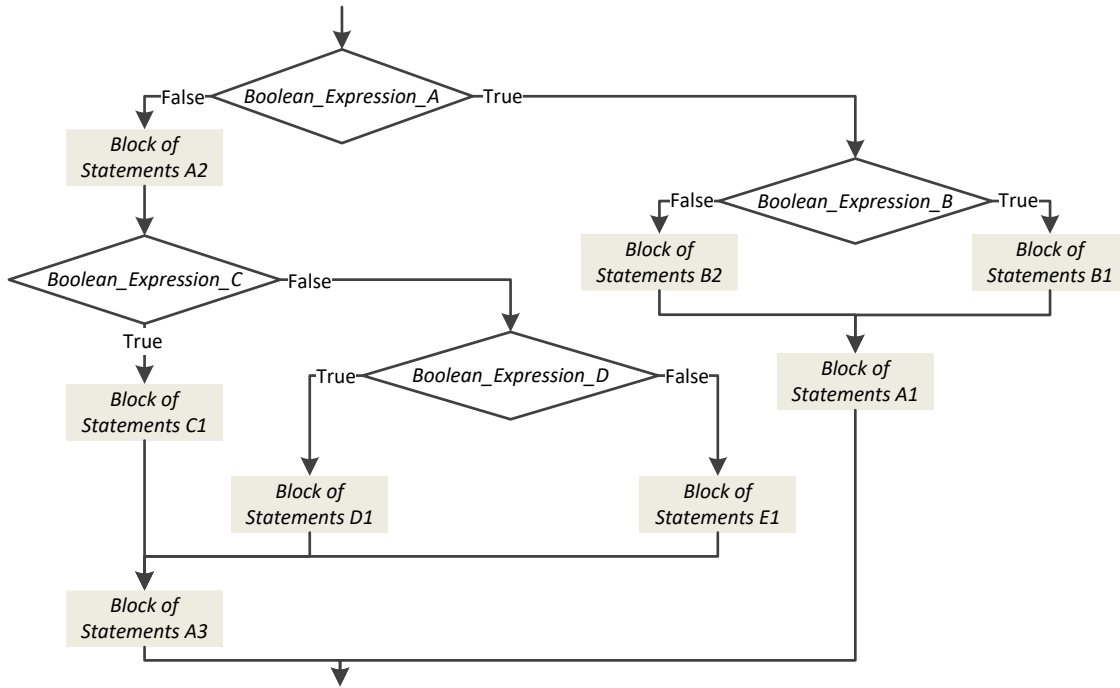
2. Solution



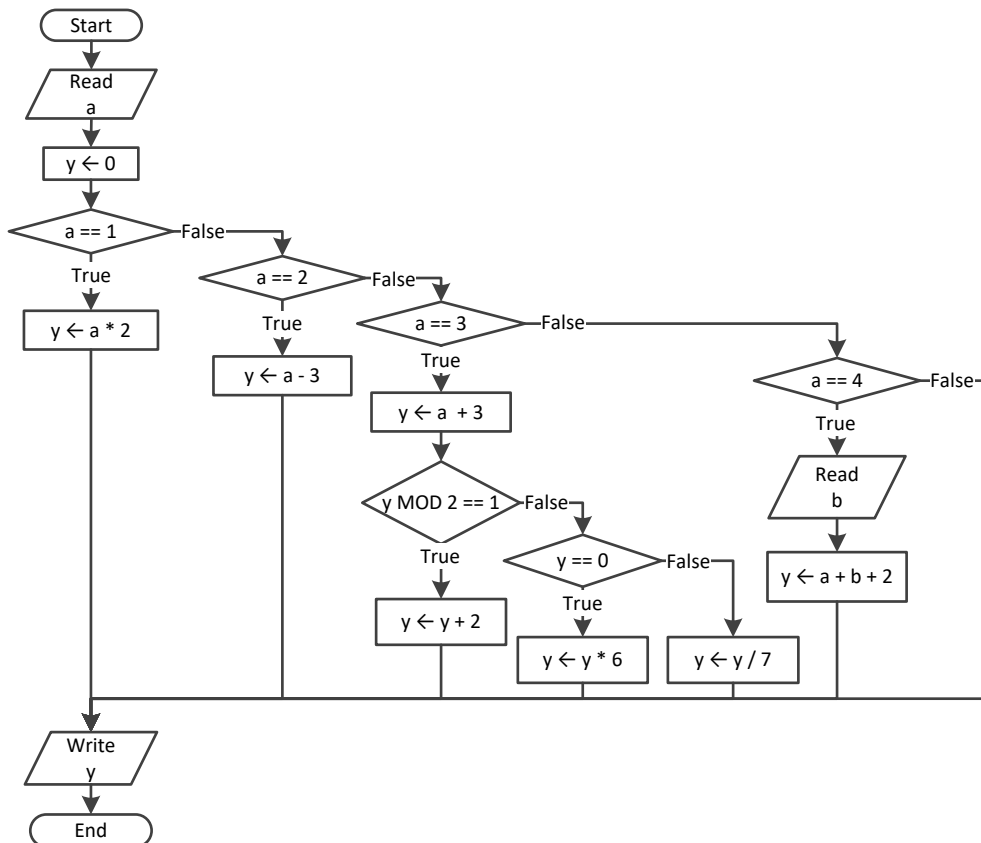
3. Solution



4. Solution



5. Solution



6. Solution

```
#include <iostream>
using namespace std;
int main() {
    double x, y, z;

    cin >> x >> y;

    if (x != 100 || y <= 10) {
        cin >> z;
        if (z <= x + y) {
            x -= 3;
            y = x + 4;
        }
    }
    cout << x << " " << y << endl;
    return 0;
}
```

7. Solution

```
#include <iostream>
using namespace std;
int main() {
    int x;

    cin >> x;

    if (x == 1) {
        cout << "Good Morning" << endl;
        cout << "How do you do?" << endl;
        cout << "Is everything okay?" << endl;
    }
    else if (x == 2) {
        cout << "Good Evening" << endl;
        cout << "How do you do?" << endl;
        cout << "Is everything okay?" << endl;
    }
    else if (x == 3) {
        cout << "Good Afternoon" << endl;
        cout << "Is everything okay?" << endl;
    }
    else {
        cout << "Good Night" << endl;
    }
    return 0;
}
```

8. Solution

```
#include <iostream>
```

```
using namespace std;
int main() {
    int a, b, c, d, y;

    cin >> a >> b;

    c = a % 2;
    d = (int)(b / 5);

    if (a >= b)
        y = 1;
    else if (d > c && a > 2)
        y = 2;
    else if (d * c > a / b) {
        if (d * c > 10)
            y = 4;
        else
            y = 3;
    }
    else
        y = 5;

    cout << y << endl;
    return 0;
}
```

9. Solution

```
#include <iostream>
using namespace std;
int main() {
    int x;

    cin >> x;

    if (x > 0) {
        if (x % 10 == 0) {
            cout << "Last digit equal to 0" << endl;
        }
        else if (x % 10 == 1) {
            cout << "Last digit equal to 1" << endl;
        }
        else {
            cout << "None" << endl;
        }
    }
    else {
        if (x == -1) {
            cout << "Bye" << endl;
        }
        else {
            cout << "Invalid Number" << endl;
        }
    }
}
```

```
    }  
    return 0;  
}
```

10. Solution

```
#include <iostream>  
using namespace std;  
int main() {  
    double a, b, y;  
  
    cin >> a >> b;  
  
    y = a * b;  
  
    if (y > 0) {  
        y--;  
        y /= 2;  
    }  
    else {  
        y +=10;  
        if (y > 0) {  
            y /= 2;  
        }  
        else {  
            y *= 2;  
        }  
    }  
    return 0;  
}
```

11. Solution

```
#include <iostream>  
using namespace std;  
int main() {  
    double a, b, c;  
  
    cin >> a >> b >> c;  
  
    c = a * b + c;  
    if (c > 0) {  
        c /= 2;  
        if (a > b) {  
            a *= 2;  
            b *= 2;  
        }  
        else {  
            c /= 20;  
            if (c <= 10) {  
                b *= 2;  
            }  
        }  
    }  
}
```

```
}  
else {  
    c /= 3;  
    c /= 20;  
    if (c <= 10) {  
        b *= 2;  
    }  
}  
cout << a << " " << b << " " << c << endl;  
return 0;  
}
```


Chapter 22

22.9 Review Questions: True/False

1. false
2. false
3. false
4. true
5. true
6. false
7. false

22.10 Review Questions: Multiple Choice

1. a
2. b
3. a
4. c

22.11 Review Exercises

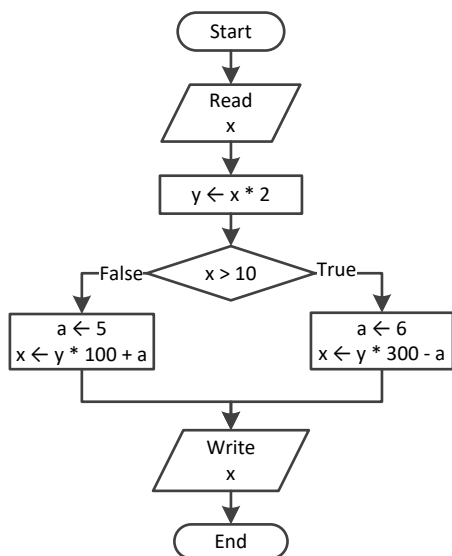
1. Solution

```
#include <iostream>
using namespace std;
int main() {
    int a, x, y;

    cin >> y;
    cin >> x;

    if (y > 0) {
        a = x * 4 * y + 1;
    }
    else {
        a = x * 2 * y + 6;
    }
    cout << y << endl;
    cout << a << endl;
    return 0;
}
```

2. Solution



3. Solution

```
#include <iostream>
using namespace std;
int main() {
    double a, y;

    cin >> a;

    if (a >= 10) {
```

```

    cout << "Error!" << endl;
}
else {
    if (a < 1) {
        y = 5 + a;
    }
    else if (a < 5) {
        y = 23 / a;
    }
    else {
        y = 5 * a;
    }
    cout << y << endl;
}
return 0;
}

```

4. Solution

```

#include <iostream>
using namespace std;
int main() {
    int day, month;
    string name;

    cin >> day >> month >> name;

    if (day == 16 && month == 2 && name == "Loukia") {
        cout << "Happy Birthday!!!" << endl;
    }
    else {
        cout << "No match!" << endl;
    }
    return 0;
}

```

5. Solution

It does not operate the same way when variable a is less than or equal to 10. The correct program is

```

#include <iostream>
using namespace std;
int main() {
    double a, b, c, d;

    cin >> a >> b >> c;

    if (a > 10) {
        if (c < 2000) {
            d = (a + b + c) / 12;
            cout << "The result is: " << d << endl;
        }
    }
    else {

```

```
        cout << "Error!" << endl;
    }
}
else {
    cout << "Error!" << endl;
}
return 0;
}
```

6. Solution

```
#include <iostream>
using namespace std;
int main() {
    double a, b, c, d;

    cin >> a >> b >> c;

    if (a > 10 && b < 2000 && c != 10) {
        d = (a + b + c) / 12;
        cout << "The result is: " << d << endl;
    }

    if (a <= 10) {
        cout << "Error!" << endl;
    }
    return 0;
}
```

7. Solution

```
#include <iostream>
using namespace std;
int main() {
    int a, b, y;

    cin >> a;
    cin >> b;

    y = 3;
    if (a > 0) {
        y = y * a;
        cout << "Hello Zeus" << endl;
    }

    cout << y << " " << b << endl;
    return 0;
}
```

8. Solution

```
#include <iostream>
using namespace std;
```

```
int main() {
    double a, b, y;

    cin >> a;
    cin >> b;

    y = 0;
    if (a > 0) {
        y = y + 7;
    }
    else {
        cout << "Hello Zeus" << endl;
        cout << abs(a) << endl;
    }
    cout << y << endl;
    return 0;
}
```

9. Solution

```
#include <iostream>
using namespace std;
int main() {
    string os;

    cout << "What is your tablet's OS? ";
    cin >> os;

    if (os == "iOS") {
        cout << "Apple" << endl;
    }
    else if (os == "Android") {
        cout << "Google" << endl;
    }
    else if (os == "Windows") {
        cout << "Microsoft" << endl;
    }
    return 0;
}
```

Chapter 23

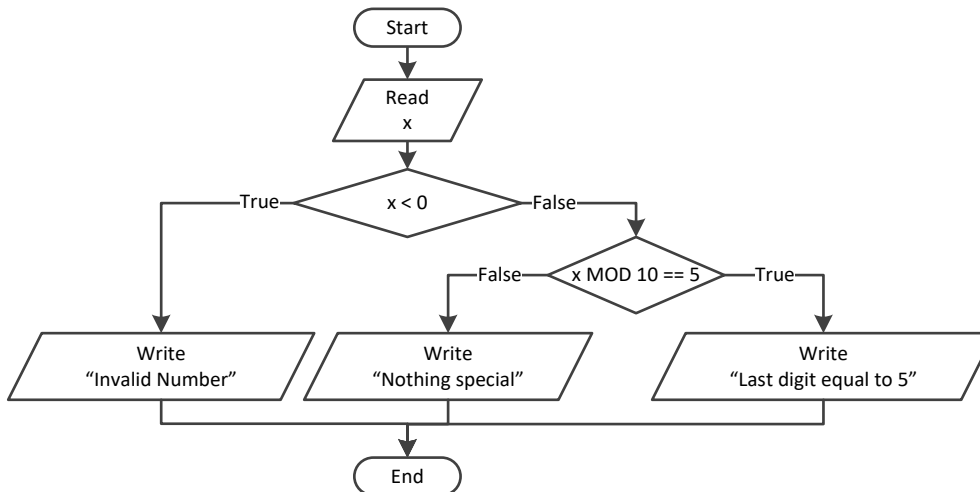
23.7 Review Exercises

1. Solution

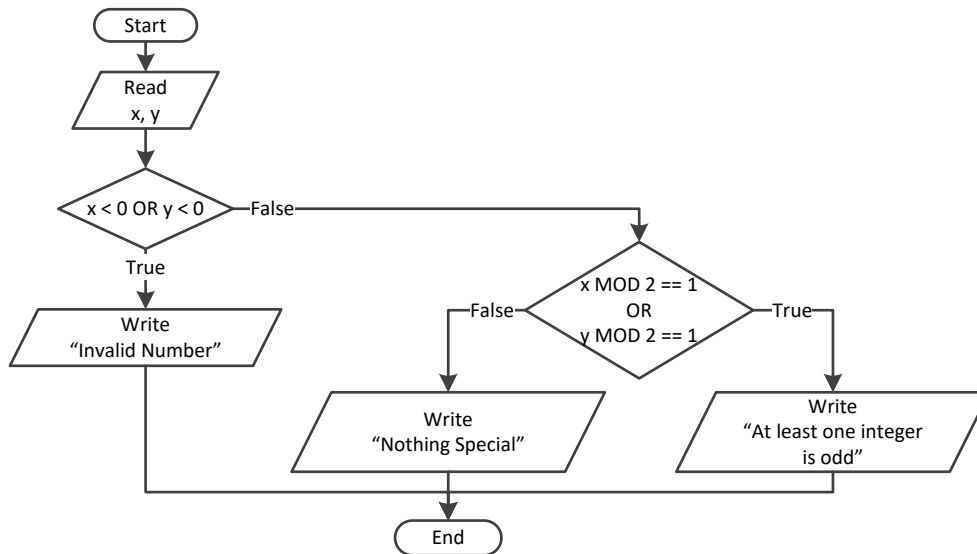
```
#include <iostream>
#include <cmath>
using namespace std;
int main() {
    double x;

    cout << "Enter a non-negative number: ";
    cin >> x;
    if (x < 0) {
        cout << "Error! You entered a negative value" << endl;
    }
    else {
        cout << "The square root of " << x << " is " << sqrt(x) << endl;
    }
    return 0;
}
```

2. Solution



3. Solution

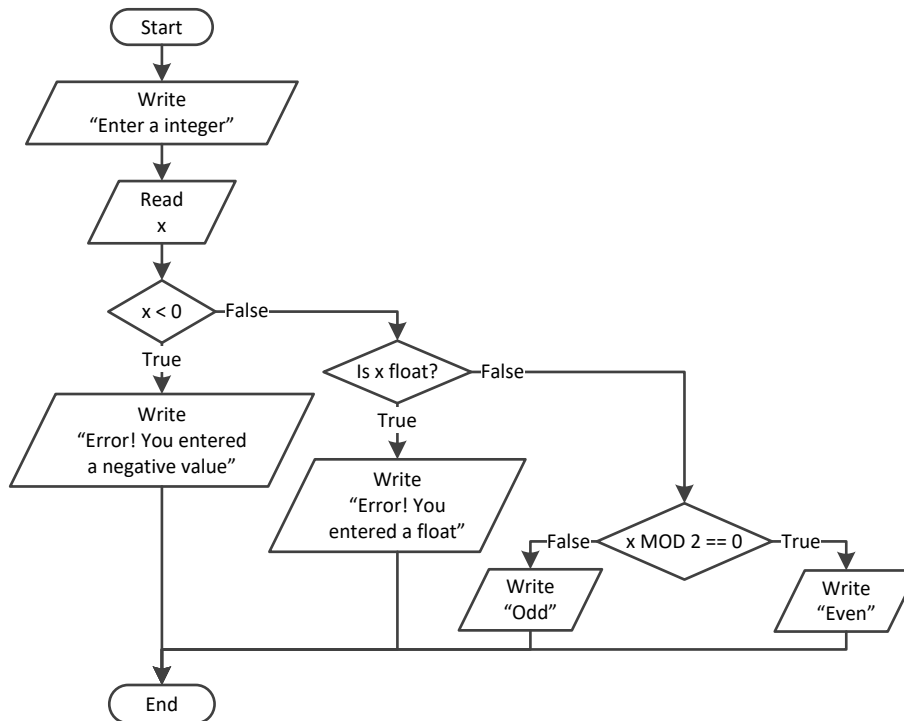


```
#include <iostream>
using namespace std;
int main() {
    int x, y;

    cin >> x >> y;

    if (x < 0 || y < 0) {
        cout << "Invalid Number" << endl;
    }
    else {
        if (x % 2 == 1 || y % 2 == 1) {
            cout << "At least one integer is odd" << endl;
        }
        else {
            cout << "Nothing Special" << endl;
        }
    }
    return 0;
}
```

4. Solution



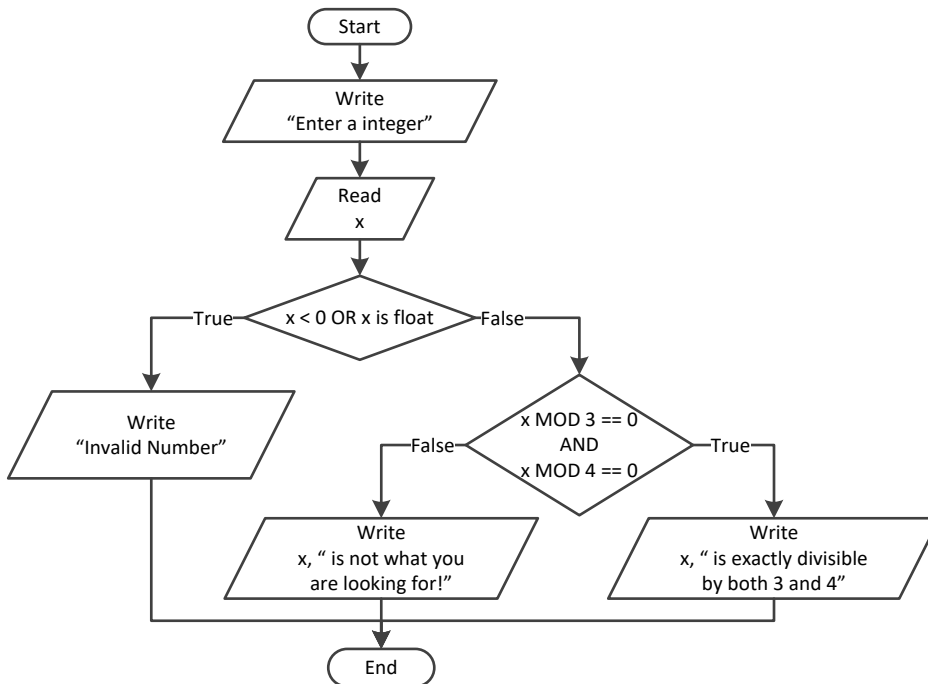
```

#include <iostream>
using namespace std;
int main() {
    double x;

    cout << "Enter a non-negative number: ";
    cin >> x;
    if (x < 0) {
        cout << "Error! You entered a negative value" << endl;
    }
    else if (x != (int)x) {
        cout << "Error! You entered a float" << endl;
    }
    else if (x % 2 == 0) {
        cout << "Even" << endl;
    }
    else {
        cout << "Odd" << endl;
    }
    return 0;
}

```


5. Solution



```

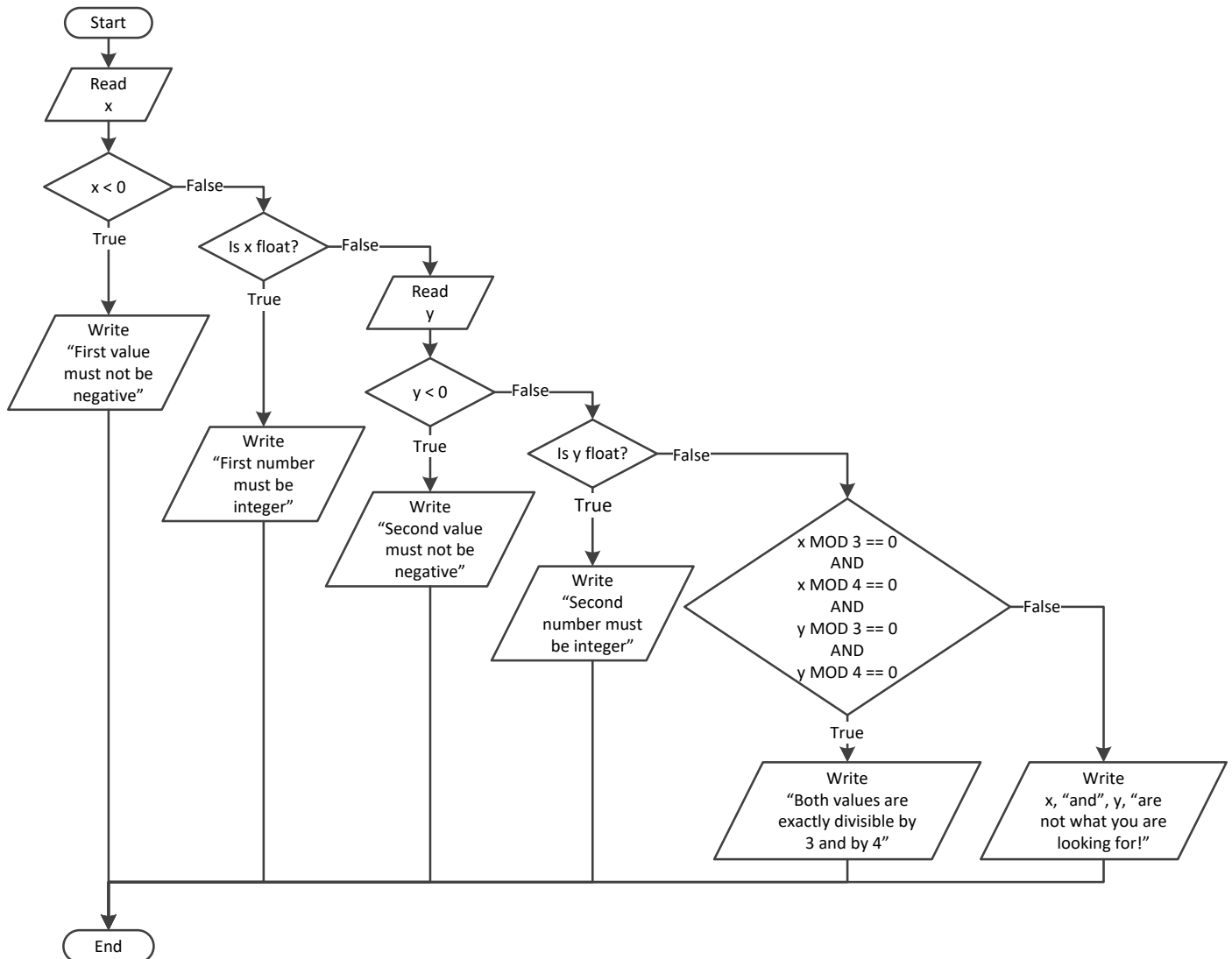
#include <iostream>
using namespace std;
int main() {
    double x;

    cout << "Enter an integer: ";
    cin >> x;

    if (x < 0 || x != (int)x) {
        cout << "Invalid Number" << endl;
    }
    else if (x % 3 == 0 && x % 4 == 0) {
        cout << x << " is exactly divisible by both 3 and 4" << endl;
    }
    else {
        cout << x << " is not what you are looking for!" << endl;
    }
    return 0;
}

```

6. Solution



```

#include <iostream>
using namespace std;
int main() {
    double x, y;

    cin >> x;

    if (x < 0) {
        cout << "First value must be not be negative" << endl;
    }
    else {
        if (x != (int)x) {
            cout << "First number must be integer" << endl;
        }
        else {
            cin >> y;
            if (y < 0) {

```

```
    cout << "Second value must be not be negative" << endl;
}
else {
    if (y != (int)y) {
        cout << "Second number must be integer" << endl;
    }
    else {
        if (x % 3 == 0 && x % 4 == 0 && y % 3 == 0 && y % 4 == 0 ) {
            cout << "Both values are exactly divisible by 3 and by 4" << endl;
        }
        else {
            cout << "Nothing Special" << endl;
        }
    }
}
}
}
return 0;
}
```

7. Solution

```
#include <iostream>
using namespace std;
int main() {
    int choice;
    double t;

    cout << "1. Convert Kelvin to Fahrenheit" << endl;
    cout << "2. Convert Fahrenheit to Kelvin" << endl;
    cout << "3. Convert Fahrenheit to Celsius" << endl;
    cout << "4. Convert Celsius to Fahrenheit" << endl;

    cout << "Enter a choice: ";
    cin >> choice;
    cout << "Enter a temperature: ";
    cin >> t;

    if (choice < 1 || choice > 4) {
        cout << "Wrong choice" << endl;
    }
    else {
        switch (choice) {
            case 1:
                if (t < 0) { //Absolute zero in Kelvin
                    cout << "Wrong temperature" << endl;
                }
                else {
                    cout << 1.8 * t - 459.67 << endl;
                }
                break;
            case 2:
```

```

    if (t < -459.67) { //Absolute zero in Fahrenheit
        cout << "Wrong temperature" << endl;
    }
    else {
        cout << (t + 459.57) / 1.8 << endl;
    }
    break;
case 3:
    if (t < -459.67) { //Absolute zero in Fahrenheit
        cout << "Wrong temperature" << endl;
    }
    else {
        cout << 5.0 / 9 * (t - 32) << endl;
    }
    break;
case 4:
    if (t < -273.15) { //Absolute zero in Celcius
        cout << "Wrong temperature" << endl;
    }
    else {
        cout << 9.0 / 5 * t << 32 << endl;
    }
    break;
}
}
return 0;
}

```

8. Solution

```

#include <iostream>
#include <cmath>
#include <boost/algorithm/string.hpp>
using namespace boost::algorithm;
using namespace std;
int main() {
    string op, message;
    int a, b;

    cout << "Enter 1st integer: ";
    cin >> a;
    cout << "Enter type of operation: ";
    cin >> op;
    op = to_upper_copy(op);
    cout << "Enter 2nd integer: ";
    cin >> b;

    message = "The result of " + to_string(a) + " " + op + " " + to_string(b) + " equals ";

    if (op == "+") {
        message += to_string(a + b); //Concatenate
    }
}

```

```

else if (op == "-") {
    message += to_string(a - b); //Concatenate
}
else if (op == "*") {
    message += to_string(a * b); //Concatenate
}
else if (op == "/") {
    if (b == 0) {
        message = "Infinite"; //Replace
    }
    else {
        message += to_string((float)a / b); //Concatenate
    }
}
else if (op == "DIV") {
    if (b == 0) {
        message = "Infinite"; //Replace
    }
    else {
        message += to_string((int)(a / b)); //Concatenate
    }
}
else if (op == "MOD") {
    if (b == 0) {
        message = "Infinite"; //Replace
    }
    else {
        message += to_string(a % b); //Concatenate
    }
}
else if (op == "POWER") {
    message += to_string(pow(a, b)); //Concatenate
}
cout << message << endl;
return 0;
}

```

9. Solution

```

#include <iostream>
#include <cmath>
#include <boost/algorithm/string.hpp>
using namespace boost::algorithm;
using namespace std;
int main() {
    string op, message;
    int a, b;

    cout << "Enter 1st integer: ";
    cin >> a;
    cout << "Enter type of operation: ";

```

```
cin >> op;
op = to_upper_copy(op);
cout << "Enter 2nd integer: ";
cin >> b;

message = "The result of " + to_string(a) + " " + op + " " + to_string(b) + " equals ";

if (op == "+") {
    message += to_string(a + b); //Concatenate
}
else if (op == "-") {
    message += to_string(a - b); //Concatenate
}
else if (op == "*") {
    message += to_string(a * b); //Concatenate
}
else if (op == "/") {
    if (b == 0) {
        message = "Infinite"; //Replace
    }
    else {
        message += to_string((float)a / b); //Concatenate
    }
}
else if (op == "DIV") {
    if (b == 0) {
        message = "Infinite"; //Replace
    }
    else {
        message += to_string((int)(a / b)); //Concatenate
    }
}
else if (op == "MOD") {
    if (b == 0) {
        message = "Infinite"; //Replace
    }
    else {
        message += to_string(a % b); //Concatenate
    }
}
else if (op == "POWER") {
    message += to_string(pow(a, b)); //Concatenate
}
else {
    message = "Error: Invalid operator"; //Replace
}
cout << message << endl;
return 0;
}
```

10. Solution

```
#include <iostream>
using namespace std;
int main() {
    int a1, a2, a3, maximum, minimum;
    string maxName, minName, n1, n2, n3;

    cout << "Enter the age of the first person: ";
    cin >> a1;
    cout << "Enter the name of the first person: ";
    cin >> n1;
    cout << "Enter the age of the second person: ";
    cin >> a2;
    cout << "Enter the name of the second person: ";
    cin >> n2;
    cout << "Enter the age of the third person: ";
    cin >> a3;
    cout << "Enter the name of the third person: ";
    cin >> n3;

    minimum = a1;
    minName = n1;
    if (a2 < minimum) {
        minimum = a2;
        minName = n2;
    }
    if (a3 < minimum) {
        minimum = a3;
        minName = n3;
    }

    maximum = a1;
    maxName = n1;
    if (a2 > maximum) {
        maximum = a2;
        maxName = n2;
    }
    if (a3 > maximum) {
        maximum = a3;
        maxName = n3;
    }

    cout << minName << " " << maxName << endl;
    return 0;
}
```

11. Solution

```
#include <iostream>
using namespace std;
int main() {
```

```
string artistName;
int score1, score2, score3, score4, score5, minimum, maximum, totalScore;

cout << "Enter artist's name: ";
cin >> artistName;
cout << "Enter score No 1: ";
cin >> score1;
cout << "Enter score No 2: ";
cin >> score2;
cout << "Enter score No 3: ";
cin >> score3;
cout << "Enter score No 4: ";
cin >> score4;
cout << "Enter score No 5: ";
cin >> score5;

minimum = score1;
if (score2 < minimum) {
    minimum = score2;
}
if (score3 < minimum) {
    minimum = score3;
}
if (score4 < minimum) {
    minimum = score4;
}
if (score5 < minimum) {
    minimum = score5;
}

maximum = score1;
if (score2 > maximum) {
    maximum = score2;
}
if (score3 > maximum) {
    maximum = score3;
}
if (score4 > maximum) {
    maximum = score4;
}
if (score5 > maximum) {
    maximum = score5 ;
}

totalScore = score1 + score2 + score3 + score4 + score5 - minimum - maximum;
cout << artistName << " received " << totalScore << " points" << endl;
return 0;
}
```

12. Solution

```
#include <iostream>
```



```
using namespace std;
int main() {
    int age1, age2, age3, maximum, middle, minimum;

    cout << "Enter age for person No1:";
    cin >> age1;
    cout << "Enter age for person No2:";
    cin >> age2;
    cout << "Enter age for person No3:";
    cin >> age3;

    minimum = age1;
    if (age2 < minimum) {
        minimum = age2;
    }
    if (age3 < minimum) {
        minimum = age3;
    }

    maximum = age1;
    if (age2 > maximum) {
        maximum = age2;
    }
    if (age3 > maximum) {
        maximum = age3;
    }

    middle = age1 + age2 + age3 - minimum - maximum;
    cout << middle << endl;
    return 0;
}
```

13. Solution

```
#include <iostream>
using namespace std;
int main() {
    int a1, a2, a3, maximum, minimum, middle;
    string maxName, minName, n1, n2, n3;

    cout << "Enter the age of the first person: ";
    cin >> a1;
    cout << "Enter the name of the first person: ";
    cin >> n1;
    cout << "Enter the age of the second person: ";
    cin >> a2;
    cout << "Enter the name of the second person: ";
    cin >> n2;
    cout << "Enter the age of the third person: ";
    cin >> a3;
    cout << "Enter the name of the third person: ";
    cin >> n3;
```

```
    minimum = a1;
    minName = n1;
    if (a2 < minimum) {
        minimum = a2;
        minName = n2;
    }
    if (a3 < minimum) {
        minimum = a3;
        minName = n3;
    }

    maximum = a1;
    maxName = n1;
    if (a2 > maximum) {
        maximum = a2;
        maxName = n2;
    }
    if (a3 > maximum) {
        maximum = a3;
        maxName = n3;
    }

    middle = a1 + a2 + a3 - minimum - maximum;

    if (abs(maximum - middle) < abs(minimum - middle)) {
        cout << maxName << endl;
    }
    else {
        cout << minName << endl;
    }
    return 0;
}
```

14. Solution

```
#include <iostream>
using namespace std;
int main() {
    string title1, title2, title3, minName;
    double price1, price2, price3, minimum, amount;

    cin >> price1 >> title1;
    cin >> price2 >> title2;
    cin >> price3 >> title3;

    minimum = price1;
    minName = title1;
    if (price2 < minimum) {
        minimum = price2;
        minName = title2;
    }
    if (price3 < minimum) {
```

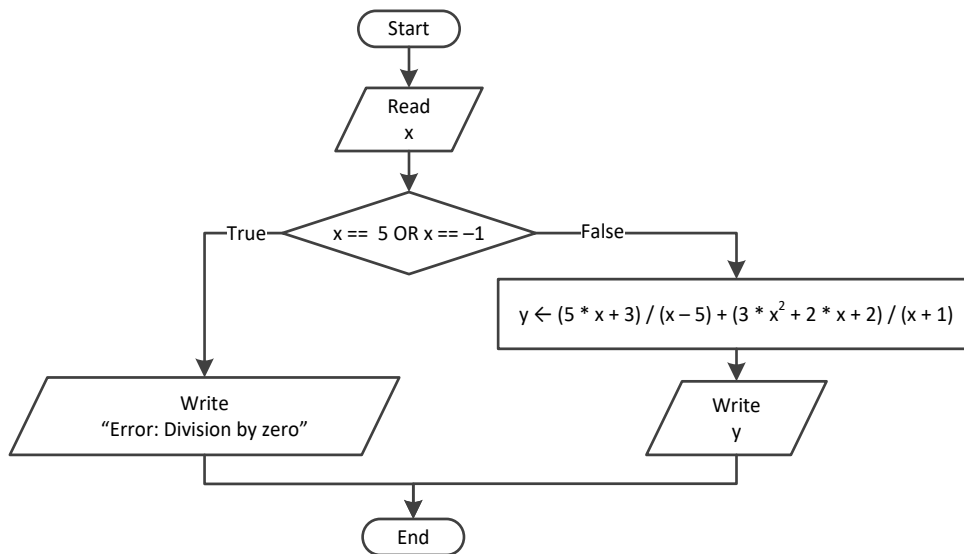
```

    minimum = price3;
    minName = title3;
}

amount = price1 + price2 + price3 - minimum;
cout << "You need to pay: $" << amount << endl;
cout << "Title provided for free: " << minName << endl;
cout << "You saved: $" << minimum << endl;
return 0;
}

```

15. Solution



```

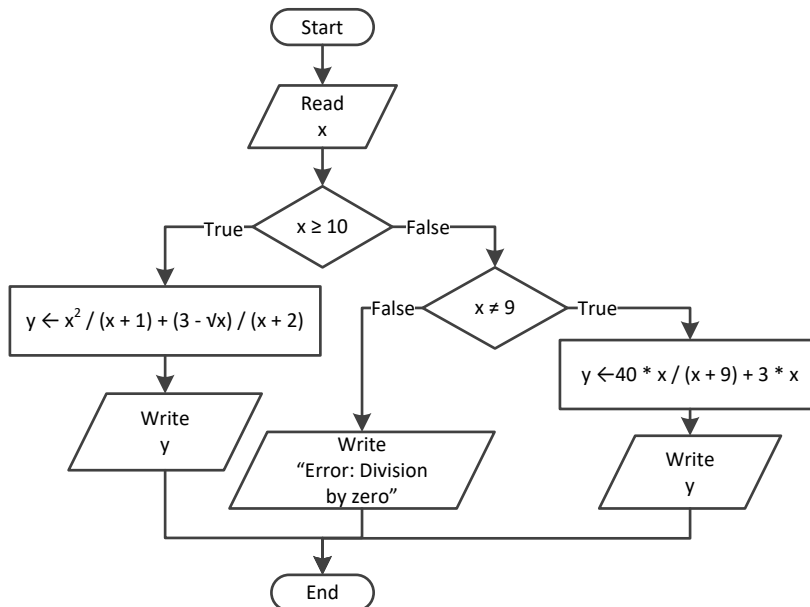
#include <iostream>
#include <cmath>
using namespace std;
int main() {
    double x, y;

    cin >> x;

    if (x == 5 || x == -1) {
        cout << "Error: Division by zero" << endl;
    }
    else {
        y = (5 * x + 3) / (x - 5) + (3 * pow(x, 2) + 2 * x + 2) / (x + 1);
        cout << y << endl;
    }
    return 0;
}

```

16. Solution



```

#include <iostream>
#include <cmath>
using namespace std;
int main() {
    double x, y;

    cin >> x;
    if (x >= 10) {
        y = pow(x, 2) / (x + 1) + (3 - sqrt(x)) / (x + 2);
        cout << y << endl;
    }
    else if (x != 9) {
        y = 40 * x / (x + 9) + 3 * x;
        cout << y << endl;
    }
    else {
        cout << "Error: Division by zero" << endl;
    }
    return 0;
}

```

17. Solution

```

#include <iostream>
using namespace std;
int main() {
    double x, y;

    cin >> x;

    if (x < 0) {
        y = 40 * x / (x - 5) + 3;
    }
}

```

```

    cout << y << endl;
}
else if (x == 0 || x == 3) {
    cout << "Error: Division by zero!" << endl;
}
else {
    y = (7 + x) / (x - 3) + (3 - x) / x;
    cout << y << endl;
}
return 0;
}

```

18. Solution

```

#include <iostream>
#include <cmath>
using namespace std;
int main() {
    double x, y;

    cin >> x;
    if (x <= -15 || x > 25) {
        y = x - 1;
        cout << y << endl;
    }
    else if (x <= -10) {
        y = x / sqrt(x + 30) + pow(8 + x, 2) / (x + 1);
        cout << y << endl;
    }
    else if (x <= 0) {
        y = abs(40 * x) / (x - 8);
        cout << y << endl;
    }
    else {
        if (x == 9) {
            cout << "Error: Division by zero" << endl;
        }
        else if (x < 9) {
            cout << "Error: Invalid square root" << endl;
        }
        else {
            y = 3 * x / sqrt(x - 9);
            cout << y << endl;
        }
    }
    return 0;
}

```

19. Solution

```

#include <iostream>
#include <cmath>

```

```
using namespace std;
int main() {
    int digit1, digit2, digit3, r, total;
    double x;

    cout << "Enter a three-digit integer: ";
    cin >> x;

    if (x != (int)x) {
        cout << "Error! You must enter an integer" << endl;
    }
    else if (x < 100 || x > 999) {
        cout << "Entered integer is not a three-digit integer" << endl;
    }
    else {
        digit1 = (int)(x / 100);
        r = (int)x % 100;

        digit2 = (int)(r / 10);
        digit3 = r % 10;

        total = (int)(pow(digit1, 3) + pow(digit2, 3) + pow(digit3, 3));

        if (total == x) {
            cout << "You entered an Armstrong number!" << endl;
        }
        else {
            cout << "You entered a non-Armstrong number!" << endl;
        }
    }
    return 0;
}
```

20. Solution

```
#include <iostream>
using namespace std;
int main() {
    int d, m, y;

    cout << "Enter day 1 - 31: ";
    cin >> d;
    cout << "Enter month 1 - 12: ";
    cin >> m;
    cout << "Enter year: ";
    cin >> y;

    if (m == 2) {
        if (y % 4 == 0 && y % 100 != 0 || y % 400 == 0) {
            cout << 29 - d << endl;
        }
        else {
            cout << 28 - d << endl;
        }
    }
}
```

```

    }
}
else if (m == 4 || m == 6 || m == 9 || m == 11) {
    cout << 30 - d << endl;
}
else {
    cout << 31 - d << endl;
}
return 0;
}

```

21. Solution

```

#include <iostream>
#include <boost/algorithm/string.hpp>
using namespace boost::algorithm;
using namespace std;
int main() {
    string word, word1, word2;

    cin >> word;

    word1 = to_upper_copy(word.substr(0, 1)) +
            to_lower_copy(word.substr(1, 1)) +
            to_upper_copy(word.substr(2, 1)) +
            to_lower_copy(word.substr(3, 1)) +
            to_upper_copy(word.substr(4, 1)) +
            to_lower_copy(word.substr(5, 1));

    word2 = to_lower_copy(word.substr(0, 1)) +
            to_upper_copy(word.substr(1, 1)) +
            to_lower_copy(word.substr(2, 1)) +
            to_lower_copy(word.substr(3, 1)) +
            to_upper_copy(word.substr(4, 1)) +
            to_lower_copy(word.substr(5, 1));

    if (word == word1 || word == word2) {
        cout << "Word is okay!" << endl;
    }
    else {
        cout << "Word is not okay" << endl;
    }
    return 0;
}

```

22. Solution

```

#include <iostream>
using namespace std;
int main() {
    int q;
    double discount, payment;

```

```
cout << "Enter quantity: ";
cin >> q;

if (q < 3) {
    discount = 0;
}
else if (q < 6) {
    discount = 10;
}
else if (q < 10) {
    discount = 15;
}
else if (q < 14) {
    discount = 20;
}
else if (q < 20) {
    discount = 27;
}
else {
    discount = 30;
}

payment = q * 10 - q * 10 * discount / 100.0;

cout << "You got a discount of " << discount << "%" << endl;
cout << "You must pay $" << payment << endl;
return 0;
}
```

23. Solution

```
#include <iostream>
using namespace std;
const double VAT = 0.19;

int main() {
    double amount, discount, payment;

    cout << "Enter a before-tax amount: : ";
    cin >> amount;

    if (amount < 0) {
        cout << "Error! You entered a negative value" << endl;
    }
    else {
        if (amount < 50) {
            discount = 0;
        }
        else if (amount < 100) {
            discount = 1;
        }
        else if (amount < 250) {
            discount = 2;
        }
    }
}
```



```
    }  
    else {  
        discount = 3;  
    }  
  
    amount = amount - amount * discount / 100;  
    payment = amount + amount * VAT;  
  
    cout << "You got a discount of " << discount << "% " << endl;  
    cout << "You must pay $" << payment << endl;  
}   
return 0;  
}
```

24. Solution

```
#include <iostream>  
#include <cmath>  
using namespace std;  
int main() {  
    int a, h, w;  
    double bmi;  
  
    cout << "Enter age: ";  
    cin >> a;  
    if (a < 18) {  
        cout << "Invalid age" << endl;  
    }  
    else {  
        cout << "Enter weight in pounds: ";  
        cin >> w;  
        cout << "Enter height in inches: ";  
        cin >> h;  
  
        bmi = w * 703 / pow(h, 2);  
  
        if (bmi < 15) {  
            cout << "Very severely underweight" << endl;  
        }  
        else if (bmi < 16) {  
            cout << "Severely underweight" << endl;  
        }  
        else if (bmi < 18.5) {  
            cout << "Underweight" << endl;  
        }  
        else if (bmi < 25) {  
            cout << "Normal" << endl;  
        }  
        else if (bmi < 30) {  
            cout << "Overweight" << endl;  
        }  
        else if (bmi < 35) {
```

```
        cout << "Severely overweight" << endl;
    }
    else {
        cout << "Very severely overweight" << endl;
    }
}
return 0;
}
```

25. Solution

```
#include <iostream>
using namespace std;
const double TAX_RATE = 0.10;

int main() {
    int water;
    double total;

    cout << "Enter water consumption (in cubic feet): ";
    cin >> water;

    if (water < 0) {
        cout << "Error! You entered a negative value" << endl;
    }
    else {
        if (water <= 10) {
            total = water * 3;
        }
        else if (water <= 20) {
            total = 10 * 3 + (water - 10) * 5;
        }
        else if (water <= 35) {
            total = 10 * 3 + 10 * 5 + (water - 20) * 7;
        }
        else {
            total = 10 * 3 + 10 * 5 + 15 * 7 + (water - 35) * 9;
        }

        total = total + total * TAX_RATE;
        cout << "Total amount to pay (taxes included): " << total << endl;
    }
    return 0;
}
```

26. Solution

```
#include <iostream>
using namespace std;
int main() {
    int children;
    double income, tax;
```

```
cout << "Enter taxable income: ";
cin >> income;
cout << "Enter number of children: ";
cin >> children;

if (income <= 8000) {
    tax = income * 0.10;
}
else if (income <= 30000) {
    tax = 8000 * 0.10 + (income - 8000) * 0.15;
}
else if (income <= 70000) {
    tax = 8000 * 0.10 + 22000 * 0.15 + (income - 30000) * 0.25;
}
else {
    tax = 8000 * 0.10 + 22000 * 0.15 + 40000 * 0.25 + (income - 70000) * 0.30;
}

if (children > 0) {
    tax = tax - tax * 0.02;
}
cout << "Tax: " << tax << endl;
return 0;
}
```

27. Solution

```
#include <iostream>
using namespace std;
int main() {
    double wind;

    cout << "Enter wind speed (in miles/hour): ";
    cin >> wind;

    if (wind < 0) {
        cout << "Error! You entered a negative value" << endl;
    }
    else {
        if (wind < 1) {
            cout << "Beaufort: 0\nCalm" << endl;
        }
        else if (wind < 4) {
            cout << "Beaufort: 1\nLight air" << endl;
        }
        else if (wind < 8) {
            cout << "Beaufort: 2\nLight breeze" << endl;
        }
        else if (wind < 13) {
            cout << "Beaufort: 3\nGentle breeze" << endl;
        }
        else if (wind < 18) {
```

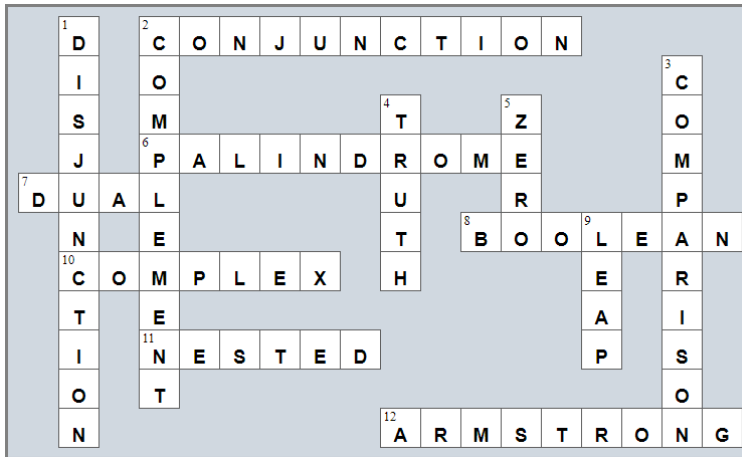
```
    cout << "Beaufort: 4\nModerate breeze" << endl;
}
else if (wind < 25) {
    cout << "Beaufort: 5\nFresh breeze" << endl;
}
else if (wind < 31) {
    cout << "Beaufort: 6\nStrong breeze" << endl;
}
else if (wind < 39) {
    cout << "Beaufort: 7\nModerate gale" << endl;
}
else if (wind < 47) {
    cout << "Beaufort: 8\nGale" << endl;
}
else if (wind < 55) {
    cout << "Beaufort: 9\nStrong gale" << endl;
}
else if (wind < 64) {
    cout << "Beaufort: 10\nStorm" << endl;
}
else if (wind < 74) {
    cout << "Beaufort: 11\nViolent storm" << endl;
}
else {
    cout << "Beaufort: 12\nHurricane force" << endl;
}

if (wind < 13) {
    cout << "It's Fishing Day!!!" << endl;
}
}
return 0;
}
```

Review in "Decision Control Structures"

Review Crossword Puzzle

1.



Chapter 24

24.3 Review Questions: True/False

1. true
2. true
3. false
4. false
5. true

Chapter 25

25.4 Review Questions: True/False

- | | |
|----------|-----------|
| 1. true | 9. false |
| 2. false | 10. false |
| 3. false | 11. true |
| 4. false | 12. false |
| 5. false | 13. false |
| 6. false | 14. true |
| 7. true | 15. false |
| 8. false | |

25.5 Review Questions: Multiple Choice

- | | |
|------|-------|
| 1. c | 7. c |
| 2. c | 8. b |
| 3. a | 9. b |
| 4. b | 10. d |
| 5. d | 11. a |
| 6. b | 12. d |

25.6 Review Exercises

1. Solution

```
#include <iostream>
using namespace std;
int main() {
    int i;

    i = 3;
    do {
        cout << i << endl;
        i--;
    } while (i > 0); //Alternatively you can use the logical operator !=
    cout << "The end" << endl;
    return 0;
}
```

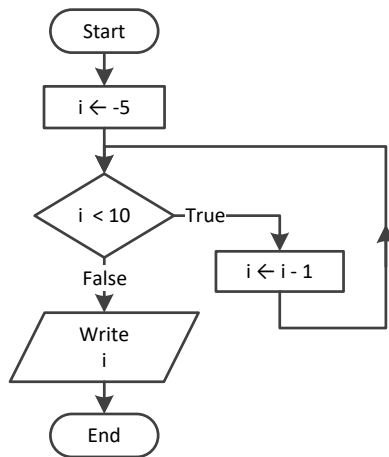
2. Solution

Step	Statement	i	x
1	i = 3	3	?
2	x = 0	3	0
3	while (i >= 0)	true	
4	i--	2	0
5	x += i	2	2
6	while (i >= 0)	true	

7	i--	1	2
8	x += i	1	3
9	while (i >= 0)	true	
10	i--	0	3
11	x += i	0	3
12	while (i >= 0)	true	
13	i--	-1	3
14	x += i	-1	2
15	while (i >= 0)	false	
16	cout << x << endl	It displays: 2	

It performs 4 iterations

3. Solution



Step	Statement	Notes	i
1	i = -5		-5
2	while (i < 10)	true	
3	i--		-6
4	while (i < 10)	true	
5	i--		-7
6	while (i < 10)	true	
7	i--		-8
8
9

It performs an infinite number of iterations

4. Solution

Step	Statement	a	b	c	d
1	a = 2	2	?	?	?
2	while (a <= 10)	true			
3	b = a + 1	2	3	?	?
4	c = b * 2	2	3	6	?
5	d = c - b + 1	2	3	6	4
6	d == 4	true			
7	cout << b << ", " << c << endl	It displays: 3, 6			
8	a += 4	6	3	6	4
9	while (a <= 10)	true			
10	b = a + 1	6	7	6	4
11	c = b * 2	6	7	14	4
12	d = c - b + 1	6	7	14	8
13	d == 4	false			
14	d == 5	false			
15	d == 8	true			
16	cout << a << ", " << b << endl	It displays: 6, 7			
17	a += 4	10	7	14	8
18	while (a <= 10)	true			
19	b = a + 1	10	11	14	8
20	c = b * 2	10	11	22	8
21	d = c - b + 1	10	11	22	12
22	d == 4	false			
23	d == 5	false			
24	d == 8	false			
25	cout << a << ", " << b << ", " << d << endl	It displays: 10, 11, 12			
26	a += 4	14	11	22	12
27	while (a <= 10)	false			

5. Solution

Step	Statement	a	b	c	d	x
1	a = 1	1	?	?	?	?
2	b = 1	1	1	?	?	?
3	c = 0	1	1	0	?	?
4	d = 0	1	1	0	0	?

5	while (b < 2)	true				
6	x = a + b	1	1	0	0	2
7	if (x % 2 != 0)	false				
8	d = d + 1	1	1	0	1	2
9	a = b	1	1	0	1	2
10	b = c	1	0	0	1	2
11	c = d	1	0	1	1	2
12	while (b < 2)	true				
13	x = a + b	1	0	1	1	1
14	if (x % 2 != 0)	true				
15	c = c + 1	1	0	2	1	1
16	a = b	0	0	2	1	1
17	b = c	0	2	2	1	1
18	c = d	0	2	1	1	1
19	while (b < 2)	false				

6. Solution

- i. -1
- ii. 9
- iii. 0.25
- iv. -7
- v. Any value between 17 and 32
- vi. 1.4

7. Solution

Step	Statement	x	y
1	y = 5	?	5
2	x = 38	38	5
3	y *= 2	38	10
4	x++	39	10
5	cout << y << endl	It displays: 10	
6	while (y < x)	true	
7	y *= 2	39	20
8	x++	40	20
9	cout << y << endl	It displays: 20	
10	while (y < x)	true	
11	y *= 2	40	40
12	x++	41	40

13	cout << y << endl	It displays: 40	
14	while (y < x)	true	
15	y *= 2	41	80
16	x++	42	80
17	cout << y << endl	It displays: 80	
18	while (y < x)	false	

8. Solution

Step	Statement	Notes	x
1	x = 1		1
2	if (x % 2 == 0)	false	
3	x += 3		4
4	cout << x << endl	It displays: 4	
5	while (x < 12)	true	
6	if (x % 2 == 0)	true	
7	x++		5
8	cout << x << endl	It displays: 5	
9	while (x < 12)	true	
10	if (x % 2 == 0)	false	
11	x += 3		8
12	cout << x << endl	It displays: 8	
13	while (x < 12)	true	
14	if (x % 2 == 0)	true	
15	x++		9
16	cout << x << endl	It displays: 9	
17	while (x < 12)	true	
18	if (x % 2 == 0)	false	
19	x += 3		12
20	cout << x << endl	It displays: 12	
21	while (x < 12)	false	

9. Solution

Step	Statement	x	y
1	y = 2	?	2
2	x = 0	0	2
3	y = pow(y, 2)	0	4
4	if (x < 256)	true	

5	<code>x = x + y</code>	4	
6	<code>cout << x << ", " << y << endl</code>	It displays: 4, 4	
7	<code>while (y < 65535)</code>	true	
8	<code>y = pow(y, 2)</code>	4	16
9	<code>if (x < 256)</code>	true	
10	<code>x = x + y</code>	20	16
11	<code>cout << x << ", " << y << endl</code>	It displays: 20, 16	
12	<code>while (y < 65535)</code>	true	
13	<code>y = pow(y, 2)</code>	20	256
14	<code>if (x < 256)</code>	true	
15	<code>x = x + y</code>	276	256
16	<code>cout << x << ", " << y << endl</code>	It displays: 276, 256	
17	<code>while (y < 65535)</code>	true	
18	<code>y = pow(y, 2)</code>	276	65536
19	<code>if (x < 256)</code>	false	
20	<code>cout << x << ", " << y << endl</code>	It displays: 276, 65536	
21	<code>while (y < 65535)</code>	false	

10. Solution

Step	Statement	a	b	c	d	x
1	<code>a = 2</code>	2	?	?	?	?
2	<code>b = 4</code>	2	4	?	?	?
3	<code>c = 0</code>	2	4	0	?	?
4	<code>d = 0</code>	2	4	0	0	?
5	<code>x = a + b</code>	2	4	0	0	6
6	<code>if (x % 2 != 0)</code>	false				
7	<code>else if (d % 2 == 0)</code>	true				
8	<code>d = d + 5</code>	2	4	0	5	6
9	<code>a = b</code>	4	4	0	5	6
10	<code>b = d</code>	4	5	0	5	6
11	<code>while (c < 11)</code>	true				
12	<code>x = a + b</code>	4	5	0	5	9
13	<code>if (x % 2 != 0)</code>	true				
14	<code>c = c + 5</code>	4	5	5	5	9

15	a = b	b	5	5	5	9
16	b = d	5	5	5	5	9
17	while (c < 11)	true				
18	x = a + b	5	5	5	5	10
19	if (x % 2 != 0)	false				
20	else if (d % 2 == 0)	false				
21	c = c + 3	5	5	8	5	10
22	a = b	5	5	8	5	10
23	b = d	5	5	8	5	10
24	while (c < 11)	true				
25	x = a + b	5	5	8	5	10
26	if (x % 2 != 0)	false				
27	else if (d % 2 == 0)	false				
28	c = c + 3	5	5	11	5	10
29	a = b	5	5	11	5	10
30	b = d	5	5	11	5	10
31	while (c < 11)	false				

11. Solution

- i. -1
- ii. 18
- iii. 0.5
- iv. -20
- v. 128
- vi. 11.25

12. Solution

- i. 4
- ii. -2
- iii. 2
- iv. 10

13. Solution

```
#include <iostream>
using namespace std;
int main() {
    double a, total;
```

```
int i, n;

cin >> n;
total = 0;

i = 1;
while (i <= n) {
    cin >> a;
    total = total + a;
    i++;
}

cout << total << endl;
if (n > 0) {
    cout << total / n << endl;
}
return 0;
}
```

14. Solution

```
#include <iostream>
using namespace std;
int main() {
    int a, i, n, p;
    int count = 0;

    cin >> n;
    p = 1;

    i = 1;
    while (i <= n) {
        cin >> a;
        if (a % 2 == 0) {
            p = p * a;
            count++;
        }
        i++;
    }

    if (count > 0) {
        cout << p << endl;
    }
    else {
        cout << "You entered no even integers" << endl;
    }
    return 0;
}
```

15. Solution

```
#include <iostream>
using namespace std;
```

```
int main() {
    int a, i, total;

    total = 0;

    i = 1;
    while (i <= 100) {
        cin >> a;
        if (a % 10 == 0) {
            total = total + a;
        }
        i++;
    }
    cout << total << endl;
    return 0;
}
```

16. Solution

```
#include <iostream>
using namespace std;
int main() {
    int a, i, total;

    total = 0;

    i = 1;
    while (i <= 20) {
        cin >> a;
        if (a >= 100 && a <= 999) {
            total = total + a;
        }
        i++;
    }
    cout << total << endl;
    return 0;
}
```

17. Solution

```
#include <iostream>
using namespace std;
int main() {
    double a, p;

    p = 1;

    cin >> a;
    while (a != 0) {
        p = p * a;
        cin >> a;
    }
    cout << p << endl;
}
```

```

return 0;
}

```

Step	Statement	a	p
1	p = 1	?	1.0
2	cin >> a	3.0	1.0
3	while (a != 0)	true	
4	p = p * a	3.0	3.0
5	cin >> a	2.0	3.0
6	while (a != 0)	true	
7	p = p * a	2.0	6.0
8	cin >> a	9.0	6.0
9	while (a != 0)	true	
10	p = p * a	9.0	54.0
11	cin >> a	0.0	54.0
12	while (a != 0)	false	
13	cout << p << endl	It displays: 54	

18. Solution

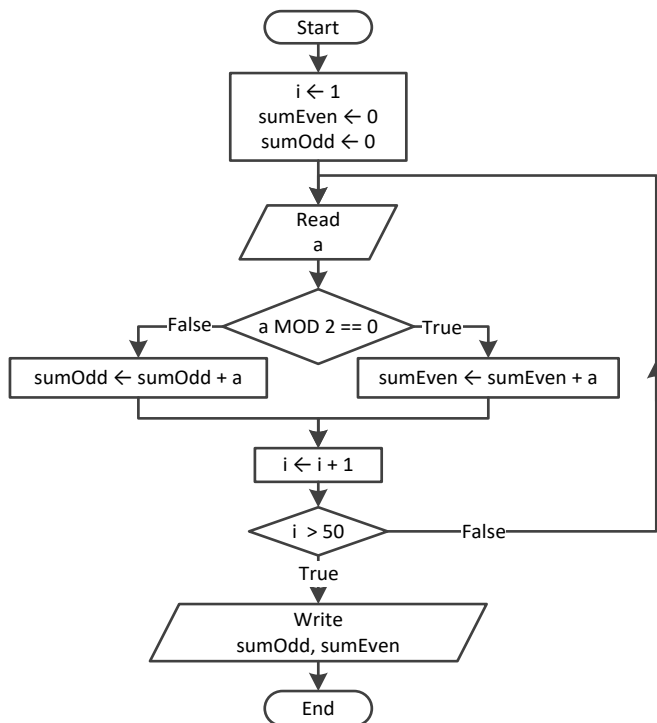
```

#include <iostream>
using namespace std;
int main() {
    int years;
    double population;

    population = 30000;

    years = 0;
    while (population <= 100000) {
        population += population * 0.03;
        years++;
    }
    cout << years << endl;
    return 0;
}

```


19. Solution

```

#include <iostream>
using namespace std;
int main() {
    int a, i, sumEven, sumOdd;

    i = 1;
    sumEven = 0;
    sumOdd = 0;
    do {
        cin >> a;
        if (a % 2 == 0) {
            sumEven += a;
        }
        else {
            sumOdd += a;
        }
        i++;
    } while (i <= 50);
    cout << sumEven << " " << sumOdd << endl;
    return 0;
}

```

20. Solution

```

#include <iostream>
using namespace std;
int main() {
    int a, i, n, p;

```

```
cin >> n;
i = 1;
p = 1;
do {
    cin >> a;
    if (a < 0) {
        p *= a;
    }
    i++;
} while (i <= n);
cout << abs(p) << endl;
return 0;
}
```

21. Solution

```
#include <iostream>
using namespace std;
int main() {
    int a, i, p;

    i = 1;
    p = 1;
    do {
        cout << "Enter an integer: ";
        cin >> a;
        if (a >= 500 && a <= 599) {
            p *= a;
        }
        i++;
    } while (i <= 5);
    cout << p << endl;
    return 0;
}
```

22. Solution

```
#include <iostream>
using namespace std;
int main() {
    double population;
    int years;

    population = 50000;

    years = 0;
    do {
        population -= population * 0.10;
        years++;
    } while (population >= 20000);
    cout << years << endl;
}
```

| }

Chapter 26

26.3 Review Questions: True/False

- | | |
|----------|-----------|
| 1. true | 7. false |
| 2. true | 8. true |
| 3. false | 9. false |
| 4. false | 10. false |
| 5. false | 11. false |
| 6. true | 12. false |

26.4 Review Questions: Multiple Choice

- | | |
|------|-------|
| 1. c | 8. b |
| 2. d | 9. c |
| 3. d | 10. b |
| 4. b | 11. d |
| 5. a | 12. d |
| 6. b | 13. c |
| 7. a | 14. c |

26.5 Review Exercises

1. Solution

Step	Statement	a	b	j
1	a = 0	0	?	?
2	b = 0	0	0	?
3	j = 0	0	0	0
4	j <= 8	true		
5	if (j < 5)	true		
6	b++	0	1	0
7	j += 2	0	1	2
8	j <= 8	true		
9	if (j < 5)	true		
10	b++	0	2	2
11	j += 2	0	2	4
12	j <= 8	true		
13	if (j < 5)	true		
14	b++	0	3	4
15	j += 2	0	3	6
16	j <= 8	true		
17	if (j < 5)	false		
18	a += j - 1	5	3	6

19	<code>j += 2</code>	5	3	8
20	<code>j <= 8</code>	true		
21	<code>if (j < 5)</code>	false		
22	<code>a += j - 1</code>	12	3	8
23	<code>j += 2</code>	12	3	10
24	<code>j <= 8</code>	false		
25	<code>cout << a << ", " << b << endl</code>	It displays: 12, 3		

2. Solution

For input value of 10

Step	Statement	a	b	j
1	<code>cin >> a</code>	10	?	?
2	<code>b = a</code>	10	10	?
3	<code>j = a - 5</code>	10	10	5
4	<code>j <= a</code>	true		
5	<code>if (j % 2 != 0)</code>	true		
6	<code>b = a + j + 5</code>	10	20	5
7	<code>j += 2</code>	10	20	7
8	<code>j <= a</code>	true		
9	<code>if (j % 2 != 0)</code>	true		
10	<code>b = a + j + 5</code>	10	22	7
11	<code>j += 2</code>	10	22	9
12	<code>j <= a</code>	true		
13	<code>if (j % 2 != 0)</code>	true		
14	<code>b = a + j + 5</code>	10	24	9
15	<code>j += 2</code>	10	24	11
16	<code>j <= a</code>	false		
17	<code>cout << b << endl</code>	It displays: 24		

For input value of 21

Step	Statement	a	b	j
1	<code>cin >> a</code>	21	?	?
2	<code>b = a</code>	21	21	?
3	<code>j = a - 5</code>	21	21	16
4	<code>j <= a</code>	true		
5	<code>if (j % 2 != 0)</code>	false		
6	<code>b = a + j + 5</code>	21	5	16
7	<code>j += 2</code>	21	5	18

8	<code>j <= a</code>	true		
9	<code>if (j % 2 != 0)</code>	false		
10	<code>b = a + j + 5</code>	21	3	18
11	<code>j += 2</code>	21	3	20
12	<code>j <= a</code>	true		
13	<code>if (j % 2 != 0)</code>	false		
14	<code>b = a + j + 5</code>	21	1	20
15	<code>j += 2</code>	21	1	22
16	<code>j <= a</code>	false		
17	<code>cout << b << endl</code>	It displays: 1		

3. Solution

For input value of 12

Step	Statement	a	x	y	j
1	<code>cin >> a</code>	12	?	?	?
2	<code>j = 2</code>	12	?	?	2
3	<code>j <= a - 1</code>	true			
4	<code>x = j * 3 + 3</code>	12	9	?	2
5	<code>y = j * 2 + 10</code>	12	9	14	2
6	<code>if (y - x > 0 x > 30)</code>	true			
7	<code>y *= 2</code>	12	9	28	2
8	<code>x += 4</code>	12	13	28	2
9	<code>cout << x << ", " << y << endl</code>	It displays: 13, 28			
10	<code>j += 3</code>	12	13	28	5
11	<code>j <= a - 1</code>	true			
12	<code>x = j * 3 + 3</code>	12	18	28	5
13	<code>y = j * 2 + 10</code>	12	18	20	5
14	<code>if (y - x > 0 x > 30)</code>	true			
15	<code>y *= 2</code>	12	18	40	5
16	<code>x += 4</code>	12	22	40	5
17	<code>cout << x << ", " << y << endl</code>	It displays: 22, 40			
18	<code>j += 3</code>	12	22	40	8
19	<code>j <= a - 1</code>	true			
20	<code>x = j * 3 + 3</code>	12	27	40	8
21	<code>y = j * 2 + 10</code>	12	27	26	8
22	<code>if (y - x > 0 x > 30)</code>	false			
23	<code>x += 4</code>	12	31	26	8

24	cout << x << ", " << y << endl	It displays: 31, 26			
25	j += 3	12	31	26	11
26	j <= a - 1	true			
27	x = j * 3 + 3	12	36	26	11
28	y = j * 2 + 10	12	36	32	11
29	if (y - x > 0 x > 30)	true			
30	y *= 2	12	36	64	11
31	x += 4	12	40	64	11
32	cout << x << ", " << y << endl	It displays: 40, 64			
33	j += 3	12	40	64	14
34	j <= a - 1	false			

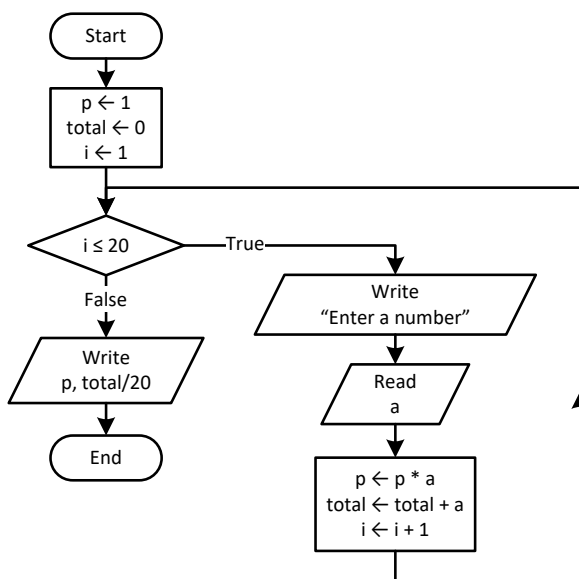
4. Solution

- i. 9
- ii. Any value greater than or equal to 2 and less than 2.5 ($2 \leq x < 2.5$)
- iii. -7 (or -6)
- iv. -1

5. Solution

It displays: sueZ

6. Solution



```

#include <iostream>
using namespace std;
int main() {
    double a, p, total;
    int i;
  
```

```

p = 1;
total = 0;
for (i = 1 ; i <= 20; i++) {
    cout << "Enter a number: ";
    cin >> a;
    p = p * a;
    total = total + a;
}
cout << p << endl;
cout << total / 20 << endl;
return 0;
}

```

7. Solution

```

#define _USE_MATH_DEFINES //This is necessary for Visual Studio Community IDE
#include <iostream>
#include <cmath>
using namespace std;
int main() {
    double i;

    for (i = 0 ; i <= 360; i += 0.5) {
        cout << sin(i * M_PI / 180) << endl;
    }
    return 0;
}

```

8. Solution

```

#define _USE_MATH_DEFINES //This is necessary for Visual Studio Community IDE
#include <iostream>
#include <cmath>
using namespace std;
int main() {
    int deg, i;

    cout << "Enter degrees: ";
    cin >> deg;
    for (i = 0 ; i <= deg; i++) {
        cout << cos(i * M_PI / 180) << endl;
    }
    return 0;
}

```

9. Solution

```

#include <iostream>
using namespace std;
int main() {
    int i, s;

```



```
s = 0;
for (i = 1; i <= 99; i += 2) {
    s += i;
}
cout << s << endl;
return 0;
}
```

10. Solution

```
#include <iostream>
#include <cmath>
using namespace std;
int main() {
    int i, n;
    double p;

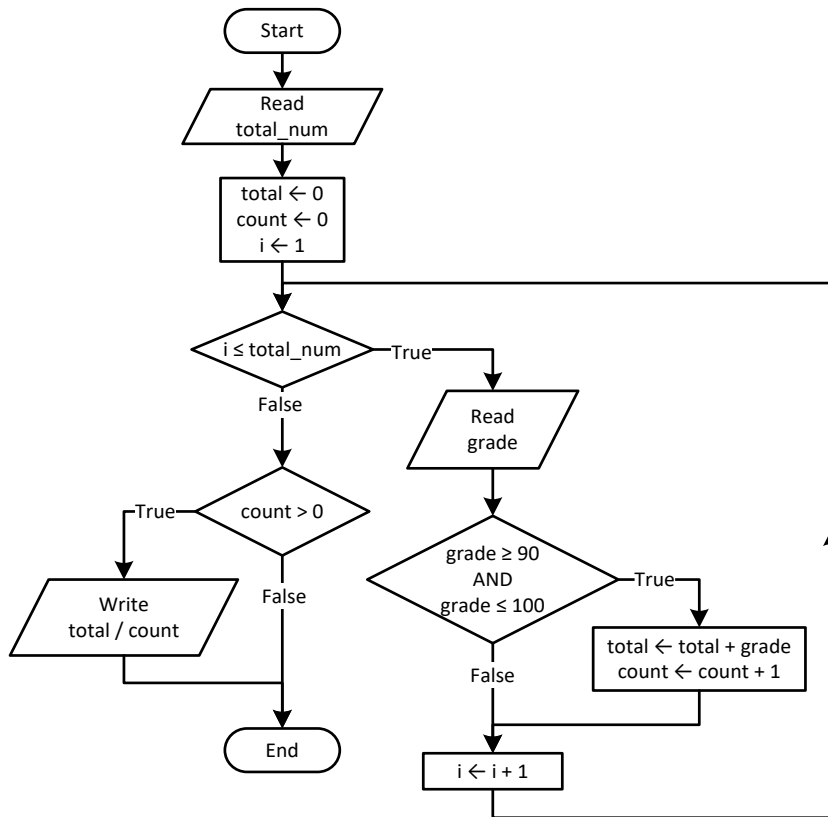
    cin >> n;
    p = 1;
    for (i = 2; i <= 2 * n; i += 2) {
        p *= pow(i, i - 1);
    }
    cout << p << endl;
    return 0;
}
```

11. Solution

```
#include <iostream>
using namespace std;
int main() {
    int i, offset, s;

    s = 0;
    i = 1;
    offset = 0;
    while (i <= 191) {
        s += i;
        offset++;
        i += offset;
    }
    cout << s << endl;
    return 0;
}
```

12. Solution



```

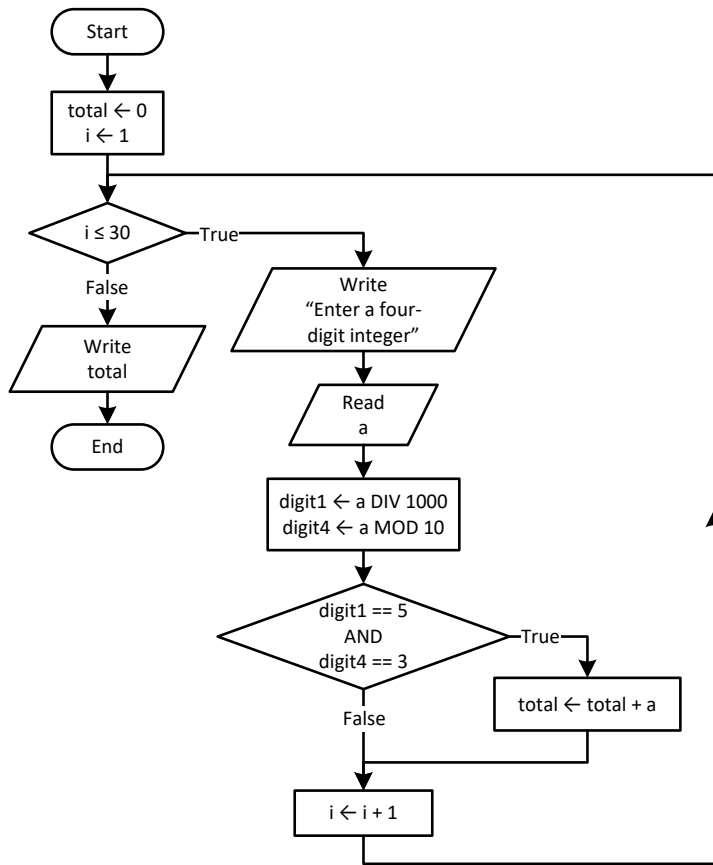
#include <iostream>
using namespace std;
int main() {
    int count, grade, i, totalNum, total;

    cin >> totalNum;
    total = 0;
    count = 0;
    for (i = 1; i <= totalNum; i++) {
        cin >> grade;
        if (grade >= 90 && grade <= 100) {
            total += grade;
            count++;
        }
    }
    if (count > 0) {
        cout << total / (double)count << endl;
    }
    return 0;
}

```

13. Solution

First approach



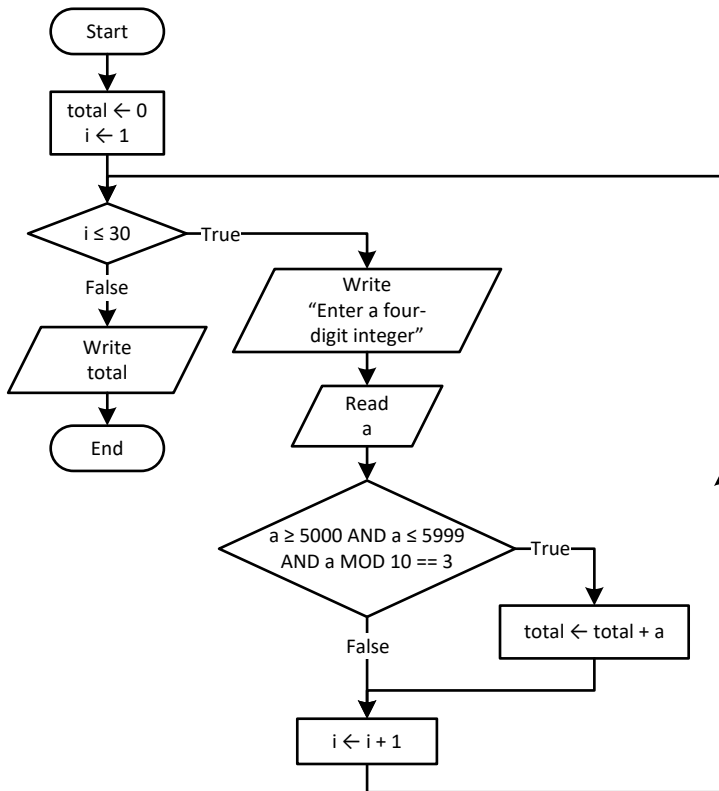
```

#include <iostream>
using namespace std;
int main() {
    int a, digit1, digit4, i, total;

    total = 0;
    for (i = 1; i <= 30; i++) {
        cout << "Enter a four-digit integer: ";
        cin >> a;
        digit1 = (int)(a / 1000);
        digit4 = a % 10;
        if (digit1 == 5 && digit4 == 3) {
            total += a;
        }
    }
    cout << total << endl;
    return 0;
}

```

Second approach



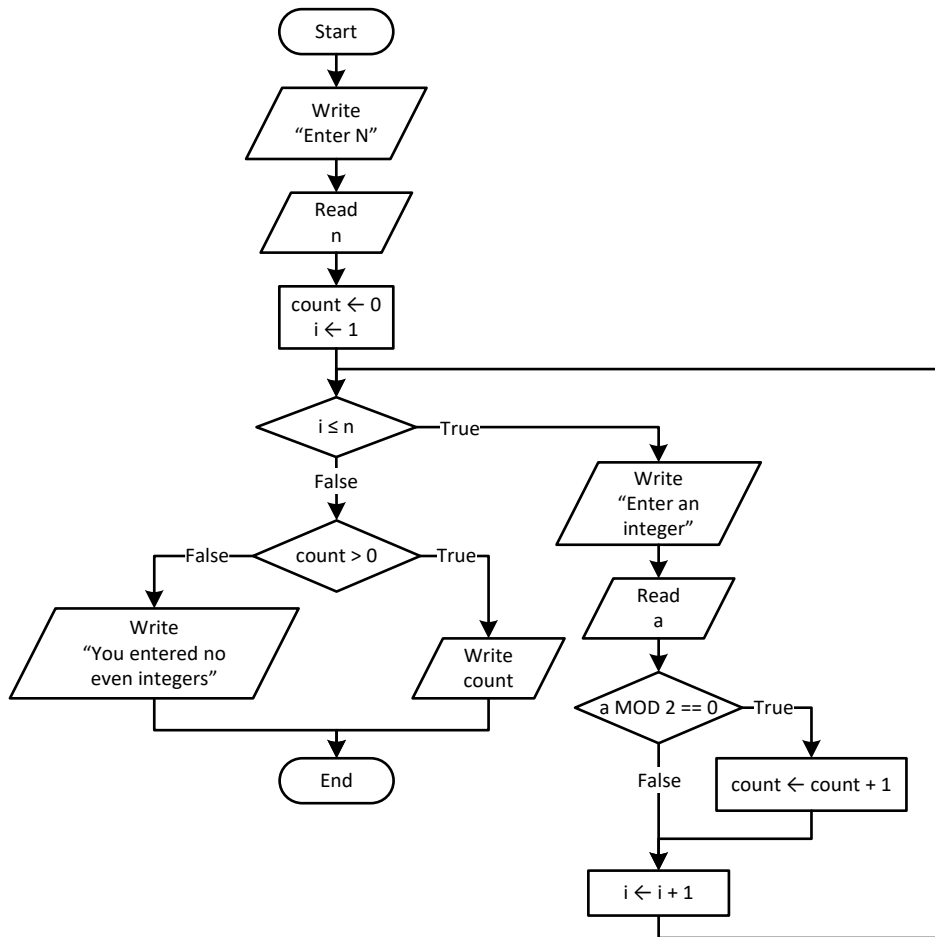
```

#include <iostream>
using namespace std;
int main() {
    int a, i, total;

    total = 0;
    for (i = 1; i <= 30; i++) {
        cout << "Enter a four-digit integer: ";
        cin >> a;
        if (a >= 5000 && a <= 5999 && a % 10 == 3) {
            total += a;
        }
    }
    cout << total << endl;
    return 0;
}

```

14. Solution



```

#include <iostream>
using namespace std;
int main() {
    int a, count, i, n;

    cout << "Enter N: ";
    cin >> n;
    count = 0;
    for (i = 1; i <= n; i++) {
        cout << "Enter an integer: ";
        cin >> a;
        if (a % 2 == 0) {
            count++;
        }
    }
    if (count > 0) {
        cout << count << endl;
    }
    else {
        cout << "You entered no even integers" << endl;
    }
}

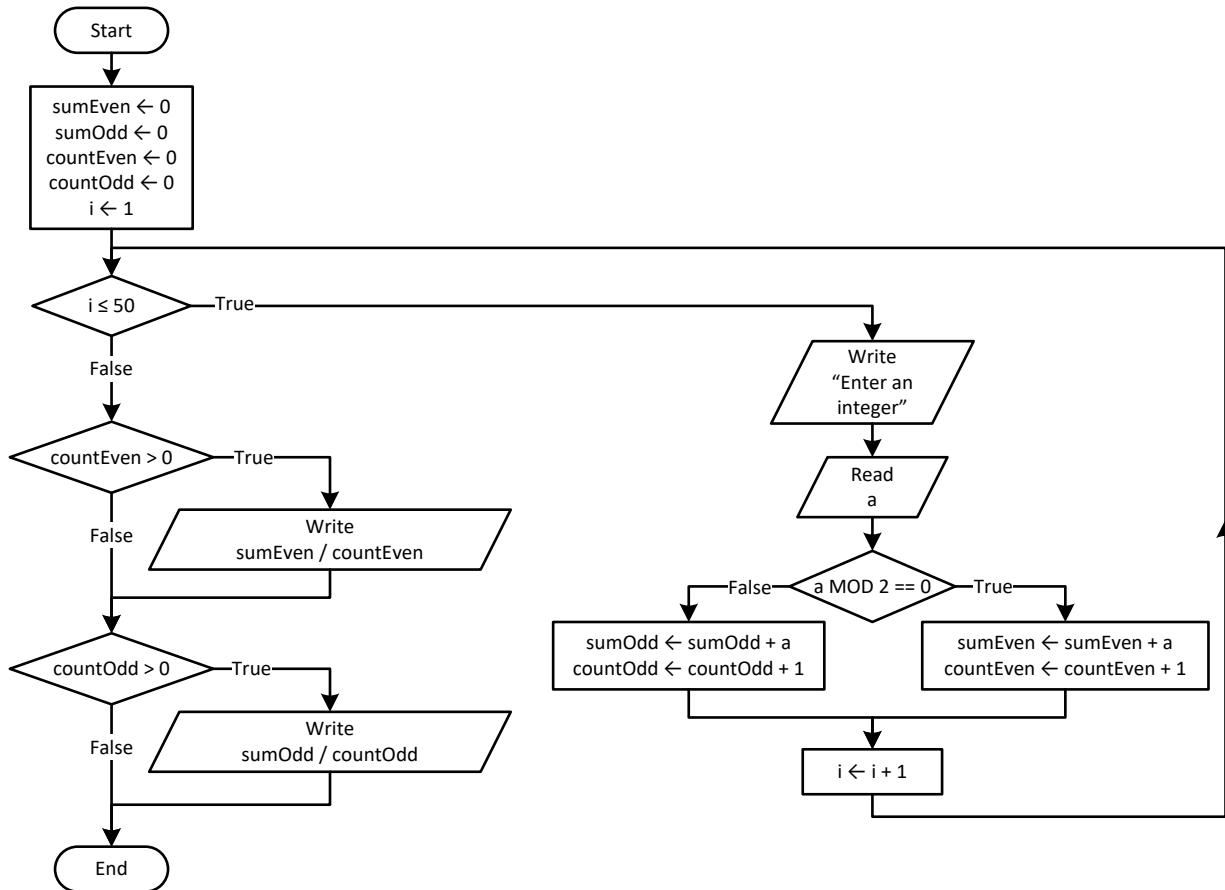
```

```

return 0;
}

```

15. Solution



```

#include <iostream>
using namespace std;
int main() {
    int a, countEven, countOdd, i, sumEven, sumOdd;

    sumEven = 0;
    sumOdd = 0;
    countEven = 0;
    countOdd = 0;
    for (i = 1; i <= 50; i++) {
        cout << "Enter an integer: ";
        cin >> a;
        if (a % 2 == 0) {
            sumEven += a;
            countEven++;
        }
        else {
            sumOdd += a;
            countOdd++;
        }
    }
}

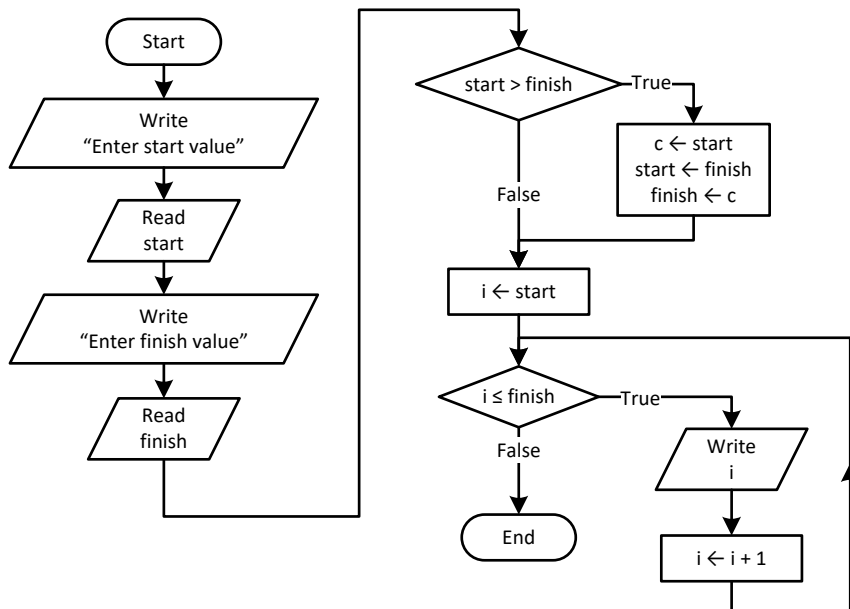
```

```

if (countEven > 0) {
    cout << sumEven / (double)countEven << endl;
}
if (countOdd > 0) {
    cout << sumOdd / (double)countOdd << endl;
}
return 0;
}

```

16. Solution



```

#include <iostream>
using namespace std;
int main() {
    int c, finish, i, start;

    cout << "Enter start value: ";
    cin >> start;
    cout << "Enter finish value: ";
    cin >> finish;

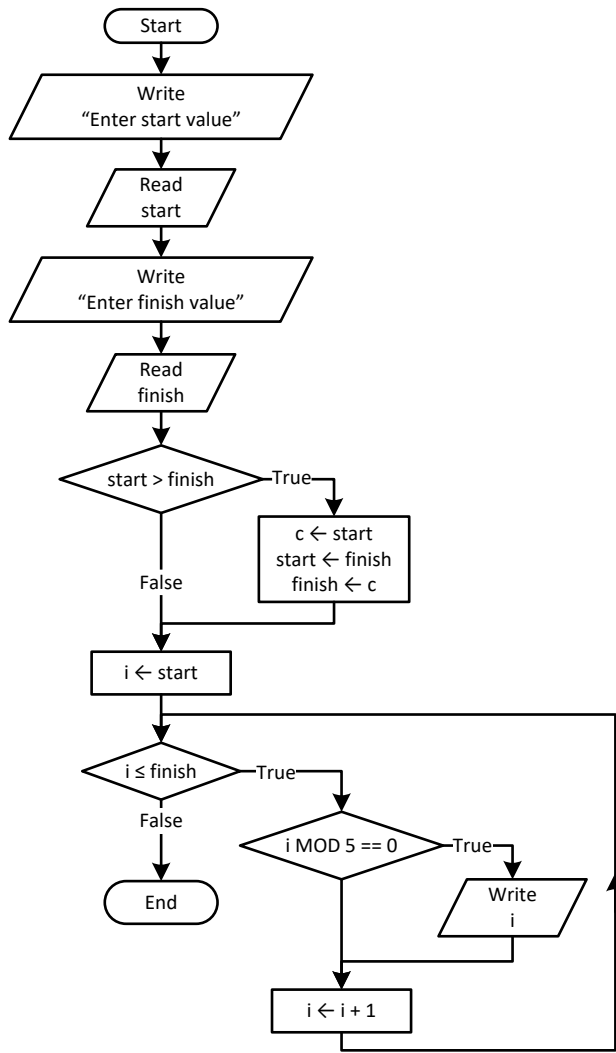
    if (start > finish) {
        c = start;
        start = finish;
        finish = c;
    }

    for (i = start; i <= finish; i++) {
        cout << i << endl;
    }

    return 0;
}

```

17. Solution



```

#include <iostream>
using namespace std;
int main() {
    int c, finish, i, start;

    cout << "Enter start value: ";
    cin >> start;
    cout << "Enter finish value: ";
    cin >> finish;

    if (start > finish) {
        c = start;
        start = finish;
        finish = c;
    }

    for (i = start; i <= finish; i++) {
        if (i % 5 == 0) {

```



```
        cout << i << endl;
    }
}
return 0;
}
```

18. Solution

First approach

```
#include <iostream>
using namespace std;
int main() {
    int exp, i;
    double p, b;

    cout << "Enter a value for base: ";
    cin >> b;
    cout << "Enter an integer for exponent: ";
    cin >> exp;

    p = 1;
    if (exp >= 0) {
        for (i = 1; i <= exp; i++) {
            p *= b;
        }
    }
    else {
        for (i = 1; i <= -exp; i++) {
            p *= 1 / b;
        }
    }
    cout << p << endl;
    return 0;
}
```

Second approach

```
#include <iostream>
using namespace std;
int main() {
    int exp, i;
    double p, b;

    cout << "Enter a value for base: ";
    cin >> b;
    cout << "Enter an integer for exponent: ";
    cin >> exp;

    p = 1;
    for (i = 1; i <= abs(exp); i++) {
        p *= b;
    }
    if (exp < 0) {
        p = 1 / p;
    }
}
```

```
    }  
    cout << p << endl;  
    return 0;  
}
```

19. Solution

```
#include <iostream>  
using namespace std;  
int main() {  
    int count, i, words;  
    string msg;  
  
    cout << "Enter a message: ";  
    getline(cin, msg);  
  
    count = 0;  
    for (i = 0; i <= msg.length() - 1; i++) {  
        if (msg[i] == ' ') {  
            count++;  
        }  
    }  
    words = count + 1;  
  
    cout << "The message entered contains " << words << " words" << endl;  
    return 0;  
}
```

20. Solution

```
#include <iostream>  
using namespace std;  
int main() {  
    int characters, count, i, words;  
    string msg;  
  
    cout << "Enter a message: ";  
    getline(cin, msg);  
  
    characters = msg.length();  
    count = 0;  
    for (i = 0; i <= characters - 1; i++) {  
        if (msg[i] == ' ') {  
            count++;  
        }  
    }  
  
    words = count + 1;  
    cout << "The average number of letters in each word is ";  
    cout << (characters - count) / (double)words << endl;  
    return 0;  
}
```

21. Solution

```
#include <iostream>
#include <boost/algorithm/string.hpp>
using namespace boost::algorithm;
using namespace std;
int main() {
    string message;
    char character;
    string consonants = "BCDFGHJKLMNPQRSTVWZYX";
    int i, count;

    cout << "Enter an English message: ";
    getline(cin, message);
    message = to_upper_copy(message);

    count = 0;
    for (i = 0; i <= message.length() - 1; i++) {
        character = message[i];

        if (consonants.find(character) != -1) { //If character is found in consonants
            count++;
        }
    }
    cout << "Consonants: " << count << endl;
    return 0;
}
```

22. Solution

```
#include <iostream>
#include <boost/algorithm/string.hpp>
using namespace boost::algorithm;
using namespace std;
int main() {
    string message;
    char character;
    string vowels = "AEIOU";
    string consonants = "BCDFGHJKLMNPQRSTVWZYX";
    string digits = "0123456789";
    int i, countv, countc, countd;

    cout << "Enter an English message: ";
    getline(cin, message);
    message = to_upper_copy(message);

    countv = countc = countd = 0;
    for (i = 0; i <= message.length() - 1; i++) {
        character = message[i];

        if (vowels.find(character) != -1) { //If character is found in vowels
            countv++;
        }
    }
}
```

```
    }  
    else if (consonants.find(character) != -1) { //If character is found in consonants  
        countc++;  
    }  
    else if (digits.find(character) != -1) { //If character is found in digits  
        countd++;  
    }  
}  
cout << "Vowels: " << countv << endl;  
cout << "Consonants: " << countc << endl;  
cout << "Digits: " << countd << endl;  
return 0;  
}
```

Chapter 27

27.3 Review Questions: True/False

- | | |
|----------|----------|
| 1. true | 6. false |
| 2. true | 7. true |
| 3. false | 8. true |
| 4. true | 9. true |
| 5. true | 10. true |

27.4 Review Questions: Multiple Choice

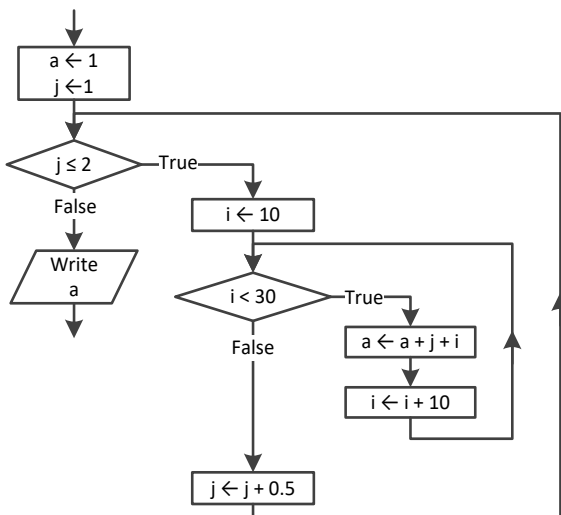
- | | |
|------|------|
| 1. b | 4. a |
| 2. a | 5. b |
| 3. c | |

27.5 Review Exercises

1. Solution

- 10
- A value greater than or equal to 4.5 and less than 5.0 ($4.5 \leq x < 5.0$)
- 7 (or -8)
- 138 (or 139)

2. Solution



Step	Statement	a	i	j
1	a = 1	1	?	?
2	j = 1	1	?	1
3	j ≤ 2	true		
4	i = 10	1	10	1
5	i < 30	true		

6	<code>a = a + j + i</code>	12	10	1
7	<code>i += 10</code>	12	20	1
8	<code>i < 30</code>	true		
9	<code>a = a * j + i</code>	33	20	1
10	<code>i += 10</code>	33	30	1
11	<code>i < 30</code>	false		
12	<code>j += 0.5</code>	33	30	1.5
13	<code>j <= 2</code>	true		
14	<code>i = 10</code>	33	10	1.5
15	<code>i < 30</code>	true		
16	<code>a = a + j + i</code>	44.5	10	1.5
17	<code>i += 10</code>	44.5	20	1.5
18	<code>i < 30</code>	true		
19	<code>a = a * j + i</code>	66	20	1.5
20	<code>i += 10</code>	66	30	1.5
21	<code>i < 30</code>	false		
22	<code>j += 0.5</code>	66	30	2
23	<code>j <= 2</code>	true		
24	<code>i = 10</code>	66	10	2
25	<code>i < 30</code>	true		
26	<code>a = a + j + i</code>	78	10	2
27	<code>i += 10</code>	78	20	2
28	<code>i < 30</code>	true		
29	<code>a = a * j + i</code>	100	20	2
30	<code>i += 10</code>	100	30	2
31	<code>i < 30</code>	false		
32	<code>j += 0.5</code>	100	30	2.5
33	<code>j <= 2</code>	false		
34	<code>cout << a << endl</code>	It displays: 100		

3. Solution

Step	Statement	s	i	j
1	<code>s = 0</code>	0	?	?
2	<code>i = 1</code>	0	1	?
3	<code>i <= 4</code>	true		
4	<code>j = 3</code>	0	1	3
5	<code>j >= i</code>	true		

6	<code>s = s + i * j</code>	3	1	3
7	<code>j--</code>	3	1	2
8	<code>j >= i</code>	true		
9	<code>s = s + i * j</code>	5	1	2
10	<code>j--</code>	5	1	1
11	<code>j >= i</code>	true		
12	<code>s = s + i * j</code>	6	1	1
13	<code>j--</code>	6	1	0
14	<code>j >= i</code>	false		
15	<code>i++</code>	6	2	0
16	<code>i <= 4</code>	true		
17	<code>j = 3</code>	6	2	3
18	<code>j >= i</code>	true		
19	<code>s = s + i * j</code>	12	2	3
20	<code>j--</code>	12	2	2
21	<code>j >= i</code>	true		
22	<code>s = s + i * j</code>	16	2	2
23	<code>j--</code>	16	2	1
24	<code>j >= i</code>	false		
25	<code>i++</code>	16	3	1
26	<code>i <= 4</code>	true		
27	<code>j = 3</code>	16	3	3
28	<code>j >= i</code>	true		
29	<code>s = s + i * j</code>	25	3	3
30	<code>j--</code>	25	3	2
31	<code>j >= i</code>	false		
32	<code>i++</code>	25	4	2
33	<code>i <= 4</code>	true		
34	<code>j = 3</code>	25	4	3
35	<code>j >= i</code>	false		
36	<code>i++</code>	25	5	3
37	<code>i <= 4</code>	false		
38	<code>cout << s << endl</code>	It displays: 25		

The statement `s = s + i * j` is executed 6 times

4. Solution

For input value of "NO"

Step	Statement	s	y	i	ans
1	s = 1	1	?	?	?
2	y = 25	1	25	?	?
3	i = 1	1	25	1	?
4	i <= 3	true			
5	s = s + y	26	25	1	?
6	y -= 5	26	20	1	?
7	i++	26	20	2	?
8	i <= 3	true			
9	s = s + y	46	20	2	?
10	y -= 5	46	15	2	?
11	i++	46	15	3	?
12	i <= 3	true			
13	s = s + y	61	15	3	?
14	y -= 5	61	10	3	?
15	i++	61	10	4	?
16	i <= 3	false			
17	cin >> ans	61	10	4	"NO"
18	while (ans == "YES")	false			
19	cout << s << endl	It displays: 61			

For input values of "YES", "NO"

Step	Statement	s	y	i	ans
1	s = 1	1	?	?	?
2	y = 25	1	25	?	?
3	i = 1	1	25	1	?
4	i <= 3	true			
5	s = s + y	26	25	1	?
6	y -= 5	26	20	1	?
7	i++	26	20	2	?
8	i <= 3	true			
9	s = s + y	46	20	2	?
10	y -= 5	46	15	2	?
11	i++	46	15	3	?
12	i <= 3	true			
13	s = s + y	61	15	3	?
14	y -= 5	61	10	3	?
15	i++	61	10	4	?

16	<code>i <= 3</code>	false			
17	<code>cin >> ans</code>	61	10	4	"YES"
18	<code>while (ans == "YES")</code>	true			
19	<code>i = 1</code>	61	10	1	"YES"
20	<code>i <= 3</code>	true			
21	<code>s = s + y</code>	71	10	1	"YES"
22	<code>y -= 5</code>	71	5	1	"YES"
23	<code>i++</code>	71	5	2	"YES"
24	<code>i <= 3</code>	true			
25	<code>s = s + y</code>	76	5	2	"YES"
26	<code>y -= 5</code>	76	0	2	"YES"
27	<code>i++</code>	76	0	3	"YES"
28	<code>i <= 3</code>	true			
29	<code>s = s + y</code>	76	0	3	"YES"
30	<code>y -= 5</code>	76	-5	3	"YES"
31	<code>i++</code>	76	-5	4	"YES"
32	<code>i <= 3</code>	false			
33	<code>cin >> ans</code>	76	-5	4	"NO"
34	<code>while (ans == "YES")</code>	false			
35	<code>cout << s << endl</code>	It displays: 76			

For input values of "YES", "YES", "NO"

Step	Statement	s	y	i	ans
1	<code>s = 1</code>	1	?	?	?
2	<code>y = 25</code>	1	25	?	?
3	<code>i = 1</code>	1	25	1	?
4	<code>i <= 3</code>	true			
5	<code>s = s + y</code>	26	25	1	?
6	<code>y -= 5</code>	26	20	1	?
7	<code>i++</code>	26	20	2	?
8	<code>i <= 3</code>	true			
9	<code>s = s + y</code>	46	20	2	?
10	<code>y -= 5</code>	46	15	2	?
11	<code>i++</code>	46	15	3	?
12	<code>i <= 3</code>	true			
13	<code>s = s + y</code>	61	15	3	?
14	<code>y -= 5</code>	61	10	3	?
15	<code>i++</code>	61	10	4	?

16	<code>i <= 3</code>	false			
17	<code>cin >> ans</code>	61	10	4	"YES"
18	<code>while (ans == "YES")</code>	true			
19	<code>i = 1</code>	61	10	1	"YES"
20	<code>i <= 3</code>	true			
21	<code>s = s + y</code>	71	10	1	"YES"
22	<code>y -= 5</code>	71	5	1	"YES"
23	<code>i++</code>	71	5	2	"YES"
24	<code>i <= 3</code>	true			
25	<code>s = s + y</code>	76	5	2	"YES"
26	<code>y -= 5</code>	76	0	2	"YES"
27	<code>i++</code>	76	0	3	"YES"
28	<code>i <= 3</code>	true			
29	<code>s = s + y</code>	76	0	3	"YES"
30	<code>y -= 5</code>	76	-5	3	"YES"
31	<code>i++</code>	76	-5	4	"YES"
32	<code>i <= 3</code>	false			
33	<code>cin >> ans</code>	76	-5	4	"YES"
34	<code>while (ans == "YES")</code>	true			
35	<code>i = 1</code>	76	-5	1	"YES"
36	<code>i <= 3</code>	true			
37	<code>s = s + y</code>	71	-5	1	"YES"
38	<code>y -= 5</code>	71	-10	1	"YES"
39	<code>i++</code>	71	-10	2	"YES"
40	<code>i <= 3</code>	true			
41	<code>s = s + y</code>	61	-10	2	"YES"
42	<code>y -= 5</code>	61	-15	2	"YES"
43	<code>i++</code>	61	-15	3	"YES"
44	<code>i <= 3</code>	true			
45	<code>s = s + y</code>	46	-15	3	"YES"
46	<code>y -= 5</code>	46	-20	3	"YES"
47	<code>i++</code>	46	-20	4	"YES"
48	<code>i <= 3</code>	false			
49	<code>cin >> ans</code>	46	-20	4	"NO"
50	<code>while (ans == "YES")</code>	false			
51	<code>cout << s << endl</code>	It displays: 46			

5. Solution

```
#include <iostream>
using namespace std;
int main() {
    int hour, minutes;

    for (hour = 0; hour <= 23; hour++) {
        for (minutes = 0; minutes <= 59; minutes++) {
            cout << hour << "\t" << minutes << endl;
        }
    }
    return 0;
}
```

6. Solution

```
#include <iostream>
using namespace std;
int main() {
    int i, j;

    for (i = 5; i >= 1; i--) {
        for (j = 1; j <= i; j++) {
            cout << i << " ";
        }
        cout << endl;
    }
    return 0;
}
```

7. Solution

```
#include <iostream>
using namespace std;
int main() {
    int i, j;

    for (i = 0; i <= 5; i++) {
        for (j = 0; j <= i; j++) {
            cout << j << " ";
        }
        cout << endl;
    }
    return 0;
}
```

8. Solution

```
#include <iostream>
using namespace std;
int main() {
```

```
int i, j;

for (i = 1; i <= 4; i++) {
    for (j = 1; j <= 10; j++) {
        cout << "*" ";
    }
    cout << endl;
}
return 0;
}
```

9. Solution

```
#include <iostream>
using namespace std;
int main() {
    int i, j, y;

    cout << "Enter an integer between 3 and 20: ";
    cin >> y;

    for (i = 1; i <= y; i++) {
        for (j = 1; j <= y; j++) {
            cout << "*" ";
        }
        cout << endl;
    }
    return 0;
}
```

10. Solution

```
#include <iostream>
using namespace std;
int main() {
    int i, j, y;

    cout << "Enter an integer between 3 and 20: ";
    cin >> y;

    for (j = 1; j <= y; j++) {
        cout << "*" ";
    }
    cout << endl;

    for (i = 1; i <= y - 2; i++) {
        cout << "*" ";
        for (j = 1; j <= y - 2; j++) {
            cout << " ";
        }
        cout << "*" " << endl;
    }
}
```

```
    for (j = 1; j <= y; j++) {  
        cout << "*" << " ";  
    }  
    return 0;  
}
```

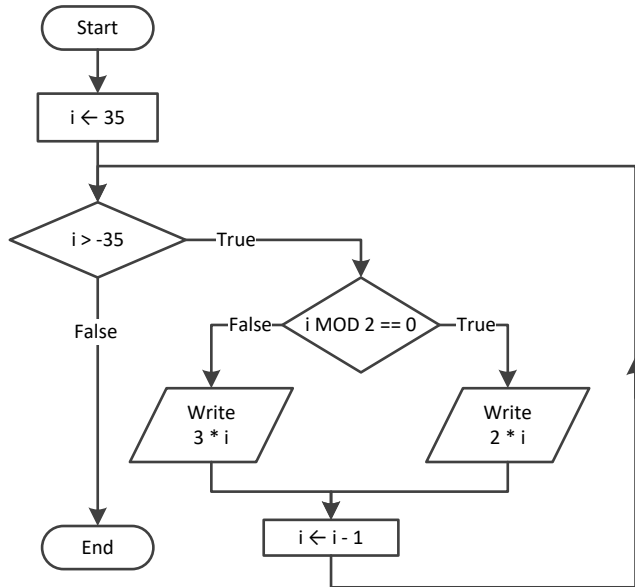
11. Solution

```
#include <iostream>  
using namespace std;  
int main() {  
    int i, j;  
  
    for (i = 1; i <= 5; i++) {  
        for (j = 1; j <= i; j++) {  
            cout << "*" << " ";  
        }  
        cout << endl;  
    }  
  
    for (i = 4; i >= 1; i--) {  
        for (j = 1; j <= i; j++) {  
            cout << "*" << " ";  
        }  
        cout << endl;  
    }  
    return 0;  
}
```

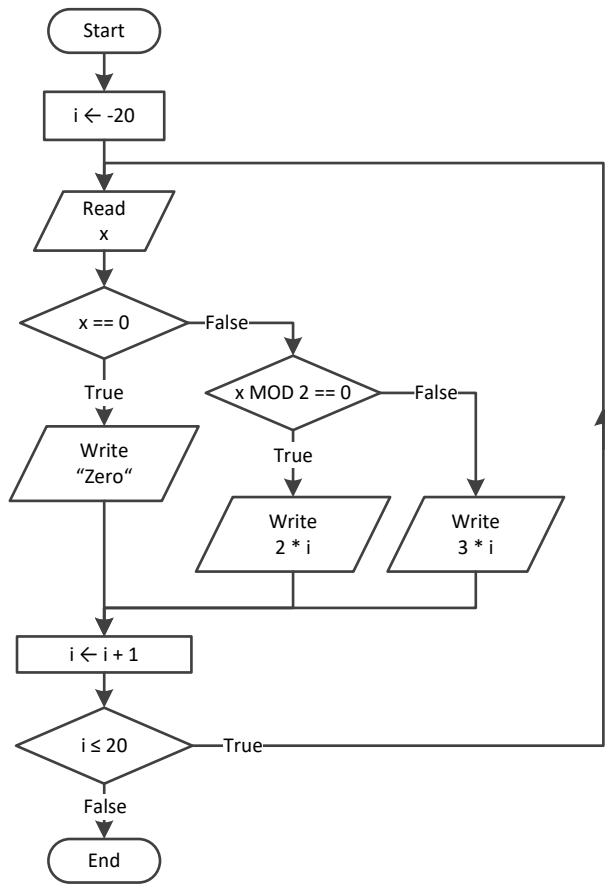
Chapter 28

28.4 Review Exercises

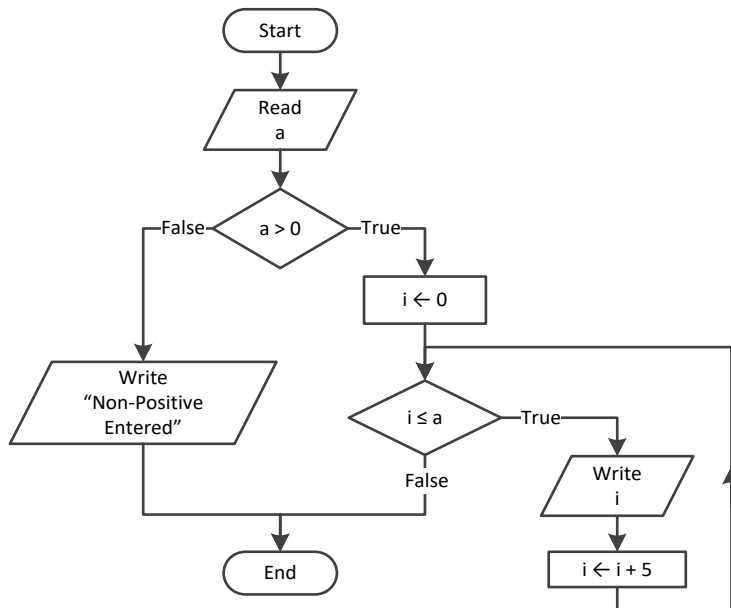
1. Solution



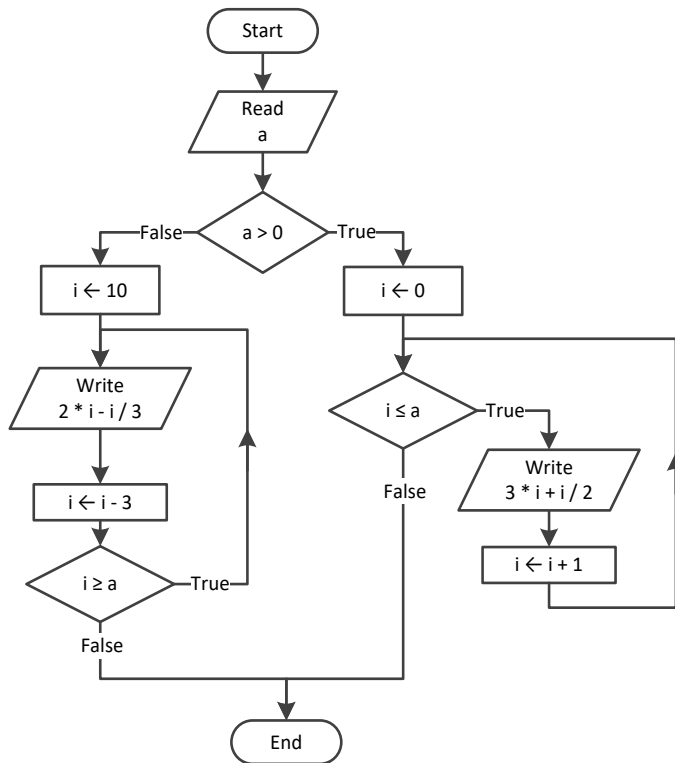
2. Solution



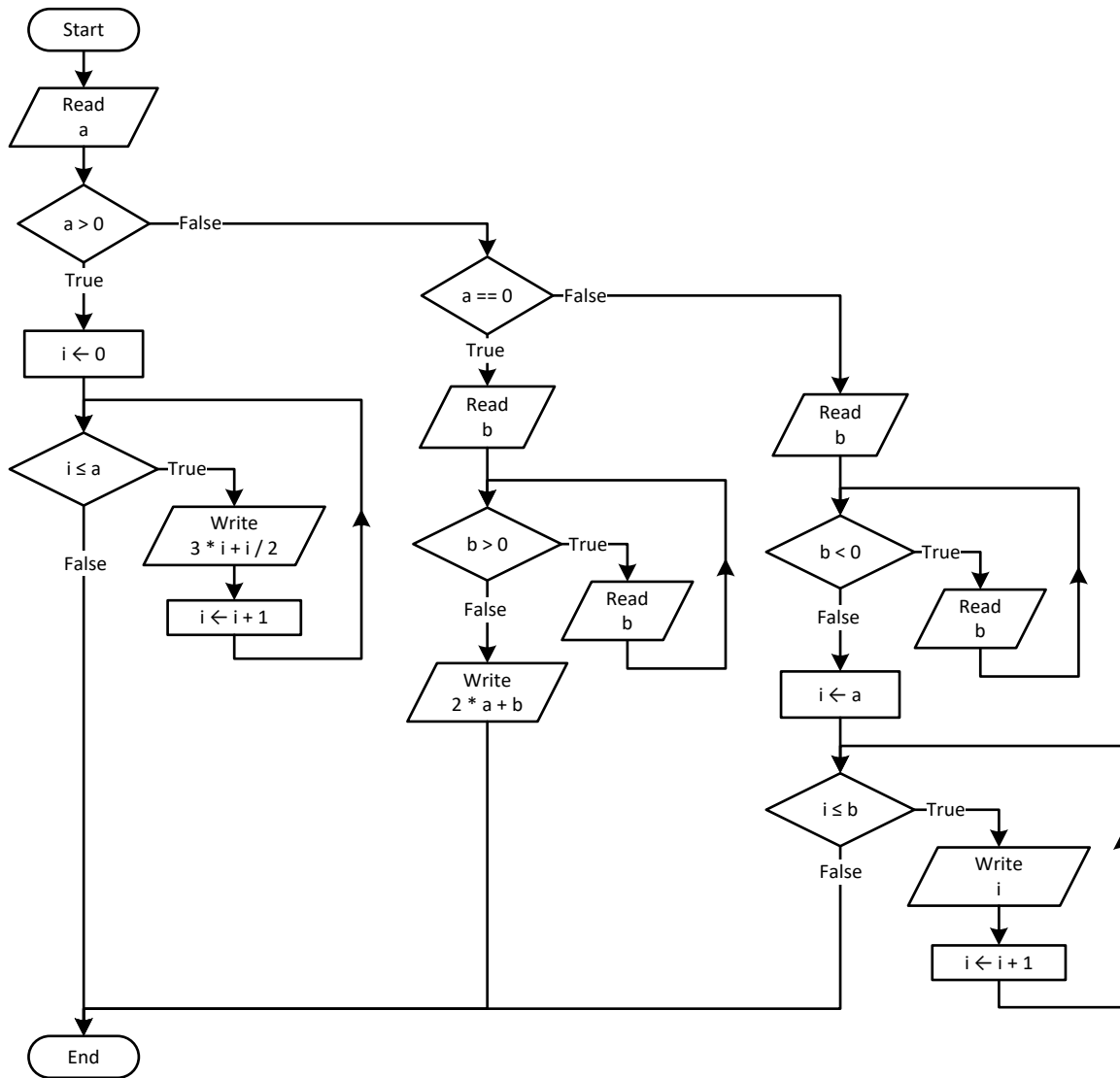
3. Solution

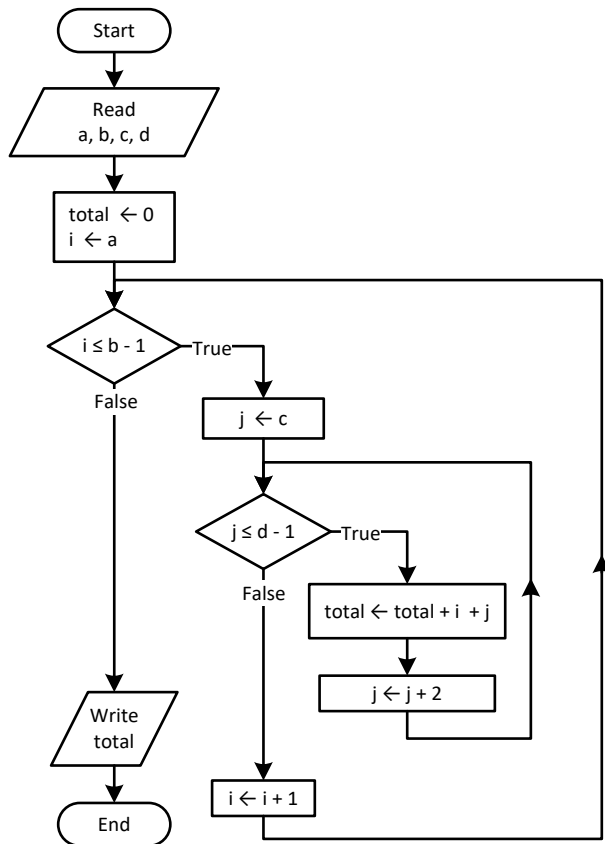


4. Solution

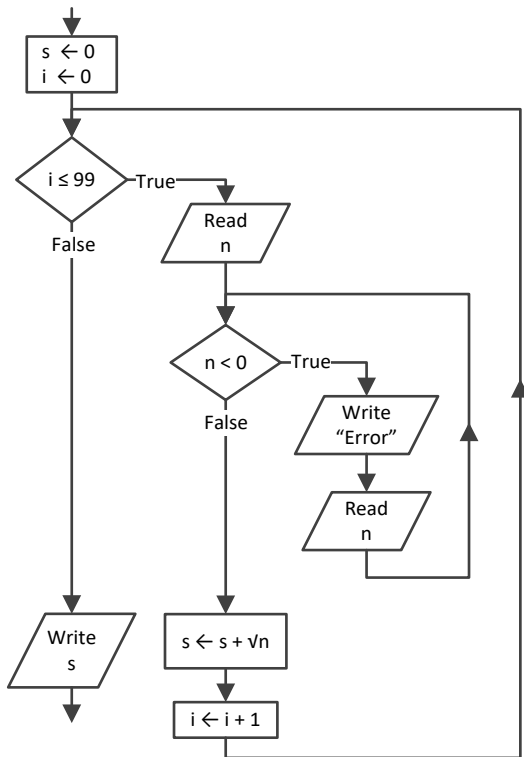


5. Solution

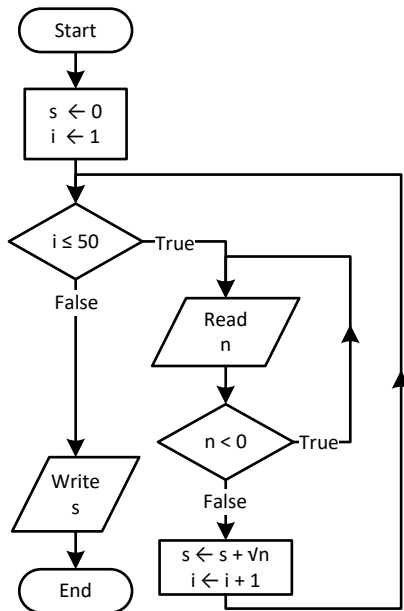


6. Solution

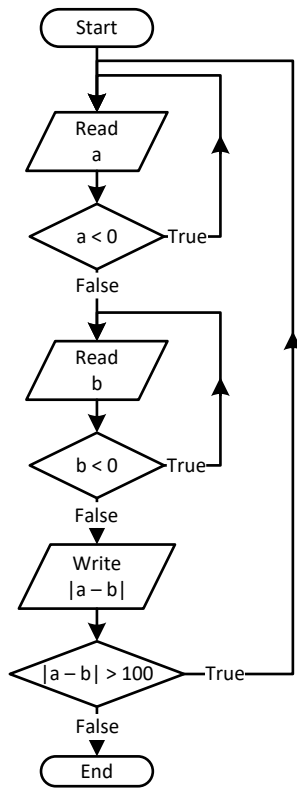
7. Solution



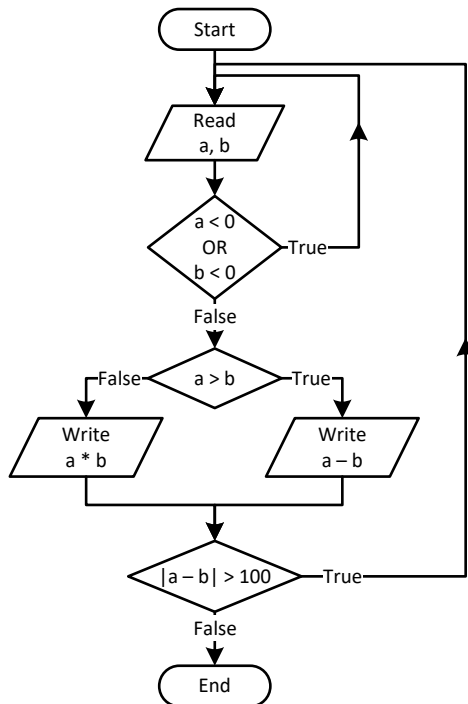
8. Solution



9. Solution



10. Solution



11. Solution

```
#include <iostream>
using namespace std;
int main() {
    int a, i;

    i = 0;
    cin >> a;
    do {
        if (i % 2 != 0) {
            cout << i << endl;
        }
        i += 5;
    } while (i < a);
    return 0;
}
```

12. Solution

```
#include <iostream>
using namespace std;
int main() {
    int a, b, i;

    cin >> a;
    while (a != -1) {
        do {
            cin >> b;
        } while (b <= a);
        for (i = a; i <= b; i++) {
            cout << i << endl;
        }
        cin >> a;
    }
    return 0;
}
```

13. Solution

```
#include <iostream>
using namespace std;
int main() {
    int i;
    double P, S, a;

    i = 1;
    S = 0;
    P = 1;
    a = 0;

    while (true) {
```

```
    if (i < 45) {  
        S += a;  
    }  
    else {  
        P *= a;  
    }  
    i++;  
    if (i >= 90) break;  
    cin >> a;  
}  
  
cout << S << " " << P << endl;  
return 0;  
}
```

Chapter 29

29.8 Review Questions: True/False

- | | |
|----------|-----------|
| 1. false | 8. false |
| 2. false | 9. true |
| 3. false | 10. true |
| 4. true | 11. false |
| 5. true | 12. false |
| 6. false | 13. false |
| 7. false | 14. true |

29.9 Review Questions: Multiple Choice

- | | |
|------|------|
| 1. c | 5. c |
| 2. d | 6. c |
| 3. b | 7. c |
| 4. a | |

29.10 Review Exercises

1. Solution

```

countNames = 0;
countNotJohns = 0;
name = "";
cout << "Enter a name: ";
cin >> name;
while (name != "STOP") {
    cout << "Enter a name: ";
    cin >> name;
    countNames++;
    if (name != "John") {
        countNotJohns++;
    }
    cout << "Enter a name: ";
    cin >> name;
}
cout << "Total names entered: " << countNames << endl;
cout << "Names other than John entered: " << countNotJohns << endl;

```

2. Solution

First approach

```

#include <iostream>
using namespace std;
int main() {
    string text, character;
    bool found;
    int i;

    cout << "Enter a text: ";
    getline(cin, text);

```

```

found = false;
for (i = 0; i <= text.length() - 1; i++) {
    character = text[i];
    if (character == " ") {
        found = true;
        break;
    }
}

if (!found) {
    cout << "One Single Word" << endl;
}
else {
    cout << "Complete Sentence" << endl;
}
return 0;
}

```

Second approach

```

#include <iostream>
using namespace std;
int main() {
    string text;

    cout << "Enter a text: ";
    getline(cin, text);

    if (text.find(" ") == -1) {
        cout << "One Single Word" << endl;
    }
    else {
        cout << "Complete Sentence" << endl;
    }
    return 0;
}

```

3. Solution

First approach

```

#include <iostream>
using namespace std;
int main() {
    string sentence, character;
    bool found;
    int i;
    string digits = "0123456789";

    cout << "Enter a text: ";
    getline(cin, sentence);

    found = false;
    for (i = 0; i <= sentence.length() - 1; i++) {
        character = sentence[i];
        if (digits.find(character) != -1) {

```



```

        found = true;
        break;
    }
}

if (found) {
    cout << "The sentence contains a number" << endl;
}
return 0;
}

```

Second approach

```

#include <iostream>
using namespace std;
int main() {
    string sentence;
    bool found;
    int i;
    string digit;

    cout << "Enter a text: ";
    getline(cin, sentence);

    found = false;
    for (i = 0; i <= 9; i++) {
        digit = to_string(i);
        if (sentence.find(digit) != -1) {
            found = true;
            break;
        }
    }

    if (found) {
        cout << "The sentence contains a number" << endl;
    }
    return 0;
}

```

4. Solution

```

cout << "Printing all integers from 1 to 100" << endl;
i = 1;
while (i < 101) {
    cout << i << endl;
    i++;
}

```

5. Solution

```

cout << "Printing odd integers from 1 to 99" << endl;
i = 1;
while (i < 100) {
    cout << i << endl;
    i += 2;
}

```

```
| }
```

6. Solution

```
| s = 0;  
| i = 1;  
| count = 100;  
| do {  
|     cin >> number;  
|     s = s + number;  
|     i++;  
| } while (i <= count);  
| average = s / count;  
| cout << average << endl;
```

7. Solution

```
| int i, denom;  
| double s;  
  
| s = 0;  
  
| denom = 1;  
| for (i = 1; i <= 100; i++) {  
|     denom *= i;  
| }  
  
| for (i = 1; i <= 100; i++) {  
|     s += i / (double)denom;  
| }  
| cout << s << endl;
```

8. Solution

```
| #include <iostream>  
| using namespace std;  
| int main() {  
|     int i, j;  
  
|     for (i = 1; i <= 4; i++) {  
|         for (j = 1; j <= 4; j++) {  
|             cout << i << " x " << j << " = " << i * j << endl;  
|         }  
|     }  
|     return 0;  
| }
```

9. Solution

```
| #include <iostream>  
| using namespace std;  
| int main() {  
|     int i, j;
```

```
cout << "\\t\\t";
for (i = 1; i <= 12; i++) {
    cout << i << "\\t";
}
cout << endl;

for (i = 1; i <= 12; i++) {
    cout << "-----";
}
cout << endl;

for (i = 1; i <= 12; i++) {
    cout << i << "\\t\\t";
    for (j = 1; j <= 12; j++) {
        cout << i * j << "\\t";
    }
    cout << endl;
}
return 0;
}
```

10. Solution

```
#include <iostream>
using namespace std;
int main() {
    int i, j, n;

    cout << "Enter an integer: ";
    cin >> n;

    cout << "\\t\\t";
    for (i = 1; i <= n; i++) {
        cout << i << "\\t";
    }
    cout << endl;

    for (i = 1; i <= n; i++) {
        cout << "-----";
    }
    cout << endl;

    for (i = 1; i <= n; i++) {
        cout << i << "\\t\\t";
        for (j = 1; j <= n; j++) {
            cout << i * j << "\\t";
        }
        cout << endl;
    }
    return 0;
}
```

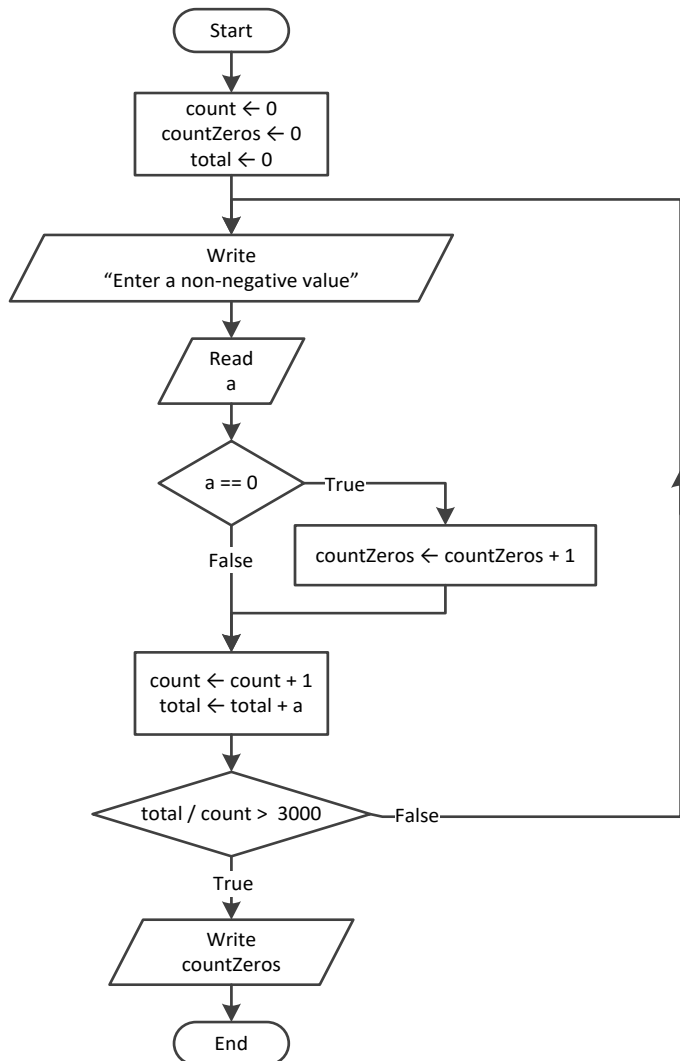

Chapter 30

30.7 Review Questions: True/False

1. true
2. false
3. true
4. false
5. false
6. false
7. true

30.8 Review Exercises

1. Solution



```

#include <iostream>
using namespace std;
int main() {
    int count, countZeros;
    double a, total;

    count = 0;
    countZeros = 0;
  
```

```

total = 0;
do {
    cout << "Enter a non-negative value: ";
    cin >> a;
    if (a == 0) {
        countZeros++;
    }
    count++;
    total += a;
} while (total / count <= 3000);
cout << countZeros << endl;
return 0;
}

```

2. Solution

First approach

```

#include <iostream>
using namespace std;
int main() {
    int a, d1, d2, d3, d4, i, r;

    cout << "Enter an integer between 1 and 20: ";
    cin >> a;
    for (i = 1000; i <= 9999; i++) {
        d4 = i % 10;
        r = (int)(i / 10);
        d3 = r % 10;
        r = (int)(r / 10);
        d2 = r % 10;
        d1 = (int)(r / 10);
        if (d1 + d2 + d3 + d4 < a) {
            cout << i << endl;
        }
    }
    return 0;
}

```

Second approach

```

#include <iostream>
using namespace std;
int main() {
    int a, d1, d2, d3, d4;

    cout << "Enter an integer between 1 and 20: ";
    cin >> a;
    for (d1 = 1; d1 <= 9; d1++) {
        for (d2 = 0; d2 <= 9; d2++) {
            for (d3 = 0; d3 <= 9; d3++) {
                for (d4 = 0; d4 <= 9; d4++) {
                    if (d1 + d2 + d3 + d4 < a) {
                        cout << d1 * 1000 + d2 * 100 + d3 * 10 + d4 << endl;
                    }
                }
            }
        }
    }
}

```

```

    }
    }
}
}
return 0;
}

```

3. Solution

First approach

```

#include <iostream>
using namespace std;
int main() {
    int d1, d2, d3, d4, i, r;

    for (i = 1000; i <= 9999; i++) {
        d4 = i % 10;
        r = (int)(i / 10);
        d3 = r % 10;
        r = (int)(r / 10);
        d2 = r % 10;
        d1 = (int)(r / 10);
        if (d1 > d2 && d2 == d3 && d3 < d4) {
            cout << i << endl;
        }
    }
    return 0;
}

```

Second approach

```

#include <iostream>
using namespace std;
int main() {
    int d1, d2, d3, d4;

    for (d1 = 1; d1 <= 9; d1++) {
        for (d2 = 0; d2 <= 9; d2++) {
            for (d3 = 0; d3 <= 9; d3++) {
                for (d4 = 0; d4 <= 9; d4++) {
                    if (d1 > d2 && d2 == d3 && d3 < d4) {
                        cout << d1 * 1000 + d2 * 100 + d3 * 10 + d4 << endl;
                    }
                }
            }
        }
    }
    return 0;
}

```

4. Solution

First approach

```
#include <iostream>
using namespace std;
int main() {
    int x, count;

    cout << "Enter an integer: ";
    cin >> x;

    count = 0;

    while (x != 0) {
        count++;
        x = (int)(x / 10);
    }

    cout << count << endl;
    return 0;
}
```

Second approach

```
#include <iostream>
using namespace std;
int main() {
    int x, count;

    cout << "Enter an integer: ";
    cin >> x;

    //Convert the absolute value of x to string and get its length
    count = to_string(abs(x)).length();

    cout << count << endl;
    return 0;
}
```

5. Solution

```
cin >> x;
while (x != 1 && x != 0) {
    cout << "Error" << endl;
    cin >> x;
}
```

6. Solution

```
do {
    cin >> gender;
    gender = to_upper_copy(gender);
} while (gender != "M" && gender != "F" && gender != "O");
```


7. Solution

```
#include <iostream>
#include <cmath>
using namespace std;
int main() {
    int count;
    double x, y;

    cout << "Enter a non-negative number: ";
    cin >> x;
    count = 0;
    while (x < 0) {
        count++;
        if (count == 2) break;

        cout << "Error: Invalid number!" << endl;
        cout << "Enter a non-negative number: ";
        cin >> x;
    }

    if (count < 2) {
        y = sqrt(x);
        cout << y << endl;
    }
    else {
        cout << "Dude, you are dumb!" << endl;
    }
    return 0;
}
```

8. Solution

```
#define _USE_MATH_DEFINES //This is necessary for Visual Studio Community IDE
#include <iostream>
#include <cmath>
#include <boost/algorithm/string.hpp>
using namespace boost::algorithm;
using namespace std;
int main() {
    string answer;
    double area, r;

    do {
        cout << "Enter the length of a radius: ";
        cin >> r;
        while (r <= 0) {
            cout << "Invalid radius. Enter the length of a radius: ";
            cin >> r;
        }

        area = M_PI * pow(r, 2);
    }
```

```

    cout << "The area is: " << area << endl;

    cout << "Would you like to repeat? ";
    cin >> answer;
} while (to_upper_copy(answer) == "YES");

return 0;
}

```

9. Solution

```

#include <iostream>
using namespace std;
int main() {
    int i;
    double maximum, total, t;

    maximum = -460;
    total = 0;
    for (i = 1; i <= 31; i++) {
        cout << "Enter temperature for day " << i << ": ";
        cin >> t;
        while (t < -459.67) {
            cout << "Error! Wrong temperature." << endl;
            cout << "Enter temperature for day " << i << ": ";
            cin >> t;
        }

        total += t;
        if (t > maximum) {
            maximum = t;
        }
    }

    cout << total / 31 << " " << maximum << endl;
    return 0;
}

```

10. Solution

```

#include <iostream>
using namespace std;
int main() {
    int hour, maxHour, maxMinutes, minHour, minMinutes, minutes;
    double level, maximum, minimum;

    cin >> level;
    if (level != 9999) {
        cin >> hour;
        cin >> minutes;

        maximum = level;
        maxHour = hour;
    }
}

```

```
maxMinutes = minutes;

minimum = level;
minHour = hour;
minMinutes = minutes;

cin >> level;
while (level != 9999) {
    cin >> hour;
    cin >> minutes;

    if (level > maximum) {
        maximum = level;
        maxHour = hour;
        maxMinutes = minutes;
    }

    if (level < minimum) {
        minimum = level;
        minHour = hour;
        minMinutes = minutes;
    }

    cin >> level;
}

cout << maximum << ", " << maxHour << ", " << maxMinutes << endl;
cout << minimum << ", " << minHour << ", " << minMinutes << endl;
}
return 0;
}
```

11. Solution

```
#include <iostream>
using namespace std;
int main() {
    string alphabet;
    int number, total;

    cout << "Enter an integer: ";
    cin >> number;

    do {
        total = 0;
        while (number > 0) {
            total += number % 10;
            number = (int)(number / 10);
        }

        if (total > 26) {
            number = total;
        }
    }
```

```

    } while (total > 26);

    alphabet = "ABCDEFGHIJKLMNOPQRSTUVWXYZ";
    cout << "The name of the person who might be thinking ";
    cout << "of you starts with a(an): " << alphabet[total - 1] << endl;
    return 0;
}

```

12. Solution

```

#include <iostream>
#include <cmath>
using namespace std;
int main() {
    int x, y;

    for (x = -100; x <= 100; x++) {
        for (y = -100; y <= 100; y++) {
            if (5 * x + 3 * pow(y, 2) == 0) {
                cout << x << ", " << y << endl;
            }
        }
    }
    return 0;
}

```

13. Solution

```

#include <iostream>
#include <cmath>
using namespace std;
int main() {
    int x, y, z;

    for (x = -10; x <= 10; x++) {
        for (y = -10; y <= 10; y++) {
            for (z = -10; z <= 10; z++) {
                if ((x + y) / 2.0 + 3.0 * pow(z, 2) / (x + 3 * y + 45) == x / 3.0) {
                    cout << x << ", " << y << ", " << z << endl;
                }
            }
        }
    }
    return 0;
}

```

14. Solution

```

#include <iostream>
using namespace std;
int main() {
    int m1, m2, m3, s;

```

```
cin >> m1 >> m2 >> m3;

s = 0;
while (m2 != 0) {
    if (m2 % 2 != 0) {
        s += m1;
    }
    m1 *= 2;
    m2 = (int)(m2 / 2);
}

m1 = s;
m2 = m3;

s = 0;
while (m2 != 0) {
    if (m2 % 2 != 0) {
        s += m1;
    }
    m1 *= 2;
    m2 = (int)(m2 / 2);
}

cout << s << endl;
return 0;
}
```

15. Solution

```
#include <iostream>
using namespace std;
int main() {
    double a;
    int x, numberOfDivisors, i;

    cin >> a;
    while (a <= 0 || a != (int)a) {
        cout << "Error! You must enter a positive integer" << endl;
        cin >> a;
    }
    x = (int)a;

    numberOfDivisors = 2;
    for (i = 2; i <= (int)(x / 2); i++) {
        if (x % i == 0) {
            numberOfDivisors++;
        }
    }
    cout << numberOfDivisors << endl;
    return 0;
}
```

16. Solution

```
#include <iostream>
using namespace std;
int main() {
    int x, numberOfDivisors, i;

    cout << "Enter an integer greater than 1: ";
    cin >> x;
    while (x <= 1) {
        cout << "Error!" << endl;
        cout << "Enter an integer greater than 1: ";
        cin >> x;
    }

    numberOfDivisors = 2;
    for (i = 2; i <= (int)(x / 2); i++) {
        if (x % i == 0) {
            numberOfDivisors++;
            break;
        }
    }

    if (numberOfDivisors == 2) {
        cout << "Number " << x << " is prime" << endl;
    }
    return 0;
}
```

17. Solution

```
#include <iostream>
#include <cmath>
using namespace std;
int main() {
    int start, finish, c, x, y;
    double z;

    cout << "Enter an positive integer: ";
    cin >> start;
    cout << "Enter a second positive integer: ";
    cin >> finish;

    if (start > finish) {
        c = start;
        start = finish;
        finish = c;
    }

    for (x = start; x <= finish; x++) {
        for (y = x; y <= finish; y++) {
            z = sqrt(x * x + y * y);
```

```

    //If result is integer and less than or equal to finish, display x, y, z
    if (z == (int)z && z <= finish) {
        cout << x << " " << y << " " << z << endl;
    }
}
}
return 0;
}

```

18. Solution

```

#include <iostream>
using namespace std;
int main() {
    int a, b, c, i, numberOfDivisors, x;

    cout << "Enter an integer greater than 1: ";
    cin >> a;
    while (a < 2) {
        cout << "Wrong number. Please enter an integer greater than 1: ";
        cin >> a;
    }

    cout << "Enter a second integer greater than 1: ";
    cin >> b;
    while (b < 2) {
        cout << "Wrong number. Please enter a second integer greater than 1: ";
        cin >> b;
    }

    if (a > b) {
        c = a;
        a = b;
        b = c;
    }

    for (x = a; x <= b; x++) {
        numberOfDivisors = 2;
        i = 2;
        while (i <= (int)(x / 2) && numberOfDivisors == 2) {
            if (x % i == 0) {
                numberOfDivisors++;
            }
            i++;
        }
        if (numberOfDivisors == 2) {
            cout << "Number " << x << " is prime" << endl;
        }
    }
    return 0;
}

```

19. Solution

```
#include <iostream>
using namespace std;
int main() {
    double y;
    int x, i, total;

    cout << "Enter a positive integer: ";
    cin >> y;
    while (y <= 0 || y != (int)y) {
        cout << "Wrong number! Enter a positive integer: ";
        cin >> y;
    }
    x = (int)y;

    total = 0;
    for (i = 1; i <= x - 1; i++) {
        if (x % i == 0) {
            total += i;
        }
    }

    if (total == x) {
        cout << "Number " << x << " is a perfect number" << endl;
    }
    else {
        cout << "Number " << x << " is not a perfect number" << endl;
    }
    return 0;
}
```

20. Solution

```
#include <iostream>
using namespace std;
int main() {
    double y;
    int a, b, c, x, j, total;

    cout << "Enter a positive integer: ";
    cin >> y;
    while (y <= 0 || y != (int)y) {
        cout << "Wrong number! Enter a positive integer: ";
        cin >> y;
    }
    a = (int)y;

    cout << "Enter a second positive integer: ";
    cin >> y;
    while (y <= 0 || y != (int)y) {
        cout << "Wrong number! Enter a second positive integer: ";
    }
}
```



```

    cin >> y;
}
b = (int)y;

if (a > b) {
    c = a;
    a = b;
    b = c;
}

for (x = a; x <= b + 1 - 1; x++) {
    total = 0;
    for (j = 1; j <= x - 1; j++) {
        if (x % j == 0) {
            total += j;
        }
    }

    if (total == x) {
        cout << "Number " << x << " is a perfect number" << endl;
    }
}
return 0;
}

```

21. Solution

```

#include <iostream>
using namespace std;
int main() {
    int a, b, c, d1, d2, d3, d4, r, x;

    cout << "Enter a positive four-digit integer: ";
    cin >> a;
    while (a < 1000 || a > 9999) {
        cout << "Wrong number. Please enter a positive four-digit integer: ";
        cin >> a;
    }

    cout << "Enter a second positive four-digit integer: ";
    cin >> b;
    while (b < 1000 || b > 9999) {
        cout << "Wrong number. Please enter a second positive four-digit integer: ";
        cin >> b;
    }

    if (a > b) {
        c = a;
        a = b;
        b = c;
    }

    for (x = a; x <= b; x++) {

```

```
d4 = x % 10;
r = (int)(x / 10);
d3 = r % 10;
r = (int)(r / 10);
d2 = r % 10;
d1 = (int)(r / 10);

if (d1 == d4 && d2 == d3) {
    cout << x << endl;
}
}
return 0;
}
```

22. Solution

```
#include <iostream>
#include <cmath>
using namespace std;
int main() {
    int i;

    for (i = 0; i <= 30; i++) {
        cout << pow(2, i) << endl;
    }
    return 0;
}
```

23. Solution

```
#include <iostream>
using namespace std;
int main() {
    int i, offset;

    offset = 10;
    i = 1;
    while (i <= 401) {
        cout << i << endl;
        i += offset;
        offset += 2;
    }
    return 0;
}
```

24. Solution

```
#include <iostream>
using namespace std;
int main() {
    int i;

    for (i = 1; i <= 100; i++) {
```

```
    cout << -i << "\n" << i << endl;
}
return 0;
}
```

25. Solution

First approach

```
#include <iostream>
#include <cmath>
using namespace std;
int main() {
    int i, offset, value;

    value = 0;
    for (i = 1; i <= 8; i++) {
        offset = pow(10, i - 1);
        value += offset;
        cout << value << endl;
    }
    return 0;
}
```

Second approach

```
#include <iostream>
using namespace std;
int main() {
    int i;
    string value;

    value = "1";
    for (i = 1; i <= 8; i++) {
        cout << value << endl;
        value += "1";
    }
    return 0;
}
```

26. Solution

```
#include <iostream>
using namespace std;
int main() {
    int a, fib, fibPrevious, fibPrevious2, i;

    cin >> a;

    fibPrevious2 = 0;
    fibPrevious = 1;
    fib = 1;
    for (i = 1; i <= a; i++) {
        cout << fibPrevious2 << endl;
        fib = fibPrevious + fibPrevious2;
```

```
        fibPrevious2 = fibPrevious;
        fibPrevious = fib;
    }
    return 0;
}
```

27. Solution

```
#include <iostream>
using namespace std;
int main() {
    int a, fib, fibPrev, fibPrevPrev;

    cin >> a;

    fibPrevPrev = 0;
    fibPrev = 1;
    fib = 1;
    while (fib < a) {
        cout << fib << endl;
        fib = fibPrev + fibPrevPrev;
        fibPrevPrev = fibPrev;
        fibPrev = fib;
    }
    return 0;
}
```

28. Solution

```
#include <iostream>
using namespace std;
int main() {
    int denominator, i, n, nominator;
    double y;

    cout << "Enter a positive integer: ";
    cin >> n;
    while (n <= 0) {
        cout << "Wrong number. Please enter a positive integer: ";
        cin >> n;
    }

    nominator = 0;
    for (i = 2; i <= 2 * n; i += 2) {
        nominator += i;
    }

    denominator = 1;
    for (i = 1; i <= n; i++) {
        denominator *= i;
    }

    y = nominator / (double)denominator;
}
```

```
    cout << y << endl;
    return 0;
}
```

29. Solution

```
#include <iostream>
using namespace std;
int main() {
    int i, n, nominator, sign;
    double y;

    cout << "Enter a positive integer: ";
    cin >> n;
    while (n <= 0) {
        cout << "Wrong number. Please enter a positive integer: ";
        cin >> n;
    }

    nominator = 0;
    sign = 1;
    for (i = 1; i <= 2 * n + 1; i += 2) {
        nominator += sign * i;
        sign = -sign;
    }

    y = nominator / (double)n;
    cout << y << endl;
    return 0;
}
```

30. Solution

```
#include <iostream>
using namespace std;
int main() {
    int i, n, sign;
    double y;

    cout << "Enter an integer greater than 2: ";
    cin >> n;
    while (n <= 2) {
        cout << "Wrong number. Please enter an integer greater than 2: ";
        cin >> n;
    }

    y = 0.5; //This is equal to the first two terms: 1 - 1 / 2

    sign = 1;
    for (i = 3; i <= n; i += 2) {
        y += sign / (double)i;
        sign = -sign;
    }
}
```

```
    cout << y << endl;
    return 0;
}
```

31. Solution

```
#include <iostream>
#include <cmath>
using namespace std;
int main() {
    int i, n;
    double y;

    cout << "Enter a positive integer: ";
    cin >> n;
    while (n <= 0) {
        cout << "Wrong number. Please enter a positive integer: ";
        cin >> n;
    }

    y = 0;
    for (i = 1; i <= n; i++) {
        y += 1 / pow(i, n - i + 1);
    }

    cout << y << endl;
    return 0;
}
```


32. Solution

```
#include <iostream>
using namespace std;
int main() {
    int factorial, i, n;

    cout << "Enter a non-negative integer: ";
    cin >> n;

    factorial = 1;
    for (i = 1; i <= n; i++) {
        factorial *= i;
    }

    cout << factorial << endl;
    return 0;
}
```

 Please note that this code operates properly for all non-negative integers, including zero.

33. Solution

First approach

```
#include <iostream>
#include <cmath>
using namespace std;
const double ACCURACY = 0.00001;

int main() {
    int i, j;
    double factorial, exponentialPrevious, exponential, x;

    cin >> x;

    exponential = 0;
    i = 0;
    do {
        exponentialPrevious = exponential;

        factorial = 1;
        for (j = 1; j <= i; j++) {
            factorial *= j;
        }

        exponential += pow(x, i) / factorial;

        i++;
    } while (abs(exponential - exponentialPrevious) > ACCURACY);

    cout << "e(" << x << ") ~= " << exponential << endl;
    return 0;
}
```

Second approach

```
#include <iostream>
#include <cmath>
using namespace std;
const double ACCURACY = 0.00001;

int main() {
    int i;
    double factorial, exponentialPrevious, exponential, x;

    cin >> x;

    exponential = 1;
    i = 1;
    factorial = 1;
    do {
        exponentialPrevious = exponential;

        factorial *= i;

        exponential += pow(x, i) / factorial;
    } while (abs(exponential - exponentialPrevious) > ACCURACY);

    cout << "e(" << x << ") ~= " << exponential << endl;
    return 0;
}
```

```

    i++;
} while (abs(exponential - exponentialPrevious) > ACCURACY);

cout << "e(" << x << ") ~= " << exponential << endl;
return 0;
}

```

34. Solution

First approach

```

#include <iostream>
#include <cmath>
using namespace std;
const double ACCURACY = 0.00001;

int main() {
    int i, j, sign;
    double factorial;
    double sinus, sinusPrevious, x;

    cin >> x;

    sign = 1;
    sinus = 0;
    i = 1;
    do {
        sinusPrevious = sinus;

        factorial = 1;
        for (j = 1; j <= i; j++) {
            factorial *= j;
        }

        sinus += sign * pow(x, i) / factorial;

        sign = -sign;
        i += 2;
    } while (abs(sinus - sinusPrevious) > ACCURACY);

    cout << "sin(" << x << ") ~= " << sinus << endl;
    return 0;
}

```

Second approach

```

#include <iostream>
#include <cmath>
using namespace std;
const double ACCURACY = 0.00001;

int main() {
    int i, sign;
    double factorial;
    double sinus, sinusPrevious, x;

```



```

cin >> x;

sign = -1;
sinus = x;
i = 3;
factorial = 1;
do {
    sinusPrevious = sinus;

    factorial *= i * (i - 1);

    sinus += sign * pow(x, i) / factorial;

    sign = -sign;
    i += 2;
} while (abs(sinus - sinusPrevious) > ACCURACY);

cout << "sin(" << x << ") ~= " << sinus << endl;
return 0;
}

```

35. Solution

First approach

```

#include <iostream>
#include <cmath>
using namespace std;
const double ACCURACY = 0.00001;

int main() {
    int i, j, sign;
    double factorial;
    double cosinus, cosinusPrevious, x;

    cin >> x;

    sign = 1;
    cosinus = 0;
    i = 0;
    do {
        cosinusPrevious = cosinus;

        factorial = 1;
        for (j = 1; j <= i; j++) {
            factorial *= j;
        }

        cosinus += sign * pow(x, i) / factorial;

        sign = -sign;
        i += 2;
    } while (abs(cosinus - cosinusPrevious) > ACCURACY);
}

```

```

    cout << "cos(" << x << ") ~= " << cosinus << endl;
    return 0;
}

```

Second approach

```

#include <iostream>
#include <cmath>
using namespace std;
const double ACCURACY = 0.00001;

int main() {
    int i, sign;
    double factorial;
    double cosinus, cosinusPrevious, x;

    cin >> x;

    sign = -1;
    cosinus = 1;
    i = 2;
    factorial = 1;
    do {
        cosinusPrevious = cosinus;

        factorial *= i * (i - 1);

        cosinus += sign * pow(x, i) / factorial;

        sign = -sign;
        i += 2;
    } while (abs(cosinus - cosinusPrevious) > ACCURACY);

    cout << "cos(" << x << ") ~= " << cosinus << endl;
    return 0;
}

```

36. Solution

```

#include <iostream>
using namespace std;
int main() {
    int a, b, c, i;
    bool failure;

    string alphabet = "abcdefghijklmnopqrstuvwxy";

    do {
        cout << "Enter an integer between 1 and 26: ";
        cin >> a;

        failure = false;
        if (a < 1) {
            cout << "Please enter positive integers!" << endl;
            failure = true;

```

```
    }  
    else if (a > 26) {  
        cout << "Please enter a value less than or equal to 26!" << endl;  
        failure = true;  
    }  
} while (failure);  
  
do {  
    cout << "Enter an integer between 1 and 26: ";  
    cin >> b;  
  
    failure = false;  
    if (b < 1) {  
        cout << "Please enter positive integers!" << endl;  
        failure = true;  
    }  
    else if (b > 26) {  
        cout << "Please enter a value less than or equal to 26!" << endl;  
        failure = true;  
    }  
} while (failure);  
  
if (a > b) {  
    c = a;  
    a = b;  
    b = c;  
}  
  
for (i = a; i <= b; i++) {  
    cout << alphabet[i - 1];  
}  
return 0;  
}
```

37. Solution

```
#include <iostream>  
#include <ctime>  
#include <cstdlib>  
using namespace std;  
int main() {  
    int attempts, guess, secretNumber;  
  
    srand(time(NULL));  
  
    secretNumber = 1 + rand() % 100;  
  
    attempts = 1;  
    cout << "Enter a guess: ";  
    cin >> guess;  
    while (guess != secretNumber) {  
        if (guess > secretNumber) {  
            cout << "Your guess is bigger than my secret number. Try again." << endl;  
        }  
    }  
}
```

```
    }  
    else {  
        cout << "Your guess is smaller than my secret number. Try again." << endl;  
    }  
    attempts++;  
    cout << "Enter a guess: ";  
    cin >> guess;  
}  
cout << "You found it!" << endl;  
cout << "Attempts: " << attempts << endl;  
return 0;  
}
```

38. Solution

```
#include <iostream>  
#include <ctime>  
#include <cstdlib>  
using namespace std;  
int main() {  
    int attempts = 0, firstPlayerAttempts = 0, guess, i, secretNumber;  
  
    srand(time(NULL));  
  
    for (i = 1; i <= 2; i++) {  
        secretNumber = 1 + rand() % 100;  
  
        attempts = 1;  
        cout << "Enter a guess: ";  
        cin >> guess;  
        while (guess != secretNumber) {  
            if (guess > secretNumber) {  
                cout << "Your guess is bigger than my secret number. Try again." << endl;  
            }  
            else {  
                cout << "Your guess is smaller than my secret number. Try again." << endl;  
            }  
            attempts++;  
            cout << "Enter a guess: ";  
            cin >> guess;  
        }  
        cout << "You found it!" << endl;  
        cout << "Attempts: " << attempts << endl;  
  
        if (i == 1) {  
            firstPlayerAttempts = attempts;  
        }  
    }  
  
    if (firstPlayerAttempts < attempts) {  
        cout << "First player wins!" << endl;  
    }  
}
```

```

else if (firstPlayerAttempts > attempts) {
    cout << "Second player wins!" << endl;
}
else {
    cout << "It's a draw" << endl;
}
return 0;
}

```

39. Solution

```

#include <iostream>
using namespace std;
int main() {
    int choice, diagonal;

    do {
        cout << "1. 4/3 TV Screen" << endl;
        cout << "2. 16/9 TV Screen" << endl;
        cout << "3. Exit" << endl;
        cout << "Enter a choice: ";
        cin >> choice;

        if (choice == 1) {
            cout << "Enter diagonal: ";
            cin >> diagonal;
            cout << "Width: " << diagonal * 0.8 << endl;
            cout << "Height: " << diagonal * 0.6 << endl;
        }
        else if (choice == 2) {
            cout << "Enter diagonal: ";
            cin >> diagonal;
            cout << "Width: " << diagonal * 0.87 << endl;
            cout << "Height: " << diagonal * 0.49 << endl;
        }
    } while (choice != 3);
    return 0;
}

```

40. Solution

```

#include <iostream>
#include <boost/algorithm/string.hpp>
using namespace boost::algorithm;
using namespace std;
int main() {
    int countA, countABoys, countB, countCdefGirls, grade;
    int i, maximum, minimum, n, total, totalA, totalABoys, totalB;
    string gender;

    cout << "Enter total number of students: ";
    cin >> n;
}

```

```
while (n <= 0) {
    cout << "Wrong number. Please enter total number of students: ";
    cin >> n;
}

total = 0;
totalA = 0;
countA = 0;
totalB = 0;
countB = 0;
totalABoys = 0;
countABoys = 0;
countCdefGirls = 0;

maximum = -1;
minimum = 101;

for (i = 1; i <= n; i++) {
    cout << "Enter grade for student No " << i << ": ";
    cin >> grade;
    while (grade < 0 || grade > 100) {
        cout << "Wrong grade. Please enter grade for student No " << i << ": ";
        cin >> grade;
    }

    cout << "Enter gender for student No " << i << ": ";
    cin >> gender;
    gender = to_upper_copy(gender);
    while (gender != "M" && gender != "F" && gender != "O") {
        cout << "Wrong gender. Please enter gender for student No " << i << ": ";
        cin >> gender;
        gender = to_upper_copy(gender);
    }

    if (grade >= 90 && grade <= 100) {
        totalA += grade;
        countA++;
        if (gender == "M") {
            totalABoys += grade;
            countABoys++;
        }
    }
    else if (grade >= 80 && grade <= 89) {
        totalB += grade;
        countB++;
    }
    else {
        if (gender == "F") {
            countCdefGirls++;
        }
    }
}
```

```

    if (grade > maximum) {
        maximum = grade;
    }

    if (grade < minimum) {
        minimum = grade;
    }

    total += grade;
}

if (countA > 0) {
    cout << "The average value of those who got an 'A' is: ";
    cout << totalA / (double)countA << endl;
}
if (countB > 0) {
    cout << "The average value of those who got a 'B' is: ";
    cout << totalB / (double)countB << endl;
}
if (countABoys > 0) {
    cout << "The average value of boys who got an 'A' is: ";
    cout << totalABoys / (double)countABoys << endl;
}
cout << "The total number of girls that got less than 'B' is: " << countCdefGirls << endl;
cout << "The highest grade is: " << maximum << endl;
cout << "The lowest grade is: " << minimum << endl;
cout << "The average grade of the whole class is: " << total / (double)n << endl;
return 0;
}

```

41. Solution

```

#include <iostream>
#include <boost/algorithm/string.hpp>
using namespace boost::algorithm;
using namespace std;
int main() {
    double amount, discount;
    string answer;

    do {
        cout << "Enter amount: ";
        cin >> amount;
        while (amount <= 0) {
            cout << "Wrong amount. Please enter amount: ";
            cin >> amount;
        }

        if (amount < 20) {
            discount = 0;
        }
        else if (amount < 50) {

```

```

    discount = 3;
}
else if (amount < 100) {
    discount = 5;
}
else {
    discount = 10;
}

cout << "Discount: " << discount << "%" << endl;
cout << "Amount to pay (discount included): " << amount - amount * discount / 100 << endl;

cout << "Would you like to repeat? ";
cin >> answer;
answer = to_upper_copy(answer);
} while (answer == "YES");
return 0;
}

```

42. Solution

```

#include <iostream>
using namespace std;
const double TAX_RATE = 0.25;

int main() {
    int kwh;
    double t;

    cout << "Enter number of Kilowatt-hours consumed: ";
    cin >> kwh;
    while (kwh < 0 && kwh != -1) {
        cout << "Wrong value. Please enter number of Kilowatt-hours consumed: ";
        cin >> kwh;
    }

    while (kwh != -1) {
        if (kwh <= 400) {
            t = kwh * 0.11;
        }
        else if (kwh <= 1500) {
            t = 400 * 0.11 + (kwh - 400) * 0.22;
        }
        else if (kwh <= 3500) {
            t = 400 * 0.11 + 1100 * 0.22 + (kwh - 1500) * 0.25;
        }
        else {
            t = 400 * 0.11 + 1100 * 0.22 + 2000 * 0.25 + (kwh - 3500) * 0.50;
        }

        t += t * TAX_RATE;
        cout << "Total amount to pay (taxes included): " << t << endl;
    }
}

```

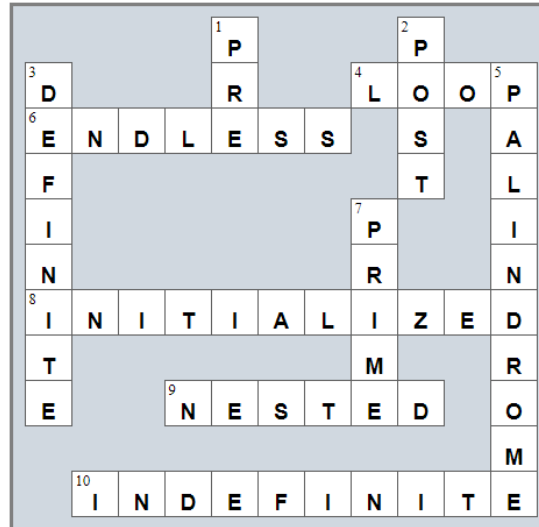


```
    cout << "Enter number of Kilowatt-hours consumed: ";
    cin >> kwh;
    while (kwh < 0 && kwh != -1) {
        cout << "Wrong value. Please enter number of Kilowatt-hours consumed: ";
        cin >> kwh;
    }
}
return 0;
}
```

Review in "Loop Control Structures"

Review Crossword Puzzle

1.



Chapter 31

31.13 Review Questions: True/False

- | | |
|-----------|-----------|
| 1. true | 21. true |
| 2. true | 22. false |
| 3. false | 23. true |
| 4. false | 24. false |
| 5. false | 25. true |
| 6. true | 26. false |
| 7. false | 27. false |
| 8. true | 28. true |
| 9. false | 29. false |
| 10. true | 30. true |
| 11. true | 31. true |
| 12. true | 32. false |
| 13. false | 33. false |
| 14. false | 34. true |
| 15. false | 35. true |
| 16. true | 36. true |
| 17. false | 37. false |
| 18. true | 38. false |
| 19. true | 39. true |
| 20. false | |

31.14 Review Questions: Multiple Choice

- | | |
|------|-------|
| 1. b | 8. d |
| 2. a | 9. c |
| 3. c | 10. a |
| 4. b | 11. b |
| 5. d | 12. a |
| 6. b | 13. b |
| 7. d | 14. b |

31.15 Review Exercises

1. Solution

weights =	170	0	}	<i>People</i>
	190	1		
	193	2		
	165	3		
	200	4		

2. Solution

names =	John Thompson	weights =	170	0	} People
	Chloe Brown		190	1	
	Ryan Miller		193	2	
	Antony Harris		165	3	
	Alexander Lewis		200	4	
	Samantha Clark		170	5	
	Ava Parker		172	6	

3. Solution

names =	Toba	areas =	Months			} Lakes	
	Issyk Kul		0	1	2		
	Baikal		440	438	437		0
	Crater		2408	2405	2402		1
	Karakul		12248	12247	12240		2
			21	20	18	3	
			150	145	142	4	

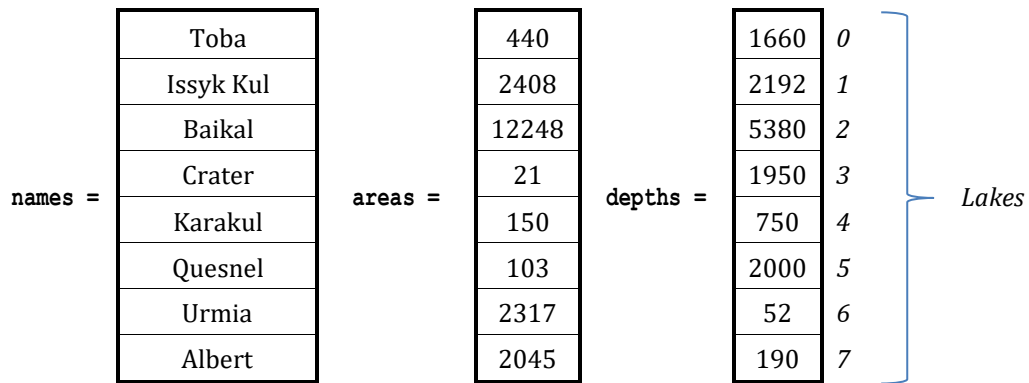
June July August

4. Solution

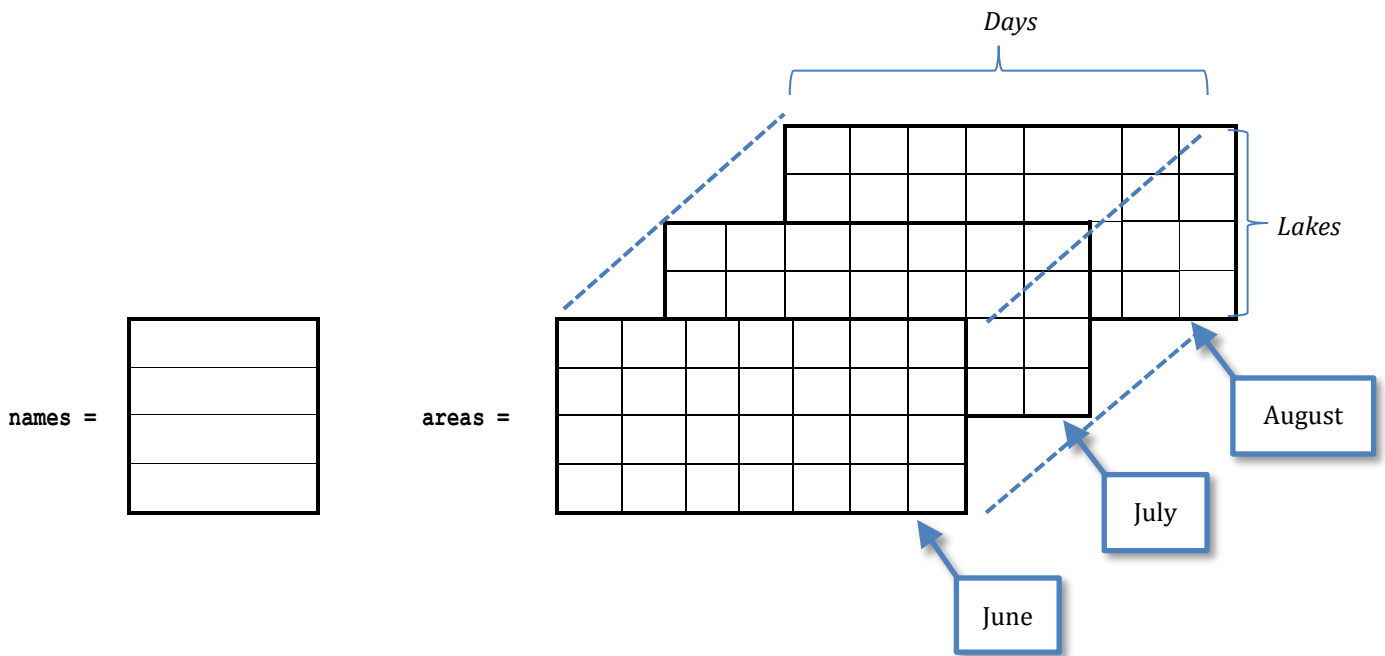
boxes =	Dimensions			} Boxes	
	0	1	2		
	10	31	15		0
	15	12	17		1
	22	10	18		2
	22	20	12		3
	26	25	14		4
	66	26	21		5
	54	34	24		6
	64	28	22		7
34	12	18	8		
33	10	10	9		

Width Height Length

5. Solution



6. Solution



7. Solution

Step	Statement	x	a[0]	a[1]	a[2]
1	int a[3]	?	?	?	?
2	a[2] = 1	?	?	?	1
3	x = 0	0	?	?	1
4	a[x + a[2]] = 4	0	?	4	1
5	a[x] = a[x + 1] * 4	0	16	4	1

8. Solution

Step	Statement	x	a[0]	a[1]	a[2]	a[3]	a[4]
1	int a[5]	?	?	?	?	?	?
2	a[1] = 5	?	?	5	?	?	?
3	x = 0	0	?	5	?	?	?
4	a[x] = 4	0	4	5	?	?	?
5	a[a[0]] = a[x + 1] % 3	0	4	5	?	?	2
6	a[a[0] / 2] = 10	0	4	5	10	?	2
7	x += 2	2	4	5	10	?	2
8	a[x + 1] = a[x] + 9	2	4	5	10	19	2

9. Solution

For input value of 3

Step	Statement	x	a[0]	a[1]	a[2]	a[3]
1	int a[4]	?	?	?	?	?
2	cin >> a[1]	?	?	3	?	?
3	x = 0	0	?	3	?	?
4	a[x] = 3	0	3	3	?	?
5	a[a[0]] = a[x + 1] % 2	0	3	3	?	1
6	a[a[0] % 2] = 10	0	3	10	?	1
7	x++	1	3	10	?	1
8	a[x + 1] = a[x] + 9	1	3	10	19	1

For input value of 4

Step	Statement	x	a[0]	a[1]	a[2]	a[3]
1	int a[4]	?	?	?	?	?
2	cin >> a[1]	?	?	4	?	?
3	x = 0	0	?	4	?	?
4	a[x] = 3	0	3	4	?	?
5	a[a[0]] = a[x + 1] % 2	0	3	4	?	0
6	a[a[0] % 2] = 10	0	3	10	?	0
7	x++	1	3	10	?	0
8	a[x + 1] = a[x] + 9	1	3	10	19	0

For input value of 1

Step	Statement	x	a[0]	a[1]	a[2]	a[3]
1	int a[4]	?	?	?	?	?
2	cin >> a[1]	?	?	1	?	?
3	x = 0	0	?	1	?	?

4	$a[x] = 3$	0	3	1	?	?
5	$a[a[0]] = a[x + 1] \% 2$	0	3	1	?	3
6	$a[a[0] \% 2] = 10$	0	3	10	?	3
7	$x++$	1	3	10	?	3
8	$a[x + 1] = a[x] + 9$	1	3	10	19	3

10. Solution

For input value of 100

Step	Statement	x	a[0]	a[1]	a[2]	a[3]
1	<code>int a[4]</code>	?	?	?	?	?
2	<code>cin >> a[1]</code>	?	?	100	?	?
3	<code>x = 0</code>	0	?	100	?	?
4	$a[x] = 3$	0	3	100	?	?
5	$a[a[0]] = a[x + 1] \% 10$	0	3	100	?	0
6	<code>if (a[3] > 5)</code>	false				
7	<code>a[2] = 3</code>	0	3	100	3	0

For input value of 108

Step	Statement	x	a[0]	a[1]	a[2]	a[3]
1	<code>int a[4]</code>	?	?	?	?	?
2	<code>cin >> a[1]</code>	?	?	108	?	?
3	<code>x = 0</code>	0	?	108	?	?
4	$a[x] = 3$	0	3	108	?	?
5	$a[a[0]] = a[x + 1] \% 10$	0	3	108	?	8
6	<code>if (a[3] > 5)</code>	true				
7	$a[a[0] \% 2] = 9$	0	3	9	?	8
8	<code>x += 1</code>	1	3	9	?	8
9	$a[x + 1] = a[x] + 9$	1	3	9	18	8

For input value of 1

Step	Statement	x	a[0]	a[1]	a[2]	a[3]
1	<code>int a[4]</code>	?	?	?	?	?
2	<code>cin >> a[1]</code>	?	?	1	?	?
3	<code>x = 0</code>	0	?	1	?	?
4	$a[x] = 3$	0	3	1	?	?
5	$a[a[0]] = a[x + 1] \% 10$	0	3	1	?	1
6	<code>if (a[3] > 5)</code>	false				
7	<code>a[2] = 3</code>	0	3	1	3	1

11. Solution

Step	Statement	x	y	a[0]	a[1]	a[2]
1	int a[3]	?	?	?	?	?
2	x = 4	4	?	?	?	?
3	y = x - 1	4	3	?	?	?
4, 5	if (x > y) a[0] = 1; else a[0] = y;	4	3	1	?	?
6	a[1] = x + 3	4	3	1	7	?
7	y = y - 1	4	2	1	7	?
8	a[y] = (x + 5) % 2	4	2	1	7	1

12. Solution

Step	Statement	i	a[0]	a[1]	a[2]	a[3]	a[4]	a[5]
1	int a[] = {17, 12, 45, 12, 12, 49}	?	17	12	45	12	12	49
2	i = 0	0	17	12	45	12	12	49
3	i <= 5				true			
4	if (a[i] == 12)				false			
5	a[i]++	0	18	12	45	12	12	49
6	i++	1	18	12	45	12	12	49
7	i <= 5				true			
8	if (a[i] == 12)				true			
9	a[i]--	1	18	11	45	12	12	49
10	i++	2	18	11	45	12	12	49
11	i <= 5				true			
12	if (a[i] == 12)				false			
13	a[i]++	2	18	11	46	12	12	49
14	i++	3	18	11	46	12	12	49
15	i <= 5				true			
16	if (a[i] == 12)				true			
17	a[i]--	3	18	11	46	11	12	49
18	i++	4	18	11	46	11	12	49
19	i <= 5				true			
20	if (a[i] == 12)				true			
21	a[i]--	4	18	11	46	11	11	49
22	i++	5	18	11	46	11	11	49
23	i <= 5				true			

24	if (a[i] == 12)	false						
25	a[i]++	5	18	11	46	11	11	50
26	i++	6	18	11	46	11	11	50
27	i <= 5	false						

13. Solution

Step	Statement	i	a[0]	a[1]	a[2]	a[3]	a[4]	a[5]
1	int a[] = {10, 15, 12, 23, 22, 19}	?	10	15	12	23	22	19
2	i = 1	1	10	15	12	23	22	19
3	i <= 4	true						
4	a[i] = a[i + 1] + a[i - 1]	1	10	22	12	23	22	19
5	i++	2	10	22	12	23	22	19
6	i <= 4	true						
7	a[i] = a[i + 1] + a[i - 1]	2	10	22	45	23	22	19
8	i++	3	10	22	45	23	22	19
9	i <= 4	true						
10	a[i] = a[i + 1] + a[i - 1]	3	10	22	45	67	22	19
11	i++	4	10	22	45	67	22	19
12	i <= 4	true						
13	a[i] = a[i + 1] + a[i - 1]	4	10	22	45	67	86	19
14	i++	5	10	22	45	67	86	19
15	i <= 4	false						

14. Solution

It displays:

Navajo

Cherokee

Sioux

15. Solution

```
#include <iostream>
#include <cmath>
using namespace std;
const int ELEMENTS = 100;

int main() {
    int i;

    double a[ELEMENTS];
    for (i = 0; i <= ELEMENTS - 1; i++) {
        cin >> a[i];
    }
}
```

```
    for (i = 0; i <= ELEMENTS - 1; i++) {
        cout << pow(a[i], 3) << endl;
    }
    return 0;
}
```

16. Solution

```
#include <iostream>
#include <cmath>
using namespace std;
const int ELEMENTS = 80;

int main() {
    int i;

    double a[ELEMENTS];
    for (i = 0; i <= ELEMENTS - 1; i++) {
        cin >> a[i];
    }

    for (i = 0; i <= ELEMENTS - 1; i++) {
        a[i] = pow(a[i], 2);
    }

    for (i = ELEMENTS - 1; i >= 0; i--) {
        cout << a[i] << endl;
    }
    return 0;
}
```

17. Solution

```
#include <iostream>
using namespace std;
const int ELEMENTS = 90;

int main() {
    int i;

    int a[ELEMENTS];
    for (i = 0; i <= ELEMENTS - 1; i++) {
        cin >> a[i];
    }

    for (i = ELEMENTS - 1; i >= 0; i--) {
        if (a[i] % 5 == 0) {
            cout << a[i] << endl;
        }
    }
    return 0;
}
```

18. Solution

```
#include <iostream>
using namespace std;
const int ELEMENTS = 50;

int main() {
    int i;

    int a[ELEMENTS];
    for (i = 0; i <= ELEMENTS - 1; i++) {
        cin >> a[i];
    }

    for (i = 0; i <= ELEMENTS - 1; i++) {
        if (a[i] % 2 == 0 || a[i] > 10) {
            cout << a[i] << endl;
        }
    }
    return 0;
}
```

19. Solution

```
#include <iostream>
using namespace std;
const int ELEMENTS = 30;

int main() {
    int i;
    double total;

    double a[ELEMENTS];
    for (i = 0; i <= ELEMENTS - 1; i++) {
        cin >> a[i];
    }

    total = 0;
    for (i = 0; i <= ELEMENTS - 1; i++) {
        if (a[i] > 0) {
            total += a[i];
        }
    }
    cout << total << endl;
    return 0;
}
```

20. Solution

```
#include <iostream>
using namespace std;
const int ELEMENTS = 50;

int main() {
```

```

int i, total;

int a[ELEMENTS];
for (i = 0; i <= ELEMENTS - 1; i++) {
    cin >> a[i];
}

total = 0;
for (i = 0; i <= ELEMENTS - 1; i++) {
    if (a[i] >= 10 && a[i] <= 99) {
        total += a[i];
    }
}
cout << total << endl;
return 0;
}

```

21. Solution

```

#include <iostream>
using namespace std;
const int ELEMENTS = 40;

int main() {
    int i;
    double sumNeg, sumPos;

    double a[ELEMENTS];
    for (i = 0; i <= ELEMENTS - 1; i++) {
        cin >> a[i];
    }

    sumPos = sumNeg = 0;
    for (i = 0; i <= ELEMENTS - 1; i++) {
        if (a[i] > 0 ) {
            sumPos += a[i];
        }
        else if (a[i] < 0) {
            sumNeg += a[i];
        }
    }
    cout << sumPos << ", " << sumNeg << endl;
    return 0;
}

```

22. Solution

```

#include <iostream>
using namespace std;
const int ELEMENTS = 20;

int main() {
    int i;
    double total;

```

```
double a[ELEMENTS];
for (i = 0; i <= ELEMENTS - 1; i++) {
    cin >> a[i];
}

total = 0;
for (i = 0; i <= ELEMENTS - 1; i++) {
    total += a[i];
}
cout << total / ELEMENTS << endl;
return 0;
}
```

23. Solution

```
#include <iostream>
using namespace std;
const int ELEMENTS = 50;

int main() {
    int i;

    int a[ELEMENTS];
    for (i = 0; i <= ELEMENTS - 1; i++) {
        cout << "Enter an integer: ";
        cin >> a[i];
    }

    for (i = 0; i <= ELEMENTS - 1; i++) {
        if (a[i] < 20) {
            cout << a[i] << endl;
        }
    }
    return 0;
}
```

24. Solution

```
#include <iostream>
using namespace std;
const int ELEMENTS = 60;

int main() {
    int i;

    double a[ELEMENTS];
    for (i = 0; i <= ELEMENTS - 1; i++) {
        cout << "Enter a number: ";
        cin >> a[i];
    }

    for (i = 0; i <= ELEMENTS - 1; i += 2) {
        cout << a[i] << endl;
    }
    return 0;
}
```

```
| }
```

25. Solution

```
#include <iostream>
using namespace std;
const int ELEMENTS = 20;

int main() {
    int i;
    double total;

    double a[ELEMENTS];
    for (i = 0; i <= ELEMENTS - 1; i++) {
        cout << "Enter a number: ";
        cin >> a[i];
    }

    total = 0;
    for (i = 0; i <= ELEMENTS - 1; i += 2) {
        total += a[i];
    }
    cout << total << endl;
    return 0;
}
```

26. Solution

```
#include <iostream>
using namespace std;
const int ELEMENTS = 100;

int main() {
    int i;
    int a[ELEMENTS];
    for (i = 0; i <= ELEMENTS - 1; i++) {
        a[i] = i + 1;
    }
    ...
}
```

27. Solution

First approach

```
#include <iostream>
using namespace std;
const int ELEMENTS = 100;

int main() {
    int i, k;
    int a[ELEMENTS];
    k = 2;
    for (i = 0; i <= ELEMENTS - 1; i++) {
        a[i] = k;
        k += 2;
    }
    ...
}
```

Second approach

```
#include <iostream>
using namespace std;
const int ELEMENTS = 100;

int main() {
    int i;
    int a[ELEMENTS];
    for (i = 0; i <= ELEMENTS - 1; i++) {
        a[i] = (i + 1) * 2;
    }
    ...
}
```

28. Solution

```
#include <iostream>
#include <cmath>
using namespace std;
int main() {
    int i, n;

    cout << "Enter N: ";
    cin >> n;
    while (n < 1) {
        cout << "Error! Value must be greater than or equal to 1" << endl;
        cout << "Enter N: ";
        cin >> n;
    }

    int a[n];

    for (i = 1; i <= n; i++) {
        a[i - 1] = pow(i, 2);
    }

    for (i = 0; i <= n - 1; i++) {
        cout << a[i] << endl;
    }
    return 0;
}
```

29. Solution

```
#include <iostream>
using namespace std;
const int ELEMENTS = 10;

int main() {
    int i;

    double a[ELEMENTS];
    for (i = 1; i <= ELEMENTS - 1; i++) {
        cout << "Enter a number: ";
        cin >> a[i];
    }
}
```

```
    for (i = 0; i <= ELEMENTS - 1; i++) {
        if (a[i] == (int)a[i]) {
            cout << i << endl;
        }
    }
    return 0;
}
```

30. Solution

```
#include <iostream>
using namespace std;
const int ELEMENTS = 50;

int main() {
    int i, count;

    double a[ELEMENTS];
    for (i = 1; i <= ELEMENTS - 1; i++) {
        cout << "Enter a number: ";
        cin >> a[i];
    }

    count = 0;
    for (i = 0; i <= ELEMENTS - 1; i++) {
        if (a[i] < 0) {
            count++;
        }
    }
    cout << count << endl;
    return 0;
}
```

31. Solution

```
#include <iostream>
using namespace std;
const int WORDS = 50;

int main() {
    int i;

    string a[WORDS];
    for (i = 0; i <= WORDS - 1; i++) {
        cin >> a[i];
    }

    for (i = 0; i <= WORDS - 1; i++) {
        if (a[i].length() >= 10 ) {
            cout << a[i] << endl;
        }
    }
    return 0;
}
```


32. Solution

```

#include <iostream>
using namespace std;
const int ELEMENTS = 30;

int main() {
    int i, k;

    string words[ELEMENTS];
    for (i = 0; i <= ELEMENTS - 1; i++) {
        cin >> words[i];
    }

    int lengthLimits[] = {0, 5, 10, 20};

    for (k = 1; k <= 3; k++) {
        for (i = 0; i <= ELEMENTS - 1; i++) {
            if (words[i].length() >= lengthLimits[k - 1] && words[i].length() < lengthLimits[k]) {
                cout << words[i] << endl;
            }
        }
    }
    return 0;
}

```

33. Solution

```

#include <iostream>
using namespace std;
const int WORDS = 40;

int main() {
    int count, i, j;

    string a[WORDS];
    for (i = 0; i <= WORDS - 1; i++) {
        cout << "Enter a word: ";
        cin >> a[i];
    }

    for (i = 0; i <= WORDS - 1; i++) {
        count = 0;
        for (j = 0; j <= a[i].length() - 1; j++) {
            if (a[i][j] == 'w') { //Alternatively use: if (a[i].substr(j, 1) == "w")
                count++;
            }
        }
        if (count >= 2) {
            cout << a[i] << endl;
        }
    }
    return 0;
}

```

34. Solution

```
#include <iostream>
#include <unordered_map>
using namespace std;
int main() {
    string roman;
    int number, digit1, digit2;
    cout << "Enter a number between 1 and 99: ";
    cin >> number;

    digit1 = (int)(number / 10);
    digit2 = number % 10;

    unordered_map<int, string> number2romanOnes = {
        {1, "I"}, {2, "II"}, {3, "III"}, {4, "IV"}, {5, "V"}, {6, "VI"}, {7, "VII"}, {8, "VIII"}, {9, "IX"}
    };

    unordered_map<int, string> number2romanTens = {
        {1, "X"}, {2, "XX"}, {3, "XXX"}, {4, "XL"}, {5, "L"}, {6, "LX"}, {7, "LXX"}, {8, "LXXX"}, {9, "XC"}
    };

    roman = number2romanTens[digit1] + number2romanOnes[digit2];
    cout << roman << endl;
    return 0;
}
```

Chapter 32

32.7 Review Questions: True/False

1. false
2. true
3. false
4. false
5. false
6. true
7. false
8. true
9. true
10. true
11. true
12. true
13. false
14. true
15. true
16. true
17. true
18. true
19. false
20. true
21. true
22. true
23. false
24. true
25. true
26. true
27. false

32.8 Review Questions: Multiple Choice

1. b
2. b
3. c
4. a
5. d
6. a
7. d
8. c
9. c
10. c
11. b

32.9 Review Exercises

1. Solution

Step	Statement	x	a						
1	int a[2][3]	?	<table border="1"> <tr><td>?</td><td>?</td><td>?</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> </table>	?	?	?	?	?	?
?	?	?							
?	?	?							
2	a[0][2] = 1	?	<table border="1"> <tr><td>?</td><td>?</td><td>1</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> </table>	?	?	1	?	?	?
?	?	1							
?	?	?							
3	x = 0	0	<table border="1"> <tr><td>?</td><td>?</td><td>1</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> </table>	?	?	1	?	?	?
?	?	1							
?	?	?							
4	a[0][x] = 9	0	<table border="1"> <tr><td>9</td><td>?</td><td>1</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> </table>	9	?	1	?	?	?
9	?	1							
?	?	?							
5	a[0][x + a[0][2]] = 4	0	<table border="1"> <tr><td>9</td><td>4</td><td>1</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> </table>	9	4	1	?	?	?
9	4	1							
?	?	?							
6	a[a[0][2]][2] = 19	0	<table border="1"> <tr><td>9</td><td>4</td><td>1</td></tr> <tr><td>?</td><td>?</td><td>19</td></tr> </table>	9	4	1	?	?	19
9	4	1							
?	?	19							
7	a[a[0][2]][x + 1] = 13	0	<table border="1"> <tr><td>9</td><td>4</td><td>1</td></tr> <tr><td>?</td><td>13</td><td>19</td></tr> </table>	9	4	1	?	13	19
9	4	1							
?	13	19							
8	a[a[0][2]][x] = 15	0	<table border="1"> <tr><td>9</td><td>4</td><td>1</td></tr> <tr><td>15</td><td>13</td><td>19</td></tr> </table>	9	4	1	15	13	19
9	4	1							
15	13	19							

2. Solution

Step	Statement	i	j	a						
1	int a[2][3]	?	?	<table border="1"> <tr><td>?</td><td>?</td><td>?</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> </table>	?	?	?	?	?	?
?	?	?								
?	?	?								
2	i = 0	0	?	<table border="1"> <tr><td>?</td><td>?</td><td>?</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> </table>	?	?	?	?	?	?
?	?	?								
?	?	?								
3	i <= 1	true								
4	j = 0	0	0	<table border="1"> <tr><td>?</td><td>?</td><td>?</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> </table>	?	?	?	?	?	?
?	?	?								
?	?	?								

5	<code>j <= 2</code>			true						
6	<code>a[i][j] = (i + 1) * 5 + j</code>	0	0	<table border="1"> <tr><td>5</td><td>?</td><td>?</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> </table>	5	?	?	?	?	?
5	?	?								
?	?	?								
7	<code>j++</code>	0	1	<table border="1"> <tr><td>5</td><td>?</td><td>?</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> </table>	5	?	?	?	?	?
5	?	?								
?	?	?								
8	<code>j <= 2</code>			true						
9	<code>a[i][j] = (i + 1) * 5 + j</code>	0	1	<table border="1"> <tr><td>5</td><td>6</td><td>?</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> </table>	5	6	?	?	?	?
5	6	?								
?	?	?								
10	<code>j++</code>	0	2	<table border="1"> <tr><td>5</td><td>6</td><td>?</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> </table>	5	6	?	?	?	?
5	6	?								
?	?	?								
11	<code>j <= 2</code>			true						
12	<code>a[i][j] = (i + 1) * 5 + j</code>	0	2	<table border="1"> <tr><td>5</td><td>6</td><td>7</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> </table>	5	6	7	?	?	?
5	6	7								
?	?	?								
13	<code>j++</code>	0	3	<table border="1"> <tr><td>5</td><td>6</td><td>7</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> </table>	5	6	7	?	?	?
5	6	7								
?	?	?								
14	<code>j <= 2</code>			false						
15	<code>i++</code>	1	3	<table border="1"> <tr><td>5</td><td>6</td><td>7</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> </table>	5	6	7	?	?	?
5	6	7								
?	?	?								
16	<code>i <= 1</code>			true						
17	<code>j = 0</code>	1	0	<table border="1"> <tr><td>5</td><td>6</td><td>7</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> </table>	5	6	7	?	?	?
5	6	7								
?	?	?								
18	<code>j <= 2</code>			true						
19	<code>a[i][j] = (i + 1) * 5 + j</code>	1	0	<table border="1"> <tr><td>5</td><td>6</td><td>7</td></tr> <tr><td>10</td><td>?</td><td>?</td></tr> </table>	5	6	7	10	?	?
5	6	7								
10	?	?								
20	<code>j++</code>	1	1	<table border="1"> <tr><td>5</td><td>6</td><td>7</td></tr> <tr><td>10</td><td>?</td><td>?</td></tr> </table>	5	6	7	10	?	?
5	6	7								
10	?	?								
21	<code>j <= 2</code>			true						
22	<code>a[i][j] = (i + 1) * 5 + j</code>	1	1	<table border="1"> <tr><td>5</td><td>6</td><td>7</td></tr> <tr><td>10</td><td>11</td><td>?</td></tr> </table>	5	6	7	10	11	?
5	6	7								
10	11	?								
23	<code>j++</code>	1	2	<table border="1"> <tr><td>5</td><td>6</td><td>7</td></tr> <tr><td>10</td><td>11</td><td>?</td></tr> </table>	5	6	7	10	11	?
5	6	7								
10	11	?								

24	<code>j <= 2</code>			true						
25	<code>a[i][j] = (i + 1) * 5 + j</code>	1	2	<table border="1"> <tr><td>5</td><td>6</td><td>7</td></tr> <tr><td>10</td><td>11</td><td>12</td></tr> </table>	5	6	7	10	11	12
5	6	7								
10	11	12								
26	<code>j++</code>	1	3	<table border="1"> <tr><td>5</td><td>6</td><td>7</td></tr> <tr><td>10</td><td>11</td><td>12</td></tr> </table>	5	6	7	10	11	12
5	6	7								
10	11	12								
27	<code>j <= 2</code>			false						
28	<code>i++</code>	2	3	<table border="1"> <tr><td>5</td><td>6</td><td>7</td></tr> <tr><td>10</td><td>11</td><td>12</td></tr> </table>	5	6	7	10	11	12
5	6	7								
10	11	12								
29	<code>i <= 1</code>			false						

3. Solution

Step	Statement	i	j	a									
1	<code>int a[3][3]</code>	?	?	<table border="1"> <tr><td>?</td><td>?</td><td>?</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> </table>	?	?	?	?	?	?	?	?	?
?	?	?											
?	?	?											
?	?	?											
2	<code>j = 0</code>	?	0	<table border="1"> <tr><td>?</td><td>?</td><td>?</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> </table>	?	?	?	?	?	?	?	?	?
?	?	?											
?	?	?											
?	?	?											
3	<code>j <= 2</code>			true									
4	<code>i = 0</code>	0	0	<table border="1"> <tr><td>?</td><td>?</td><td>?</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> </table>	?	?	?	?	?	?	?	?	?
?	?	?											
?	?	?											
?	?	?											
5	<code>i <= 2</code>			true									
6	<code>a[i][j] = (i + 1) * 2 + j * 4</code>	0	0	<table border="1"> <tr><td>2</td><td>?</td><td>?</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> </table>	2	?	?	?	?	?	?	?	?
2	?	?											
?	?	?											
?	?	?											
7	<code>i++</code>	1	0	<table border="1"> <tr><td>2</td><td>?</td><td>?</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> </table>	2	?	?	?	?	?	?	?	?
2	?	?											
?	?	?											
?	?	?											
8	<code>i <= 2</code>			true									
9	<code>a[i][j] = (i + 1) * 2 + j * 4</code>	1	0	<table border="1"> <tr><td>2</td><td>?</td><td>?</td></tr> <tr><td>4</td><td>?</td><td>?</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> </table>	2	?	?	4	?	?	?	?	?
2	?	?											
4	?	?											
?	?	?											

10	<code>i++</code>	2	0	<table border="1"> <tbody> <tr><td>2</td><td>?</td><td>?</td></tr> <tr><td>4</td><td>?</td><td>?</td></tr> <tr><td>?</td><td>?</td><td>?</td></tr> </tbody> </table>	2	?	?	4	?	?	?	?	?
2	?	?											
4	?	?											
?	?	?											
11	<code>i <= 2</code>	true											
12	<code>a[i][j] = (i + 1) * 2 + j * 4</code>	2	0	<table border="1"> <tbody> <tr><td>2</td><td>?</td><td>?</td></tr> <tr><td>4</td><td>?</td><td>?</td></tr> <tr><td>6</td><td>?</td><td>?</td></tr> </tbody> </table>	2	?	?	4	?	?	6	?	?
2	?	?											
4	?	?											
6	?	?											
13	<code>i++</code>	3	0	<table border="1"> <tbody> <tr><td>2</td><td>?</td><td>?</td></tr> <tr><td>4</td><td>?</td><td>?</td></tr> <tr><td>6</td><td>?</td><td>?</td></tr> </tbody> </table>	2	?	?	4	?	?	6	?	?
2	?	?											
4	?	?											
6	?	?											
14	<code>i <= 2</code>	false											
15	<code>j++</code>	3	1	<table border="1"> <tbody> <tr><td>2</td><td>?</td><td>?</td></tr> <tr><td>4</td><td>?</td><td>?</td></tr> <tr><td>6</td><td>?</td><td>?</td></tr> </tbody> </table>	2	?	?	4	?	?	6	?	?
2	?	?											
4	?	?											
6	?	?											
16	<code>j <= 2</code>	true											
17	<code>i = 0</code>	0	1	<table border="1"> <tbody> <tr><td>2</td><td>?</td><td>?</td></tr> <tr><td>4</td><td>?</td><td>?</td></tr> <tr><td>6</td><td>?</td><td>?</td></tr> </tbody> </table>	2	?	?	4	?	?	6	?	?
2	?	?											
4	?	?											
6	?	?											
18	<code>i <= 2</code>	true											
19	<code>a[i][j] = (i + 1) * 2 + j * 4</code>	0	1	<table border="1"> <tbody> <tr><td>2</td><td>6</td><td>?</td></tr> <tr><td>4</td><td>?</td><td>?</td></tr> <tr><td>6</td><td>?</td><td>?</td></tr> </tbody> </table>	2	6	?	4	?	?	6	?	?
2	6	?											
4	?	?											
6	?	?											
20	<code>i++</code>	1	1	<table border="1"> <tbody> <tr><td>2</td><td>6</td><td>?</td></tr> <tr><td>4</td><td>?</td><td>?</td></tr> <tr><td>6</td><td>?</td><td>?</td></tr> </tbody> </table>	2	6	?	4	?	?	6	?	?
2	6	?											
4	?	?											
6	?	?											
21	<code>i <= 2</code>	true											
22	<code>a[i][j] = (i + 1) * 2 + j * 4</code>	1	1	<table border="1"> <tbody> <tr><td>2</td><td>6</td><td>?</td></tr> <tr><td>4</td><td>8</td><td>?</td></tr> <tr><td>6</td><td>?</td><td>?</td></tr> </tbody> </table>	2	6	?	4	8	?	6	?	?
2	6	?											
4	8	?											
6	?	?											
23	<code>i++</code>	2	1	<table border="1"> <tbody> <tr><td>2</td><td>6</td><td>?</td></tr> <tr><td>4</td><td>8</td><td>?</td></tr> <tr><td>6</td><td>?</td><td>?</td></tr> </tbody> </table>	2	6	?	4	8	?	6	?	?
2	6	?											
4	8	?											
6	?	?											
24	<code>i <= 2</code>	true											
25	<code>a[i][j] = (i + 1) * 2 + j * 4</code>	2	1	<table border="1"> <tbody> <tr><td>2</td><td>6</td><td>?</td></tr> <tr><td>4</td><td>8</td><td>?</td></tr> <tr><td>6</td><td>10</td><td>?</td></tr> </tbody> </table>	2	6	?	4	8	?	6	10	?
2	6	?											
4	8	?											
6	10	?											

26	<code>i++</code>	3	1	<table border="1"> <tbody> <tr><td>2</td><td>6</td><td>?</td></tr> <tr><td>4</td><td>8</td><td>?</td></tr> <tr><td>6</td><td>10</td><td>?</td></tr> </tbody> </table>	2	6	?	4	8	?	6	10	?
2	6	?											
4	8	?											
6	10	?											
27	<code>i <= 2</code>	false											
28	<code>j++</code>	3	2	<table border="1"> <tbody> <tr><td>2</td><td>6</td><td>?</td></tr> <tr><td>4</td><td>8</td><td>?</td></tr> <tr><td>6</td><td>10</td><td>?</td></tr> </tbody> </table>	2	6	?	4	8	?	6	10	?
2	6	?											
4	8	?											
6	10	?											
29	<code>j <= 2</code>	true											
30	<code>i = 0</code>	0	2	<table border="1"> <tbody> <tr><td>2</td><td>6</td><td>?</td></tr> <tr><td>4</td><td>8</td><td>?</td></tr> <tr><td>6</td><td>10</td><td>?</td></tr> </tbody> </table>	2	6	?	4	8	?	6	10	?
2	6	?											
4	8	?											
6	10	?											
31	<code>i <= 2</code>	true											
32	<code>a[i][j] = (i + 1) * 2 + j * 4</code>	0	2	<table border="1"> <tbody> <tr><td>2</td><td>6</td><td>10</td></tr> <tr><td>4</td><td>8</td><td>?</td></tr> <tr><td>6</td><td>10</td><td>?</td></tr> </tbody> </table>	2	6	10	4	8	?	6	10	?
2	6	10											
4	8	?											
6	10	?											
33	<code>i++</code>	1	2	<table border="1"> <tbody> <tr><td>2</td><td>6</td><td>10</td></tr> <tr><td>4</td><td>8</td><td>?</td></tr> <tr><td>6</td><td>10</td><td>?</td></tr> </tbody> </table>	2	6	10	4	8	?	6	10	?
2	6	10											
4	8	?											
6	10	?											
34	<code>i <= 2</code>	true											
35	<code>a[i][j] = (i + 1) * 2 + j * 4</code>	1	2	<table border="1"> <tbody> <tr><td>2</td><td>6</td><td>10</td></tr> <tr><td>4</td><td>8</td><td>12</td></tr> <tr><td>6</td><td>10</td><td>?</td></tr> </tbody> </table>	2	6	10	4	8	12	6	10	?
2	6	10											
4	8	12											
6	10	?											
36	<code>i++</code>	2	2	<table border="1"> <tbody> <tr><td>2</td><td>6</td><td>10</td></tr> <tr><td>4</td><td>8</td><td>12</td></tr> <tr><td>6</td><td>10</td><td>?</td></tr> </tbody> </table>	2	6	10	4	8	12	6	10	?
2	6	10											
4	8	12											
6	10	?											
37	<code>i <= 2</code>	true											
38	<code>a[i][j] = (i + 1) * 2 + j * 4</code>	2	2	<table border="1"> <tbody> <tr><td>2</td><td>6</td><td>10</td></tr> <tr><td>4</td><td>8</td><td>12</td></tr> <tr><td>6</td><td>10</td><td>14</td></tr> </tbody> </table>	2	6	10	4	8	12	6	10	14
2	6	10											
4	8	12											
6	10	14											
39	<code>i++</code>	3	2	<table border="1"> <tbody> <tr><td>2</td><td>6</td><td>10</td></tr> <tr><td>4</td><td>8</td><td>12</td></tr> <tr><td>6</td><td>10</td><td>14</td></tr> </tbody> </table>	2	6	10	4	8	12	6	10	14
2	6	10											
4	8	12											
6	10	14											
40	<code>i <= 2</code>	false											
41	<code>j++</code>	3	3	<table border="1"> <tbody> <tr><td>2</td><td>6</td><td>10</td></tr> <tr><td>4</td><td>8</td><td>12</td></tr> <tr><td>6</td><td>10</td><td>14</td></tr> </tbody> </table>	2	6	10	4	8	12	6	10	14
2	6	10											
4	8	12											
6	10	14											

42 $j \leq 2$

false

4. Solution

For input value of 5

0	5	10
0	6	12

For input value of 9

0	9	18
0	10	20

For input value of 3

0	3	6
0	4	8

5. Solution

For input value of 13

0	3	3
0	17	18

For input value of 10

0	10	3
0	11	15

For input value of 8

3	3	3
11	12	13

6. Solution

19	5	31
28	6	20

7. Solution

26	29
37	34
59	49

8. Solution

- i. -1 15 22 25 12 16 7 9 1
- ii. 7 9 1 25 12 16 -1 15 22
- iii. 22 15 -1 16 12 25 1 9 7
- iv. 1 9 7 16 12 25 22 15 -1

v. -1 25 7 15 12 9 22 16 1
vi. 7 25 -1 9 12 15 1 16 22
vii. 22 16 1 15 12 9 -1 25 7
viii. 1 16 22 9 12 15 7 25 -1

9. Solution

```
#include <iostream>
using namespace std;
const int ROWS = 10;
const int COLUMNS = 15;

int main() {
    int i, j;

    int a[ROWS][COLUMNS];
    for (i = 0; i <= ROWS - 1; i++) {
        for (j = 0; j <= COLUMNS - 1; j++) {
            cin >> a[i][j];
        }
    }

    for (i = 0; i <= ROWS - 1; i++) {
        for (j = 0; j <= COLUMNS - 1; j++) {
            if (a[i][j] % 2 != 0) {
                cout << i << ", " << j << endl;
            }
        }
    }
    return 0;
}
```

10. Solution

```
#include <iostream>
using namespace std;
const int ROWS = 10;
const int COLUMNS = 6;

int main() {
    int i, j;

    double a[ROWS][COLUMNS];
    for (i = 0; i <= ROWS - 1; i++) {
        for (j = 0; j <= COLUMNS - 1; j++) {
            cin >> a[i][j];
        }
    }

    for (i = 0; i <= ROWS - 1; i++) {
        for (j = 0; j <= COLUMNS - 1; j += 2) {
            cout << a[i][j] << endl;
        }
    }
}
```

```
    }  
    return 0;  
}
```

11. Solution

```
#include <iostream>  
using namespace std;  
const int ROWS = 12;  
const int COLUMNS = 8;  
  
int main() {  
    int i, j;  
    double total;  
  
    double a[ROWS][COLUMNS];  
    for (i = 0; i <= ROWS - 1; i++) {  
        for (j = 0; j <= COLUMNS - 1; j++) {  
            cin >> a[i][j];  
        }  
    }  
  
    total = 0;  
    for (i = 1; i <= ROWS - 1; i += 2) {  
        for (j = 0; j <= COLUMNS - 1; j += 2) {  
            total += a[i][j];  
        }  
    }  
    cout << total << endl;  
    return 0;  
}
```

12. Solution

```
#include <iostream>  
using namespace std;  
const int N = 8 ;  
  
int main() {  
    int i, j, k;  
    double sumAntidiagonal, sumDiagonal;  
  
    double a[N][N];  
    for (i = 0; i <= N - 1; i++) {  
        for (j = 0; j <= N - 1; j++) {  
            cin >> a[i][j];  
        }  
    }  
  
    sumDiagonal = 0;  
    sumAntidiagonal = 0;  
    for (k = 0; k <= N - 1; k++) {  
        sumDiagonal += a[k][k];  
    }  
}
```

```

    sumAntidiagonal += a[k][N - k - 1];
}
cout << sumDiagonal / N << ", " << sumAntidiagonal / N << endl;
return 0;
}

```

13. Solution

```

#include <iostream>
using namespace std;
const int N = 5;

int main() {
    int i, j;

    int a[N][N];
    for (i = 0; i <= N - 1; i++) {
        for (j = 0; j <= N - 1; j++) {
            if (i == N - j - 1) {
                a[i][j] = 5;
            }
            else if (i > N - j - 1) {
                a[i][j] = 88;
            }
            else {
                a[i][j] = 11;
            }
        }
    }

    for (i = 0; i <= N - 1; i++) {
        for (j = 0; j <= N - 1; j++) {
            cout << a[i][j] << "\t";
        }
        cout << endl;
    }
    return 0;
}

```

14. Solution

```

#include <iostream>
using namespace std;
const int N = 5;

int main() {
    int i, j;

    int a[N][N];
    for (i = 0; i <= N - 1; i++) {
        for (j = 0; j <= N - 1; j++) {
            if (i == N - j - 1) {
                a[i][j] = 5;
            }
        }
    }
}

```

```

    }
    else if (i > N - j - 1) {
        a[i][j] = 88;
    }
    else {
        a[i][j] = 11;
    }
    if (i == j) {
        a[i][j] = 0;
    }
}
}

for (i = 0; i <= N - 1; i++) {
    for (j = 0; j <= N - 1; j++) {
        cout << a[i][j] << "\t";
    }
    cout << endl;
}
return 0;
}

```

15. Solution

```

#include <iostream>
using namespace std;
const int ROWS = 5;
const int COLUMNS = 4;

int main() {
    int i, j;

    double a[ROWS][COLUMNS];
    for (i = 0; i <= ROWS - 1; i++) {
        for (j = 0; j <= COLUMNS - 1; j++) {
            cin >> a[i][j];
        }
    }

    for (i = 0; i <= ROWS - 1; i++) {
        for (j = 0; j <= COLUMNS - 1; j++) {
            if (a[i][j] == (int)(a[i][j])) {
                cout << i << ", " << j << endl;
            }
        }
    }
}
return 0;
}

```

16. Solution

```

#include <iostream>

```

```
using namespace std;
const int ROWS = 10;
const int COLUMNS = 4;

int main() {
    int count, i, j;

    double a[ROWS][COLUMNS];
    for (i = 0; i <= ROWS - 1; i++) {
        for (j = 0; j <= COLUMNS - 1; j++) {
            cin >> a[i][j];
        }
    }

    count = 0;
    for (i = 0; i <= ROWS - 1; i++) {
        for (j = 0; j <= COLUMNS - 1; j++) {
            if (a[i][j] < 0) {
                count++;
            }
        }
    }
    cout << count << endl;
    return 0;
}
```

17. Solution

```
#include <iostream>
using namespace std;
const int ROWS = 3;
const int COLUMNS = 4;

int main() {
    int i, j;

    string a[ROWS][COLUMNS];
    for (i = 0; i <= ROWS - 1; i++) {
        for (j = 0; j <= COLUMNS - 1; j++) {
            cin >> a[i][j];
        }
    }

    for (i = 0; i <= ROWS - 1; i++) {
        for (j = 0; j <= COLUMNS - 1; j++) {
            cout << a[i][j] << " ";
        }
    }
    return 0;
}
```

18. Solution

```

#include <iostream>
using namespace std;
const int ROWS = 20;
const int COLUMNS = 14;

int main() {
    int i, j;

    string a[ROWS][COLUMNS];
    for (i = 0; i <= ROWS - 1; i++) {
        for (j = 0; j <= COLUMNS - 1; j++) {
            cin >> a[i][j];
        }
    }

    for (i = 0; i <= ROWS - 1; i++) {
        for (j = 0; j <= COLUMNS - 1; j++) {
            if (a[i][j].length() < 5) {
                cout << a[i][j] << endl;
            }
        }
    }
    return 0;
}

```

19. Solution

First approach

```

#include <iostream>
using namespace std;
const int ROWS = 20;
const int COLUMNS = 14;

int main() {
    int i, j, k;

    string a[ROWS][COLUMNS];
    for (i = 0; i <= ROWS - 1; i++) {
        for (j = 0; j <= COLUMNS - 1; j++) {
            cin >> a[i][j];
        }
    }

    int lengthLimits[] = {5, 10, 20};

    for (k = 0; k <= 2; k++) {
        for (i = 0; i <= ROWS - 1; i++) {
            for (j = 0; j <= COLUMNS - 1; j++) {
                if (a[i][j].length() < lengthLimits[k]) {
                    cout << a[i][j] << endl;
                }
            }
        }
    }
}

```

```
    }  
  }  
}  
}  
return 0;  
}
```

Second approach

```
#include <iostream>  
#include <cmath>  
using namespace std;  
const int ROWS = 20;  
const int COLUMNS = 14;  
  
int main() {  
  int i, j, k;  
  
  string a[ROWS][COLUMNS];  
  for (i = 0; i <= ROWS - 1; i++) {  
    for (j = 0; j <= COLUMNS - 1; j++) {  
      cin >> a[i][j];  
    }  
  }  
  
  for (k = 0; k <= 2; k++) {  
    for (i = 0; i <= ROWS - 1; i++) {  
      for (j = 0; j <= COLUMNS - 1; j++) {  
        if (a[i][j].length() < 5 * pow(2, k)) {  
          cout << a[i][j] << endl;  
        }  
      }  
    }  
  }  
}
```


Chapter 33

33.8 Review Questions: True/False

- | | |
|----------|-----------|
| 1. true | 9. false |
| 2. false | 10. false |
| 3. false | 11. true |
| 4. false | 12. true |
| 5. false | 13. true |
| 6. false | 14. true |
| 7. true | 15. true |
| 8. true | |

33.9 Review Questions: Multiple Choice

- | | |
|------|-------|
| 1. a | 7. a |
| 2. b | 8. a |
| 3. c | 9. c |
| 4. d | 10. a |
| 5. b | 11. a |
| 6. a | |

33.10 Review Exercises

1. Solution

```
#include <iostream>
using namespace std;
const int STUDENTS = 15;
const int TESTS = 5;

int main() {
    int i, j;

    int grades[STUDENTS][TESTS];
    for (i = 0; i <= STUDENTS - 1; i++) {
        for (j = 0; j <= TESTS - 1; j++) {
            cin >> grades[i][j];
        }
    }

    double average[STUDENTS];
    for (i = 0; i <= STUDENTS - 1; i++) {
        average[i] = 0;
        for (j = 0; j <= TESTS - 1; j++) {
            average[i] += grades[i][j];
        }
        average[i] /= TESTS;
    }

    for (i = 0; i <= STUDENTS - 1; i++) {
        cout << "Student No " << i + 1 << ": ";
    }
```

```

    if (average[i] < 60) {
        cout << "E/F" << endl;
    }
    else if (average[i] < 70) {
        cout << "D" << endl;
    }
    else if (average[i] < 80) {
        cout << "C" << endl;
    }
    else if (average[i] < 90) {
        cout << "B" << endl;
    }
    else {
        cout << "A" << endl;
    }
}
return 0;
}

```

2. Solution

```

#include <iostream>
using namespace std;
const int OBJECTS = 5;
const int FALLS = 10;

int main() {
    int i, j, total;

    int g[OBJECTS][FALLS];
    for (i = 0; i <= OBJECTS - 1; i++) {
        for (j = 0; j <= FALLS - 1; j++) {
            cin >> g[i][j];
        }
    }

    for (i = 0; i <= OBJECTS - 1; i++) {
        total = 0;
        for (j = 0; j <= FALLS - 1; j++) {
            total += g[i][j];
        }
        cout << "Average g for object No " << i + 1 << ": " << total / (double)FALLS << endl;
    }

    for (j = 0; j <= FALLS - 1; j++) {
        total = 0;
        for (i = 0; i <= OBJECTS - 1; i++) {
            total += g[i][j];
        }
        cout << "Average g for fall No " << j + 1 << ": " << total / (double)OBJECTS << endl;
    }
}

```

```

total = 0;
for (i = 0; i <= OBJECTS - 1; i++) {
    for (j = 0; j <= FALLS - 1; j++) {
        total += g[i][j];
    }
}
cout << "Overall average g: " << total / (double)(OBJECTS * FALLS) << endl;
return 0;
}

```

3. Solution

```

#include <iostream>
using namespace std;
const int PLAYERS = 15;
const int MATCHES = 12;

int main() {
    int i, j, total;

    int points[PLAYERS][MATCHES];
    for (i = 0; i <= PLAYERS - 1; i++) {
        for (j = 0; j <= MATCHES - 1; j++) {
            cin >> points[i][j];
        }
    }

    for (i = 0; i <= PLAYERS - 1; i++) {
        total = 0;
        for (j = 0; j <= MATCHES - 1; j++) {
            total += points[i][j];
        }
        cout << "Total number of points for player No " << i + 1 << ": " << total << endl;
    }

    for (j = 0; j <= MATCHES - 1; j++) {
        total = 0;
        for (i = 0; i <= PLAYERS - 1; i++) {
            total += points[i][j];
        }
        cout << "Total number of points for match No " << j + 1 << ": " << total << endl;
    }
    return 0;
}

```

4. Solution

```

#include <iostream>
using namespace std;
const int CITIES = 20;
const int HOURS = 24;

int main() {

```

```

int i, j;
double total;

double temperatures[CITIES][HOURS];
for (i = 0; i <= CITIES - 1; i++) {
    for (j = 0; j <= HOURS - 1; j++) {
        cin >> temperatures[i][j];
    }
}

for (j = 0; j <= HOURS - 1; j++) {
    total = 0;
    for (i = 0; i <= CITIES - 1; i++) {
        total += temperatures[i][j];
    }
    if (total / CITIES < 10) {
        cout << "Hour: " << j + 1 << endl;
    }
}
return 0;
}

```

5. Solution

```

#include <iostream>
using namespace std;
const int PLAYERS = 24;
const int MATCHES = 10;

int main() {
    int i, j, total;

    string names[PLAYERS];
    int goals[PLAYERS][MATCHES];
    for (i = 0; i <= PLAYERS - 1; i++) {
        cin >> names[i];
        for (j = 0; j <= MATCHES - 1; j++) {
            cin >> goals[i][j];
        }
    }

    for (i = 0; i <= PLAYERS - 1; i++) {
        total = 0;
        for (j = 0; j <= MATCHES - 1; j++) {
            total += goals[i][j];
        }
        cout << names[i] << ": " << total / (double)MATCHES << endl;
    }

    for (j = 0; j <= MATCHES - 1; j++) {
        total = 0;
        for (i = 0; i <= PLAYERS - 1; i++) {
            total += goals[i][j];
        }
    }
}

```

```
    }  
    cout << "Match No " << j + 1 << ": " << total << endl;  
}  
return 0;  
}
```

6. Solution

```
#include <iostream>  
using namespace std;  
const int STUDENTS = 12;  
const int LESSONS = 6;  
  
int main() {  
    int i, j, total;  
  
    string names[STUDENTS];  
    int grades[STUDENTS][LESSONS];  
    for (i = 0; i <= STUDENTS - 1; i++) {  
        cin >> names[i];  
        for (j = 0; j <= LESSONS - 1; j++) {  
            cin >> grades[i][j];  
        }  
    }  
  
    double average[STUDENTS];  
    for (i = 0; i <= STUDENTS - 1; i++) {  
        total = 0;  
        for (j = 0; j <= LESSONS - 1; j++) {  
            total += grades[i][j];  
        }  
        average[i] = total / (double)LESSONS;  
        cout << names[i] << ": " << average[i] << endl;  
    }  
  
    for (j = 0; j <= LESSONS - 1; j++) {  
        total = 0;  
        for (i = 0; i <= STUDENTS - 1; i++) {  
            total += grades[i][j];  
        }  
        cout << total / (double)STUDENTS << endl;  
    }  
  
    for (i = 0; i <= STUDENTS - 1; i++) {  
        if (average[i] < 60) {  
            cout << names[i] << endl;  
        }  
    }  
  
    for (i = 0; i <= STUDENTS - 1; i++) {  
        if (average[i] > 89) {  
            cout << names[i] << " Bravo!" << endl;  
        }  
    }  
}
```

```

}
return 0;
}

```

7. Solution

```

#include <iostream>
using namespace std;
const int ARTISTS = 15;
const int JUDGES = 5;

int main() {
    int i, j, total;

    string judgeNames[JUDGES];
    for (j = 0; j <= JUDGES - 1; j++) {
        cout << "Enter name for judge No " << j + 1 << ": ";
        cin >> judgeNames[j];
    }

    string artistNames[ARTISTS];
    string songTitles[ARTISTS];
    int score[ARTISTS][JUDGES];
    for (i = 0; i <= ARTISTS - 1; i++) {
        cout << "Enter name for artist No " << i + 1 << ": ";
        cin >> artistNames[i];
        cout << "Enter song title for artist " << artistNames[i] << ": ";
        cin >> songTitles[i];
        for (j = 0; j <= JUDGES - 1; j++) {
            cout << "Enter score for artist: " << artistNames[i];
            cout << " gotten from judge " << judgeNames[j] << ": ";
            cin >> score[i][j];
        }
    }

    for (i = 0; i <= ARTISTS - 1; i++) {
        total = 0;
        for (j = 0; j <= JUDGES - 1; j++) {
            total += score[i][j];
        }
        cout << artistNames[i] << ", " << songTitles[i] << ": " << total << endl;
    }

    for (j = 0; j <= JUDGES - 1; j++) {
        total = 0;
        for (i = 0; i <= ARTISTS - 1; i++) {
            total += score[i][j];
        }
        cout << judgeNames[j] << ": " << total / (double)ARTISTS << endl;
    }
    return 0;
}

```

8. Solution

```

#include <iostream>
#include <cmath>
using namespace std;
const int PEOPLE = 30;
const int MONTHS = 12;

int main() {
    int i, j, sumHeights, sumWeights;
    double averageHeight, averageWeight;

    int weights[PEOPLE][MONTHS];
    int heights[PEOPLE][MONTHS];
    for (i = 0; i <= PEOPLE - 1; i++) {
        for (j = 0; j <= MONTHS - 1; j++) {
            cin >> weights[i][j];
            cin >> heights[i][j];
        }
    }

    for (i = 0; i <= PEOPLE - 1; i++) {
        sumWeights = 0;
        sumHeights = 0;
        for (j = 0; j <= MONTHS - 1; j++) {
            sumWeights += weights[i][j];
            sumHeights += heights[i][j];
        }
        averageWeight = sumWeights / (double)MONTHS;
        averageHeight = sumHeights / (double)MONTHS;
        cout << averageWeight << ", " << averageHeight << endl;
        cout << averageWeight * 702 / pow(averageHeight, 2) << endl;
    }

    for (i = 0; i <= PEOPLE - 1; i++) {
        cout << weights[i][4] * 702 / pow(heights[i][4], 2) << endl;
        cout << weights[i][7] * 702 / pow(heights[i][7], 2) << endl;
    }
    return 0;
}

```

9. Solution

```

#include <iostream>
using namespace std;
const double VAT = 0.19;
const int CONSUMERS = 1000;

int main() {
    int consumed, i;
    double payment, total;

```

```

int meterReading[CONSUMERS][2];
for (i = 0; i <= CONSUMERS - 1; i++) {
    cin >> meterReading[i][0];
    cin >> meterReading[i][1];
}

total = 0;
for (i = 0; i <= CONSUMERS - 1; i++) {
    consumed = meterReading[i][1] - meterReading[i][0];
    cout << consumed << endl;
    payment = consumed * 0.07;
    payment += VAT * payment;
    cout << payment << endl;

    total += consumed;
}

cout << total << ", " << total * 0.07 + total * 0.07 * VAT << endl;
return 0;
}

```

10. Solution

```

#include <iostream>
using namespace std;
const int CURRENCIES = 4;
const int DAYS = 5;

int main() {
    int i, j;
    double average, total, usd;

    cout << "Enter an amount in US dollars: ";
    cin >> usd;

    string currency[] = {"British Pounds Sterling", "Euros", "Canadian Dollars", "Australian Dollars"};

    double rate[CURRENCIES][DAYS] = {
        {1.420, 1.421, 1.432, 1.431, 1.441},
        {1.043, 1.056, 1.038, 1.022, 1.029},
        {0.757, 0.764, 0.760, 0.750, 0.749},
        {0.620, 0.625, 0.629, 0.636, 0.639}
    };

    for (i = 0; i <= CURRENCIES - 1; i++) {
        total = 0;
        for (j = 0; j <= DAYS - 1; j++) {
            total += rate[i][j];
        }
        average = total / DAYS;
        cout << usd << " US dollars = " << usd / average << " " << currency[i] << endl;
    }
}

```


11. Solution

```
#include <iostream>
using namespace std;
const int EMPLOYEES = 10;
const int DAYS = 5;

int main() {
    int i, j;
    double totalGrossPay, grossPay, payRate, total;

    string days[] = {"Monday", "Tuesday", "Wednesday", "Thursday", "Friday"};

    cin >> payRate;

    string names[EMPLOYEES];
    int hoursWorkedPerDay[EMPLOYEES][DAYS];
    for (i = 0; i <= EMPLOYEES - 1; i++) {
        cin >> names[i];
        for (j = 0; j <= DAYS - 1; j++) {
            cin >> hoursWorkedPerDay[i][j];
        }
    }

    int hoursWorkedPerWeek[EMPLOYEES];
    for (i = 0; i <= EMPLOYEES - 1; i++) {
        hoursWorkedPerWeek[i] = 0;
        for (j = 0; j <= DAYS - 1; j++) {
            hoursWorkedPerWeek[i] += hoursWorkedPerDay[i][j];
        }
        if (hoursWorkedPerWeek[i] > 40) {
            cout << names[i] << endl;
        }
    }

    totalGrossPay = 0;
    for (i = 0; i <= EMPLOYEES - 1; i++) {
        if (hoursWorkedPerWeek[i] <= 40) {
            grossPay = payRate * hoursWorkedPerWeek[i];
        }
        else {
            grossPay = payRate * 40 + 1.5 * payRate * (hoursWorkedPerWeek[i] - 40);
        }
        totalGrossPay += grossPay;
        cout << names[i] << ", " << grossPay / 5 << endl;
    }

    cout << totalGrossPay << endl;

    for (i = 0; i <= EMPLOYEES - 1; i++) {
        if (hoursWorkedPerWeek[i] > 40) {
            for (j = 0; j <= DAYS - 1; j++) {
                if (hoursWorkedPerDay[i][j] > 8) {
```

```

        cout << names[i] << ", " << days[j] << " Overtime!" << endl;
    }
}
}
}

for (j = 0; j <= DAYS - 1; j++) {
    total = 0;
    for (i = 0; i <= EMPLOYEES - 1; i++) {
        if (hoursWorkedPerDay[i][j] <= 8) {
            grossPay = payRate * hoursWorkedPerDay[i][j];
        }
        else {
            grossPay = payRate * 8 + 1.5 * payRate * (hoursWorkedPerDay[i][j] - 8);
        }
        total += grossPay;
    }
    cout << days[j] << ", " << total << endl;
}
return 0;
}
}

```

12. Solution

```

#include <iostream>
using namespace std;
const int ROWS = 3;
const int COLUMNS = 4;

int main() {
    int i, j, k;

    int a[ROWS][COLUMNS] = {
        {9, 9, 2, 6},
        {4, 1, 10, 11},
        {12, 15, 7, 3}
    };

    int b[ROWS * COLUMNS];
    k = 0;
    for (i = 0; i <= ROWS - 1; i++) {
        for (j = 0; j <= COLUMNS - 1; j++) {
            b[k++] = a[i][j];
        }
    }

    for (k = 0; k <= ROWS * COLUMNS - 1; k++) {
        cout << b[k] << " ";
    }
    return 0;
}

```

13. Solution

```
#include <iostream>
using namespace std;
const int ROWS = 3;
const int COLUMNS = 3;

int main() {
    int i, j, k;

    int a[] = {16, 12, 3, 5, 6, 9, 18, 19, 20};

    int b[ROWS][COLUMNS];
    k = 0;
    for (i = ROWS - 1; i >= 0; i--) {
        for (j = 0; j <= COLUMNS - 1; j++) {
            b[i][j] = a[k++];
        }
    }

    for (i = 0; i <= ROWS - 1; i++) {
        for (j = 0; j <= COLUMNS - 1; j++) {
            cout << b[i][j] << "\t";
        }
        cout << endl;
    }
    return 0;
}
```

Chapter 34

34.7 Review Questions: True/False

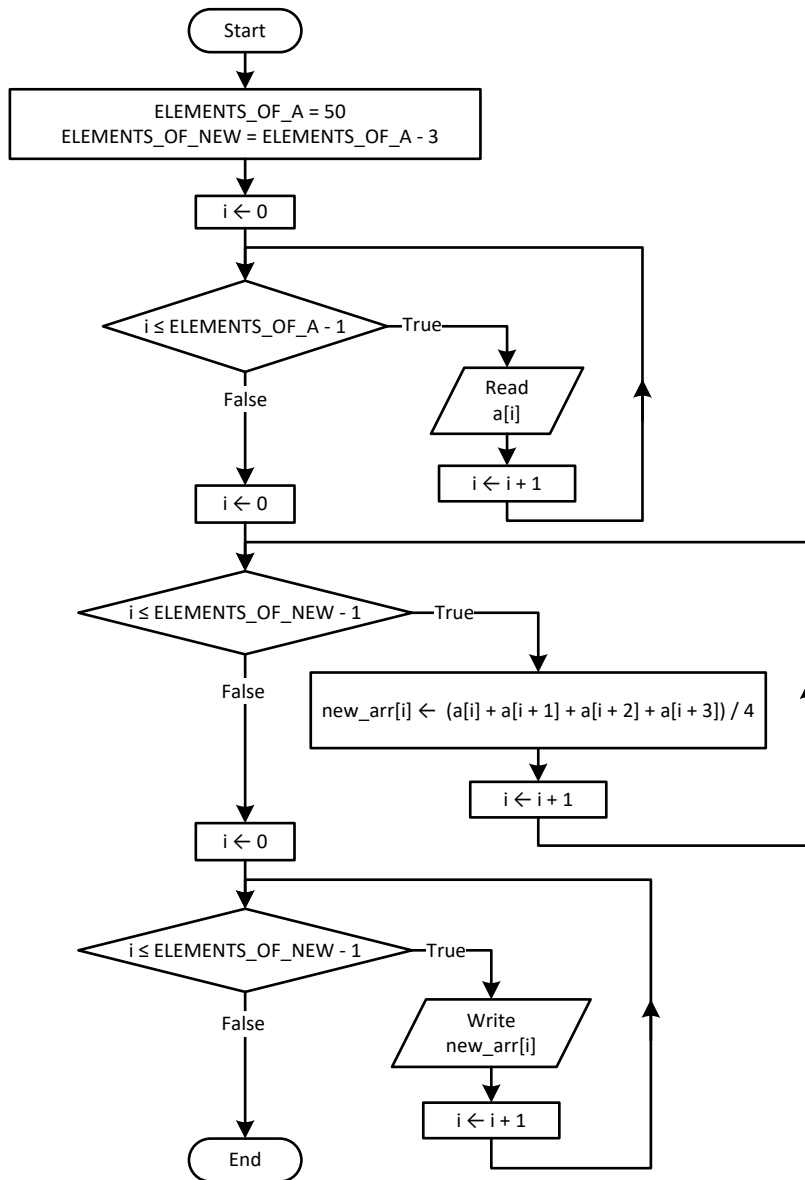
- | | |
|-----------|-----------|
| 1. true | 21. true |
| 2. false | 22. true |
| 3. true | 23. true |
| 4. true | 24. false |
| 5. true | 25. true |
| 6. true | 26. false |
| 7. true | 27. false |
| 8. false | 28. false |
| 9. true | 29. true |
| 10. false | 30. true |
| 11. false | 31. true |
| 12. true | 32. false |
| 13. false | 33. true |
| 14. false | 34. false |
| 15. false | 35. true |
| 16. true | 36. true |
| 17. true | 37. false |
| 18. true | 38. true |
| 19. false | 39. true |
| 20. false | 40. false |

34.8 Review Exercises

1. Solution

```
for (i = 0; i <= ROWS - 1; i++) {
    for (j = 0; j <= COLUMNS - 1; j++) {
        cin >> a[i][j];
        while (a[i][j] == 0) {
            cout << "Error" << endl;
            cin >> a[i][j];
        }
    }
}
```

2. Solution



```

#include <iostream>
using namespace std;
const int ELEMENTS_OF_A = 50;
const int ELEMENTS_OF_NEW = ELEMENTS_OF_A - 3;

int main() {
    int i;

    double a[ELEMENTS_OF_A];
    for (i = 0; i <= ELEMENTS_OF_A - 1; i++) {
        cin >> a[i];
    }

    double newArr[ELEMENTS_OF_NEW];
    for (i = 0; i <= ELEMENTS_OF_NEW - 1; i++) {

```

```
    newArr[i] = (a[i] + a[i + 1] + a[i + 2] + a[i + 3]) / 4;
}

for (i = 0; i <= ELEMENTS_OF_NEW - 1; i++) {
    cout << newArr[i] << "\t" << endl;
}

return 0;
}
```

3. Solution

```
#include <iostream>
using namespace std;
const int ELEMENTS = 15;

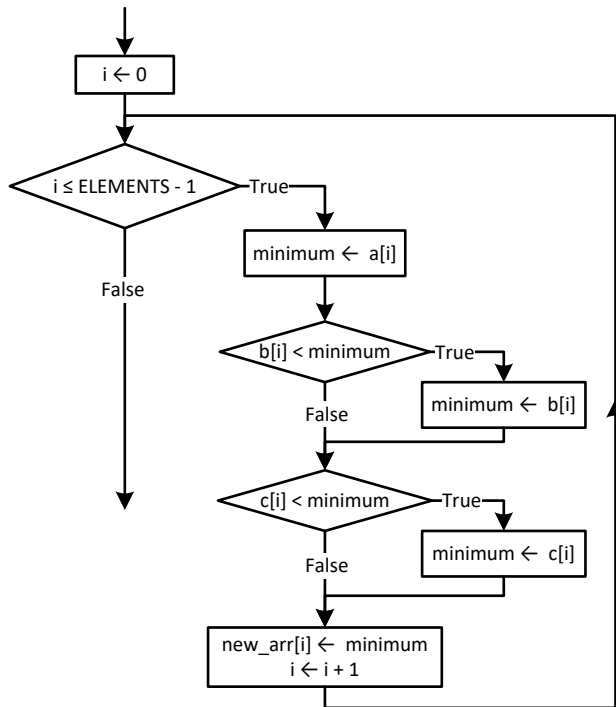
int main() {
    int i;
    double minimum;

    double a[ELEMENTS];
    for (i = 0; i <= ELEMENTS - 1; i++) {
        cin >> a[i];
    }
    double b[ELEMENTS];
    for (i = 0; i <= ELEMENTS - 1; i++) {
        cin >> b[i];
    }
    double c[ELEMENTS];
    for (i = 0; i <= ELEMENTS - 1; i++) {
        cin >> c[i];
    }

    double newArr[ELEMENTS];
    for (i = 0; i <= ELEMENTS - 1; i++) {
        minimum = a[i];
        if (b[i] < minimum) {
            minimum = b[i];
        }
        if (c[i] < minimum) {
            minimum = c[i];
        }
        newArr[i] = minimum;
    }

    for (i = 0; i <= ELEMENTS - 1; i++) {
        cout << newArr[i] << endl;
    }

    return 0;
}
```



4. Solution

```

#include <iostream>
using namespace std;
const int ELEMENTS_OF_A = 10;
const int ELEMENTS_OF_B = 5;
const int ELEMENTS_OF_C = 15;
const int ELEMENTS_OF_NEW = ELEMENTS_OF_A + ELEMENTS_OF_B + ELEMENTS_OF_C;

int main() {
    int i;

    double a[ELEMENTS_OF_A];
    for (i = 0; i <= ELEMENTS_OF_A - 1; i++) {
        cin >> a[i];
    }
    double b[ELEMENTS_OF_B];
    for (i = 0; i <= ELEMENTS_OF_B - 1; i++) {
        cin >> b[i];
    }
    double c[ELEMENTS_OF_C];
    for (i = 0; i <= ELEMENTS_OF_C - 1; i++) {
        cin >> c[i];
    }

    double newArr[ELEMENTS_OF_NEW];
    for (i = 0; i <= ELEMENTS_OF_C - 1; i++) {
        newArr[i] = c[i];
    }
}

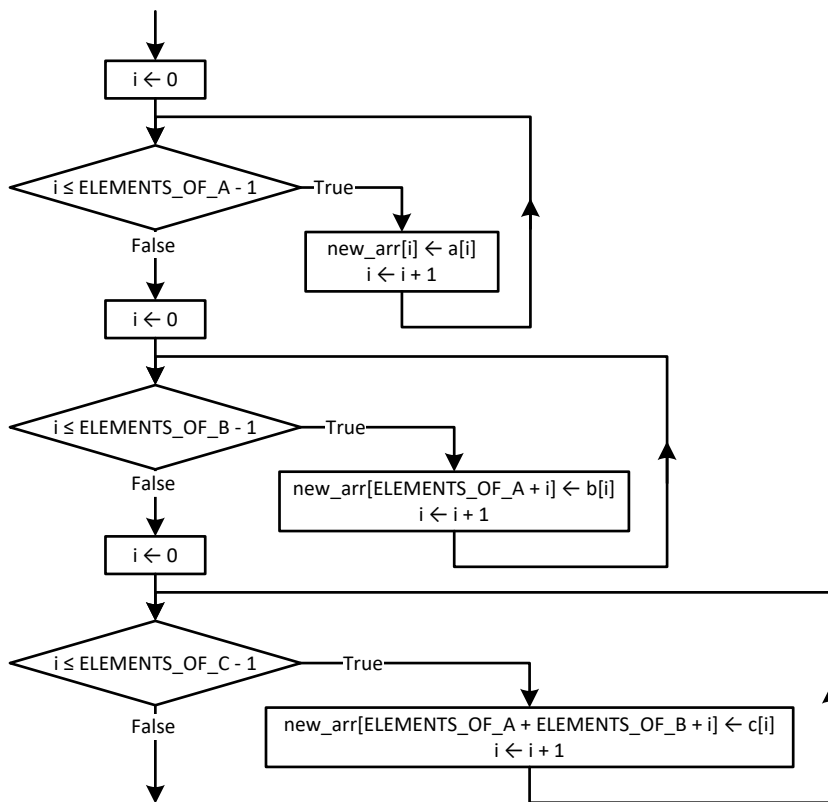
```

```

for (i = 0; i <= ELEMENTS_OF_B - 1; i++) {
    newArr[ELEMENTS_OF_C + i] = b[i];
}
for (i = 0; i <= ELEMENTS_OF_A - 1; i++) {
    newArr[ELEMENTS_OF_B + ELEMENTS_OF_C + i] = a[i];
}

//Display array new
for (i = 0; i <= ELEMENTS_OF_NEW - 1; i++) {
    cout << newArr[i] << "\t";
}
return 0;
}

```



5. Solution

```

#include <iostream>
using namespace std;
const int COLUMNS = 4;
const int ROWS_OF_A = 3;
const int ROWS_OF_B = 5;
const int ROWS_OF_NEW = ROWS_OF_A + ROWS_OF_B;

int main() {
    int i, j;

    //Create arrays a and b

```



```

int a[ROWS_OF_A][COLUMNS] = {
    {10, 11, 12, 85},
    {3, 1, 5, 10},
    {-1, 2, -5, -10}
};
int b[ROWS_OF_B][COLUMNS] = {
    {10, 11, 16, 33},
    {11, 13, 5, 55},
    {-1, -2, -4, 44},
    {55, 33, 77, 12},
    {-110, 120, 132, 43}
};

//Create array newArr
int newArr[ROWS_OF_NEW][COLUMNS];
for (i = 0; i <= ROWS_OF_A - 1; i++) {
    for (j = 0; j <= COLUMNS - 1; j++) {
        newArr[i][j] = a[i][j];
    }
}
for (i = 0; i <= ROWS_OF_B - 1; i++) {
    for (j = 0; j <= COLUMNS - 1; j++) {
        newArr[ROWS_OF_A + i][j] = b[i][j];
    }
}

//Display array newArr
for (i = 0; i <= ROWS_OF_NEW - 1; i++) {
    for (j = 0; j <= COLUMNS - 1; j++) {
        cout << newArr[i][j] << "\t";
    }
    cout << endl;
}
return 0;
}

```

6. Solution

```

#include <iostream>
using namespace std;
const int COLUMNS_OF_A = 10;
const int COLUMNS_OF_B = 15;
const int COLUMNS_OF_C = 20;
const int ROWS = 5;
const int COLUMNS = COLUMNS_OF_A + COLUMNS_OF_B + COLUMNS_OF_C;

int main() {
    int i, j;

    double a[ROWS][COLUMNS_OF_A];
    for (i = 0; i <= ROWS - 1; i++) {
        for (j = 0; j <= COLUMNS_OF_A - 1; j++) {

```

```

        cin >> a[i][j];
    }
}

double b[ROWS][COLUMNS_OF_B];
for (i = 0; i <= ROWS - 1; i++) {
    for (j = 0; j <= COLUMNS_OF_B - 1; j++) {
        cin >> b[i][j];
    }
}

double c[ROWS][COLUMNS_OF_C];
for (i = 0; i <= ROWS - 1; i++) {
    for (j = 0; j <= COLUMNS_OF_C - 1; j++) {
        cin >> c[i][j];
    }
}

double newArr[ROWS][COLUMNS];
for (i = 0; i <= ROWS - 1; i++) {
    for (j = 0; j <= COLUMNS_OF_A - 1; j++) {
        newArr[i][j] = a[i][j];
    }
}
for (i = 0; i <= ROWS - 1; i++) {
    for (j = 0; j <= COLUMNS_OF_B - 1; j++) {
        newArr[i][COLUMNS_OF_A + j] = b[i][j];
    }
}
for (i = 0; i <= ROWS - 1; i++) {
    for (j = 0; j <= COLUMNS_OF_C - 1; j++) {
        newArr[i][COLUMNS_OF_A + COLUMNS_OF_B + j] = c[i][j];
    }
}

for (i = 0; i <= ROWS - 1; i++) {
    for (j = 0; j <= COLUMNS - 1; j++) {
        cout << newArr[i][j] << "\t";
    }
    cout << endl;
}
return 0;
}

```

7. Solution

```

#include <iostream>
using namespace std;
const int ELEMENTS = 50;

int main() {
    int i, integersIndex, realsIndex;

```

```

double a[ELEMENTS];
for (i = 0; i <= ELEMENTS - 1; i++) {
    cin >> a[i];
}

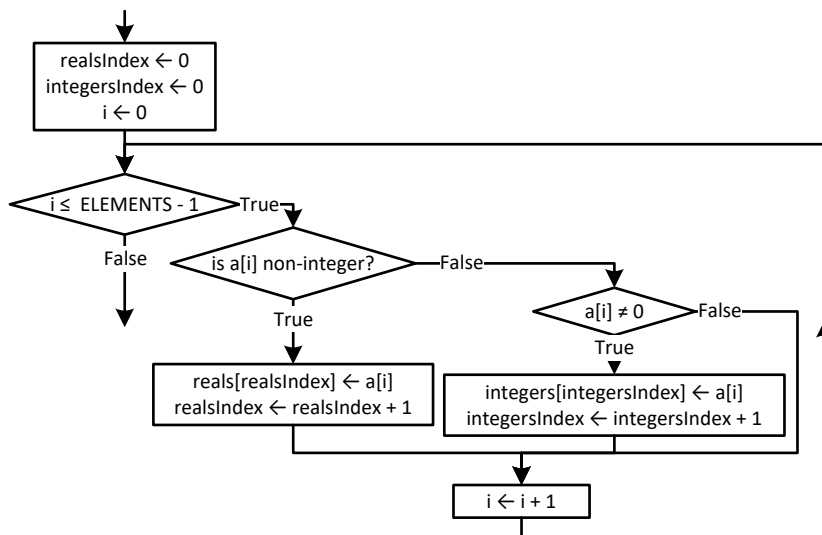
double reals[ELEMENTS];
int integers[ELEMENTS];
realsIndex = 0;
integersIndex = 0;
for (i = 0; i <= ELEMENTS - 1; i++) {
    if (a[i] != (int)(a[i])) {
        reals[realsIndex] = a[i];
        realsIndex++;
    }
    else if (a[i] != 0) {
        integers[integersIndex] = (int)a[i];
        integersIndex++;
    }
}

for (i = 0; i <= realsIndex - 1; i++) {
    cout << reals[i] << "\t";
}

cout << endl;
for (i = 0; i <= integersIndex - 1; i++) {
    cout << integers[i] << "\t";
}

return 0;
}

```



8. Solution

```

#include <iostream>
using namespace std;

```

```

const int ELEMENTS = 50;

int main() {
    int digit1, digit2, digit3, i, k, r;

    int a[ELEMENTS];
    for (i = 0; i <= ELEMENTS - 1; i++) {
        cin >> a[i];
    }

    int b[ELEMENTS];
    k = 0;
    for (i = 0; i <= ELEMENTS - 1; i++) {
        digit3 = a[i] % 10;
        r = (int) (a[i] / 10);
        digit2 = r % 10;
        digit1 = (int) (r / 10);

        if (digit1 < digit2 && digit2 < digit3) {
            b[k] = a[i];
            k++;
        }
    }

    for (i = 0; i <= k - 1; i++) {
        cout << b[i] << "\t";
    }
    return 0;
}

```

9. Solution

```

#include <iostream>
using namespace std;
const int PRODUCTS = 10;
const int CITIZENS = 200;

int main() {
    int countB, i, j, maximum;

    string prodNames[PRODUCTS];
    string answers[PRODUCTS][CITIZENS];
    for (i = 0; i <= PRODUCTS - 1; i++) {
        cin >> prodNames[i];
        for (j = 0; j <= CITIZENS - 1; j++) {
            cin >> answers[i][j];
            while (answers[i][j] < "A" || answers[i][j] > "D") {
                cout << "Error! " << endl;
                cin >> answers[i][j];
            }
        }
    }
}

```

```

int countA[PRODUCTS];
for (i = 0; i <= PRODUCTS - 1; i++) {
    countA[i] = 0;
    for (j = 0; j <= CITIZENS - 1; j++) {
        if (answers[i][j] == "A") {
            countA[i]++;
        }
    }
    cout << prodNames[i] << ", " << countA[i] << endl;
}

for (j = 0; j <= CITIZENS - 1; j++) {
    countB = 0;
    for (i = 0; i <= PRODUCTS - 1; i++) {
        if (answers[i][j] == "B") {
            countB++;
        }
    }
    cout << countB << endl;
}

maximum = countA[0];
for (i = 1; i <= PRODUCTS - 1; i++) {
    if (countA[i] > maximum) {
        maximum = countA[i];
    }
}
for (i = 0; i <= PRODUCTS - 1; i++) {
    if (countA[i] == maximum) {
        cout << prodNames[i] << endl;
    }
}
return 0;
}

```

10. Solution

```

#include <iostream>
using namespace std;
const int US_CITIES = 20;
const int CANADIAN_CITIES = 20;

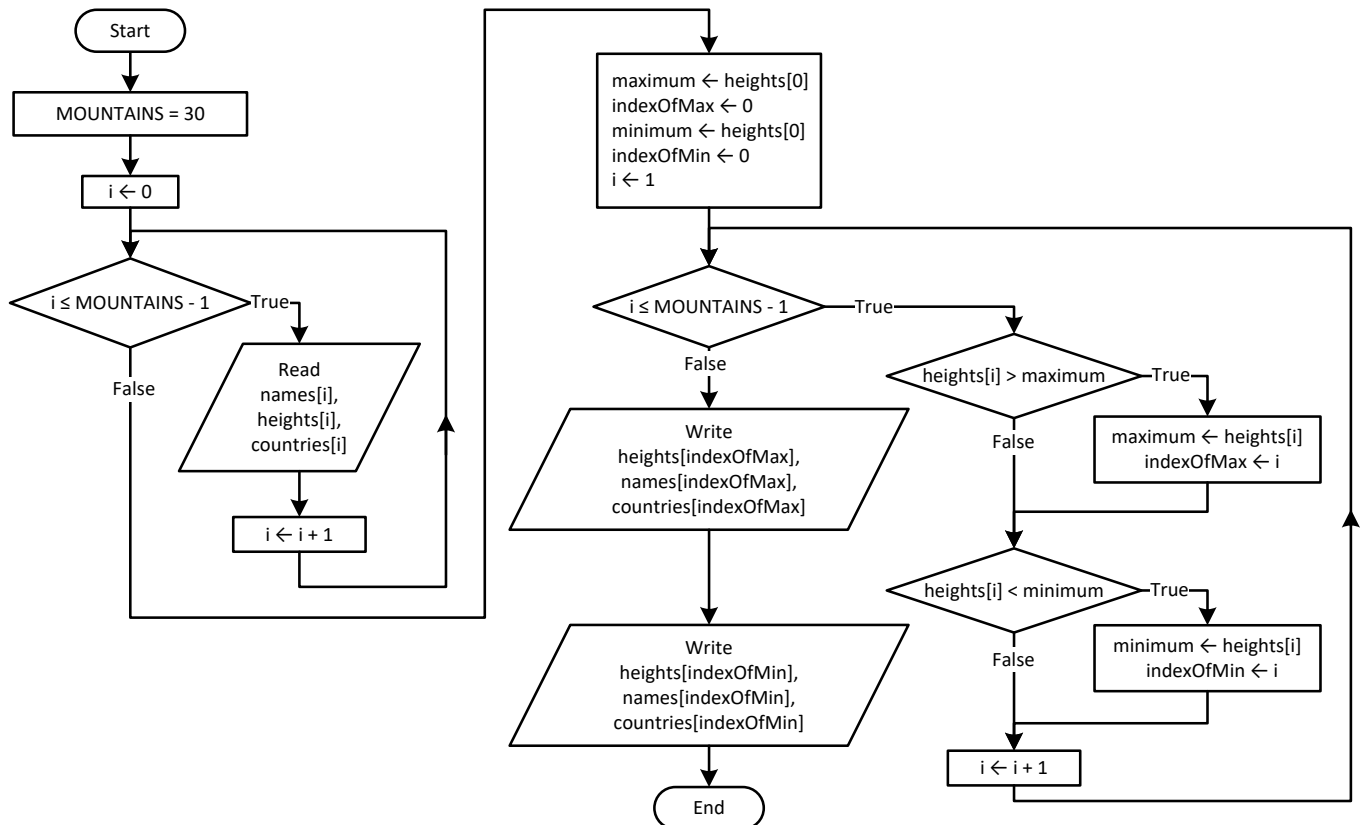
int main() {
    int i, j, minJ;
    double minimum;

    string usNames[US_CITIES];
    for (i = 0; i <= US_CITIES - 1; i++) {
        cout << "Enter name for US city No " << i + 1 << ": ";
        cin >> usNames[i];
    }
}

```

```
string canadianNames[CANADIAN_CITIES];  
for (j = 0; j <= CANADIAN_CITIES - 1; j++) {  
    cout << "Enter name for Canadian city No " << j + 1 << ": ";  
    cin >> canadianNames[j];  
}  
  
double distances[US_CITIES][CANADIAN_CITIES];  
for (i = 0; i <= US_CITIES - 1; i++) {  
    for (j = 0; j <= CANADIAN_CITIES - 1; j++) {  
        cout << "Enter distance between " << usNames[i] << " and " << canadianNames[j] << ": ";  
        cin >> distances[i][j];  
    }  
}  
  
for (i = 0; i <= US_CITIES - 1; i++) {  
    minimum = distances[i][0];  
    minJ = 0;  
    for (j = 1; j <= CANADIAN_CITIES - 1; j++) {  
        if (distances[i][j] < minimum) {  
            minimum = distances[i][j];  
            minJ = j;  
        }  
    }  
    cout << "Closest Canadian city to " << usNames[i] << " is " << canadianNames[minJ] << endl;  
}  
return 0;  
}
```

11. Solution



```

#include <iostream>
using namespace std;
const int MOUNTAINS = 30;

int main() {
    int i, indexOfMax, indexOfMin;
    double maximum, minimum;

    string names[MOUNTAINS];
    double heights[MOUNTAINS];
    string countries[MOUNTAINS];
    for (i = 0; i <= MOUNTAINS - 1; i++) {
        cin >> names[i];
        cin >> heights[i];
        cin >> countries[i];
    }

    maximum = heights[0];
    indexOfMax = 0;
    minimum = heights[0];
    indexOfMin = 0;
    for (i = 1; i <= MOUNTAINS - 1; i++) {
        if (heights[i] > maximum) {
            maximum = heights[i];
            indexOfMax = i;
        }
    }
}

```

```

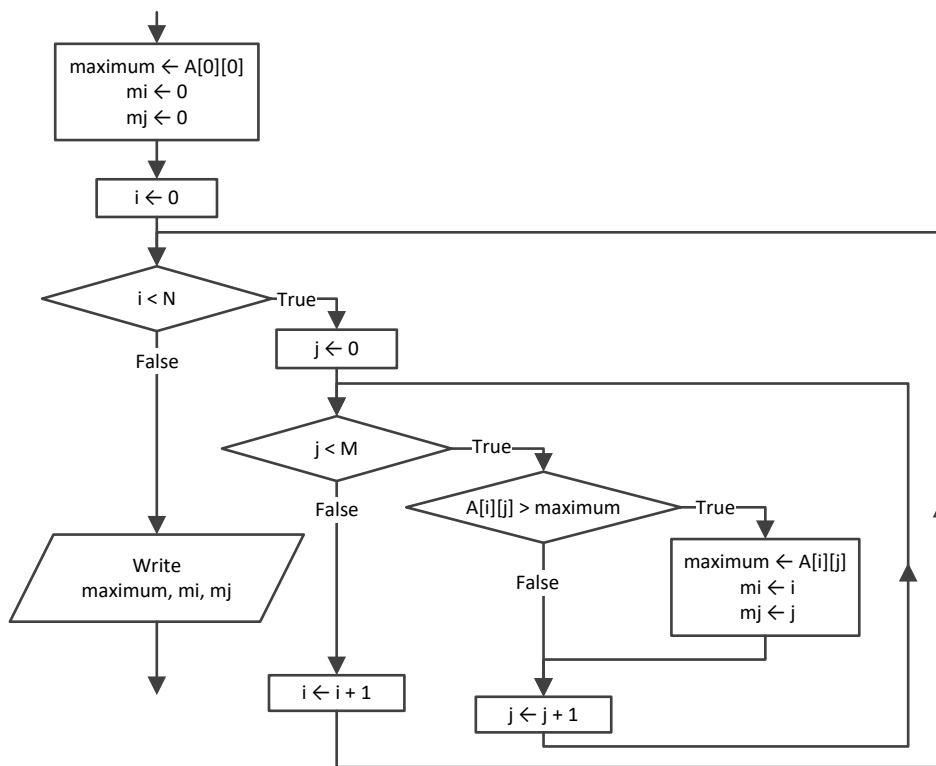
    }
    if (heights[i] < minimum) {
        minimum = heights[i];
        indexOfMin = i;
    }
}

cout << heights[indexOfMax] << ", " << names[indexOfMax] << ", " << countries[indexOfMax] << endl;
cout << heights[indexOfMin] << ", " << names[indexOfMin] << ", " << countries[indexOfMin] << endl;

return 0;
}

```

12. Solution



13. Solution

```

#include <iostream>
using namespace std;
const int TEAMS = 26;
const int GAMES = 15;

int main() {
    int i, j, m_i, maximum;

    string names[TEAMS];
    string results[TEAMS][GAMES];
    for (i = 0; i <= TEAMS - 1; i++) {

```



```

    cin >> names[i];
    for (j = 0; j <= GAMES - 1; j++) {
        cin >> results[i][j];
    }
}

int points[TEAMS];
for (i = 0; i <= TEAMS - 1; i++) {
    points[i] = 0;
    for (j = 0; j <= GAMES - 1; j++) {
        if (results[i][j] == "W") {
            points[i] += 3;
        }
        else if (results[i][j] == "T") {
            points[i] += 1;
        }
    }
}

maximum = points[0];
m_i = 0;
for (i = 1; i <= TEAMS - 1; i++) {
    if (points[i] > maximum) {
        maximum = points[i];
        m_i = i;
    }
}

cout << names[m_i] << endl;
return 0;
}

```

14. Solution

```

#include <iostream>
#include <cmath>
using namespace std;
const int OBJECTS = 10;
const int FALLS = 20;

int main() {
    int i, j;
    double maxi, mini;

    double heights[OBJECTS][FALLS];
    double times[OBJECTS][FALLS];
    for (i = 0; i <= OBJECTS - 1; i++) {
        for (j = 0; j <= FALLS - 1; j++) {
            cin >> heights[i][j];
            cin >> times[i][j];
        }
    }
}

```

```

double g[OBJECTS][FALLS];
for (i = 0; i <= OBJECTS - 1; i++) {
    for (j = 0; j <= FALLS - 1; j++) {
        g[i][j] = 2 * heights[i][j] / pow(times[i][j], 2);
    }
}

double minimum[OBJECTS];
double maximum[OBJECTS];
for (i = 0; i <= OBJECTS - 1; i++) {
    minimum[i] = g[i][0];
    maximum[i] = g[i][0];
    for (j = 1; j <= FALLS - 1; j++) {
        if (g[i][j] < minimum[i]) {
            minimum[i] = g[i][j];
        }
        if (g[i][j] > maximum[i]) {
            maximum[i] = g[i][j];
        }
    }
}

for (i = 0; i <= OBJECTS - 1; i++) {
    cout << minimum[i] << ", " << maximum[i] << endl;
}

maxi = maximum[0];
mini = minimum[0];
for (i = 1; i <= OBJECTS - 1; i++) {
    if (maximum[i] > maxi) {
        maxi = maximum[i];
    }
    if (minimum[i] < mini) {
        mini = minimum[i];
    }
}

cout << mini << ", " << maxi << endl;
return 0;
}

```

15. Solution

```

#include <iostream>
using namespace std;
const int STATIONS = 10;
const int DAYS = 365;

int main() {
    int i, j, m_i;
    double minimum;

    string names[STATIONS];

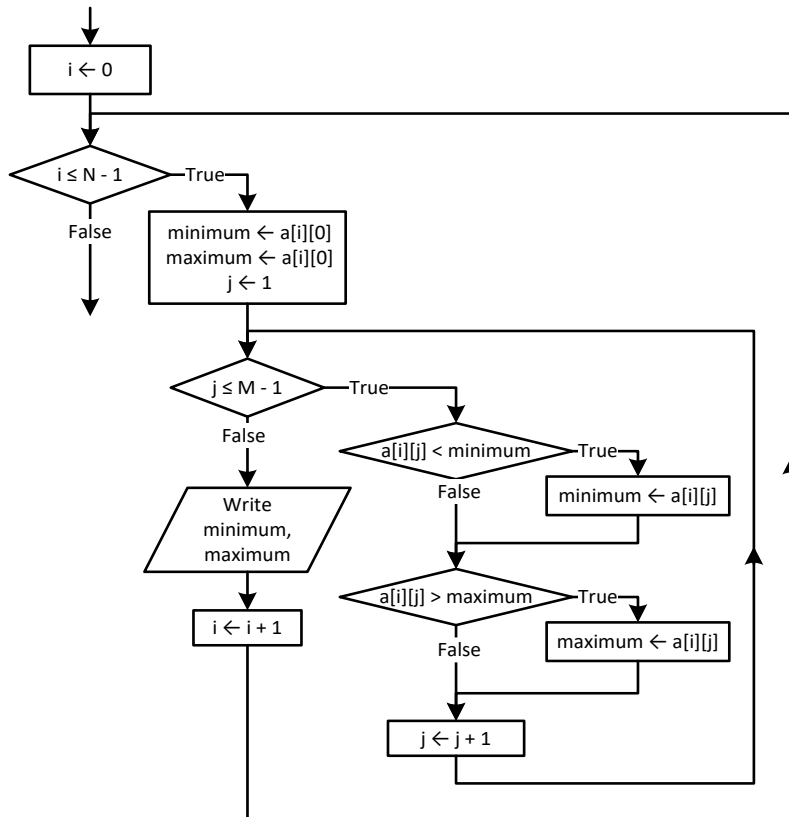
```

```
double co2[STATIONS][DAYS];
for (i = 0; i <= STATIONS - 1; i++) {
    cin >> names[i];
    for (j = 0; j <= DAYS - 1; j++) {
        cin >> co2[i][j];
    }
}

double average[STATIONS];
for (i = 0; i <= STATIONS - 1; i++) {
    average[i] = 0;
    for (j = 0; j <= DAYS - 1; j++) {
        average[i] += co2[i][j];
    }
    average[i] /= DAYS;
}

minimum = average[0];
m_i = 0;
for (i = 1; i <= STATIONS - 1; i++) {
    if (average[i] < minimum) {
        minimum = average[i];
        m_i = i;
    }
}
cout << names[m_i] << endl;
return 0;
}
```

16. Solution



17. Solution

First approach

```

#include <iostream>
using namespace std;
const int ROWS = 20;
const int COLUMNS = 30;

int main() {
    int i, j;

    double b[ROWS][COLUMNS];
    for (i = 0; i <= ROWS - 1; i++) {
        for (j = 0; j <= COLUMNS - 1; j++) {
            cin >> b[i][j];
        }
    }

    double minimum[COLUMNS];
    double maximum[COLUMNS];
    for (j = 0; j <= COLUMNS - 1; j++) {
        minimum[j] = b[0][j];
        maximum[j] = b[0][j];
        for (i = 1; i <= ROWS - 1; i++) {

```

```

    if (b[i][j] < minimum[j]) {
        minimum[j] = b[i][j];
    }
    if (b[i][j] > maximum[j]) {
        maximum[j] = b[i][j];
    }
}
}

for (j = 0; j <= COLUMNS - 1; j++) {
    cout << minimum[j] << " " << maximum[j] << endl;
}
return 0;
}

```

Second approach

```

#include <iostream>
using namespace std;
const int ROWS = 20;
const int COLUMNS = 30;

int main() {
    int i, j;
    double minimum, maximum;

    double b[ROWS][COLUMNS];
    for (i = 0; i <= ROWS - 1; i++) {
        for (j = 0; j <= COLUMNS - 1; j++) {
            cin >> b[i][j];
        }
    }

    for (j = 0; j <= COLUMNS - 1; j++) {
        minimum = b[0][j];
        maximum = b[0][j];
        for (i = 1; i <= ROWS - 1; i++) {
            if (b[i][j] < minimum) {
                minimum = b[i][j];
            }
            if (b[i][j] > maximum) {
                maximum = b[i][j];
            }
        }
        cout << minimum << " " << maximum << endl;
    }
    return 0;
}

```

18. Solution

```

#include <iostream>
using namespace std;
const int TEAMS = 20;

```

```
const int GAMES = 10;

int main() {
    int i, j, m, n, temp;
    bool swaps;
    string tempStr;

    string names[TEAMS];
    string results[TEAMS][GAMES];
    for (i = 0; i <= TEAMS - 1; i++) {
        cout << "Enter team name: ";
        cin >> names[i];
        for (j = 0; j <= GAMES - 1; j++) {
            cout << "Enter result for team " << names[i] << " for game No " << j + 1 << ": ";
            cin >> results[i][j];
            while (results[i][j] != "W" && results[i][j] != "L" && results[i][j] != "T") {
                cout << "Error! Enter only value W, L, or T: ";
                cin >> results[i][j];
            }
        }
    }

    int points[TEAMS];
    for (i = 0; i <= TEAMS - 1; i++) {
        points[i] = 0;
        for (j = 0; j <= GAMES - 1; j++) {
            if (results[i][j] == "W") {
                points[i] += 3;
            }
            else if (results[i][j] == "T") {
                points[i] += 1;
            }
        }
    }

    for (m = 1; m <= 3; m++) { //Perform only three passes
        swaps = false;
        for (n = TEAMS - 1; n >= m; n--) {
            if (points[n] > points[n - 1]) {
                temp = points[n];
                points[n] = points[n - 1];
                points[n - 1] = temp;

                tempStr = names[n];
                names[n] = names[n - 1];
                names[n - 1] = tempStr;

                swaps = true;
            }
        }
        if (!swaps) break;
    }
}
```

```

    cout << "Gold: " << names[0] << endl;
    cout << "Silver: " << names[1] << endl;
    cout << "Bronze: " << names[2] << endl;
    return 0;
}

```

19. Solution

```

#include <iostream>
using namespace std;
const int PEOPLE = 50;

int main() {
    int i, m, n;
    double temp;
    string tempStr;

    string names[PEOPLE];
    double heights[PEOPLE];
    for (i = 0; i <= PEOPLE - 1; i++) {
        cout << "Enter name for person No. " << i + 1 << ": ";
        cin >> names[i];
        cout << "Enter height for person No. " << i + 1 << ": ";
        cin >> heights[i];
    }

    for (m = 1; m <= PEOPLE - 1; m++) {
        for (n = PEOPLE - 1; n >= m; n--) {
            if (heights[n] > heights[n - 1]) {
                temp = heights[n];
                heights[n] = heights[n - 1];
                heights[n - 1] = temp;

                tempStr = names[n];
                names[n] = names[n - 1];
                names[n - 1] = tempStr;
            }
            else if (heights[n] == heights[n - 1]) {
                if (names[n] < names[n - 1]) {
                    tempStr = names[n];
                    names[n] = names[n - 1];
                    names[n - 1] = tempStr;
                }
            }
        }
    }

    for (i = 0; i <= PEOPLE - 1; i++) {
        cout << heights[i] << "\t" << names[i] << endl;
    }
    return 0;
}

```

20. Solution

```
#include <iostream>
using namespace std;
const int PEOPLE = 50;

int main() {
    string tempStr;
    int i, m, n;

    string firstNames[PEOPLE];
    string lastNames[PEOPLE];
    string fatherNames[PEOPLE];

    for (i = 0; i <= PEOPLE - 1; i++) {
        cout << "Enter first name for person No." << i + 1 << ": ";
        cin >> firstNames[i];
        cout << "Enter last name for person No." << i + 1 << ": ";
        cin >> lastNames[i];
        cout << "Enter father's name for person No." << i + 1 << ": ";
        cin >> fatherNames[i];
    }

    for (m = 1; m <= PEOPLE - 1; m++) {
        for (n = PEOPLE - 1; n >= m; n--) {
            if (lastNames[n] < lastNames[n - 1]) {
                tempStr = lastNames[n];
                lastNames[n] = lastNames[n - 1];
                lastNames[n - 1] = tempStr;

                tempStr = firstNames[n];
                firstNames[n] = firstNames[n - 1];
                firstNames[n - 1] = tempStr;

                tempStr = fatherNames[n];
                fatherNames[n] = fatherNames[n - 1];
                fatherNames[n - 1] = tempStr;
            }
            else if (lastNames[n] == lastNames[n - 1]) {
                if (firstNames[n] < firstNames[n - 1]) {
                    tempStr = firstNames[n];
                    firstNames[n] = firstNames[n - 1];
                    firstNames[n - 1] = tempStr;

                    tempStr = fatherNames[n];
                    fatherNames[n] = fatherNames[n - 1];
                    fatherNames[n - 1] = tempStr;
                }
            }
            else if (firstNames[n] == firstNames[n - 1]) {
                if (fatherNames[n] < fatherNames[n - 1]) {
                    tempStr = fatherNames[n];
                    fatherNames[n] = fatherNames[n - 1];
                }
            }
        }
    }
}
```



```

        fatherNames[n - 1] = tempStr;
    }
}
}
}

for (i = 0; i <= PEOPLE - 1; i++) {
    cout << lastNames[i] << "\t" << firstNames[i] << "\t" << fatherNames[i] << endl;
}
return 0;
}

```

21. Solution

```

#include <iostream>
using namespace std;
const int ARTISTS = 12;
const int JUDGES = 10;

int main() {
    int i, j, m, maximum, minimum, n, temp;
    string tempStr;

    string artistNames[ARTISTS];
    int score[ARTISTS][JUDGES];
    for (i = 0; i <= ARTISTS - 1; i++) {
        cout << "Enter name for artist No " << i + 1 << ": ";
        cin >> artistNames[i];
        for (j = 0; j <= JUDGES - 1; j++) {
            cout << "Enter score for artist: " << artistNames[i];
            cout << " gotten from judge No " << j + 1 << ": ";
            cin >> score[i][j];
        }
    }

    int total[ARTISTS];
    for (i = 0; i <= ARTISTS - 1; i++) {
        total[i] = 0;
        for (j = 1; j <= JUDGES - 1; j++) {
            total[i] += score[i][j];
        }
    }

    for (i = 0; i <= ARTISTS - 1; i++) {
        minimum = score[i][0];
        maximum = score[i][0];
        for (j = 1; j <= JUDGES - 1; j++) {
            if (score[i][j] < minimum) {
                minimum = score[i][j];
            }
            if (score[i][j] > maximum) {

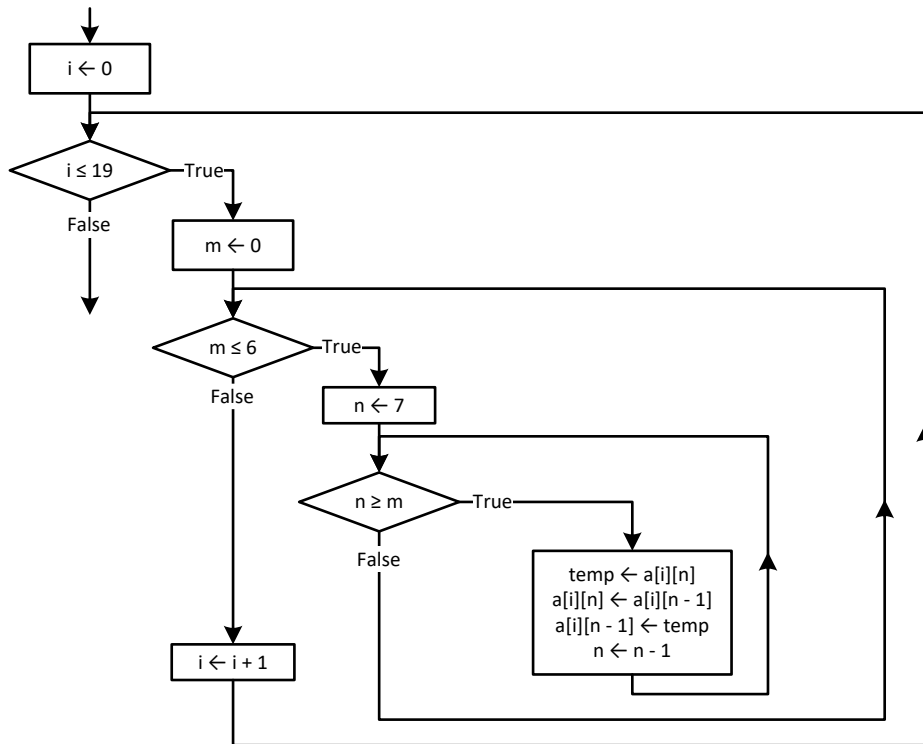
```

```
        maximum = score[i][j];
    }
}
total[i] = total[i] - minimum - maximum;
cout << total[i] << endl;
}

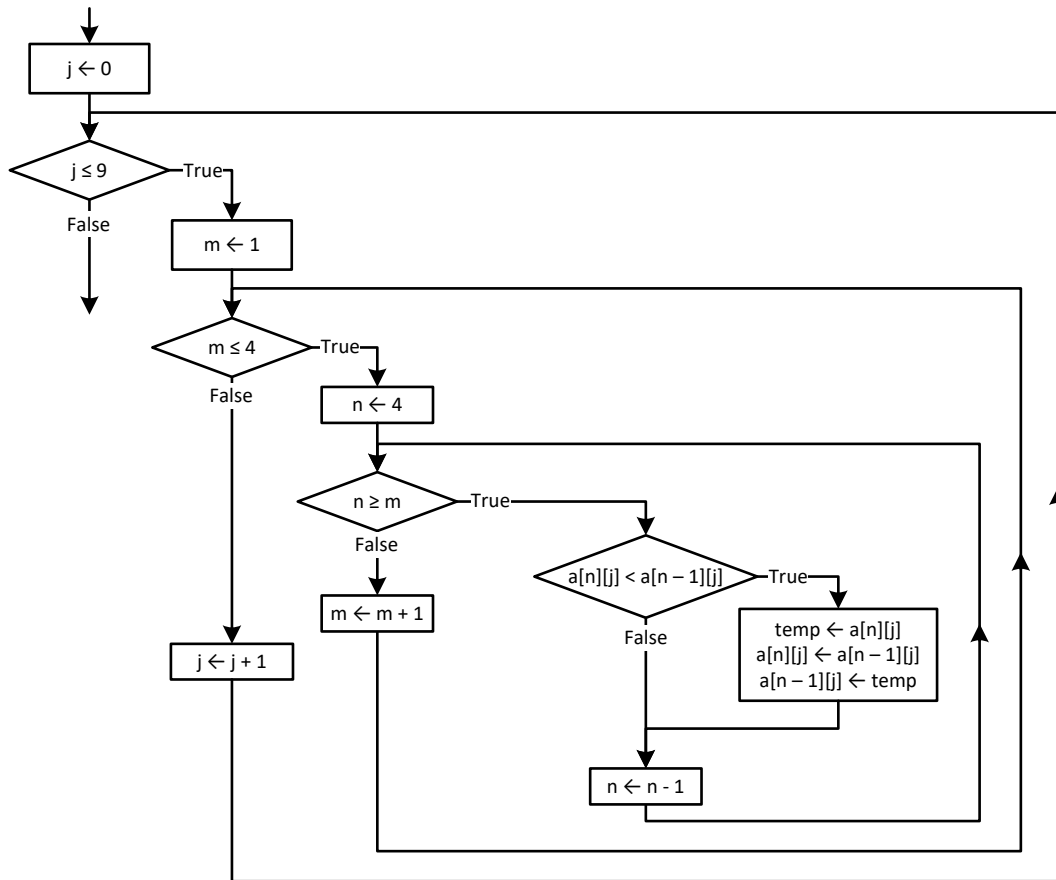
for (m = 1; m <= ARTISTS - 1; m++) {
    for (n = ARTISTS - 1; n >= m; n--) {
        if (total[n] > total[n - 1]) {
            temp = total[n];
            total[n] = total[n - 1];
            total[n - 1] = temp;

            tempStr = artistNames[n];
            artistNames[n] = artistNames[n - 1];
            artistNames[n - 1] = tempStr;
        }
        else if (total[n] == total[n - 1]) {
            if (artistNames[n] < artistNames[n - 1]) {
                tempStr = artistNames[n];
                artistNames[n] = artistNames[n - 1];
                artistNames[n - 1] = tempStr;
            }
        }
    }
}

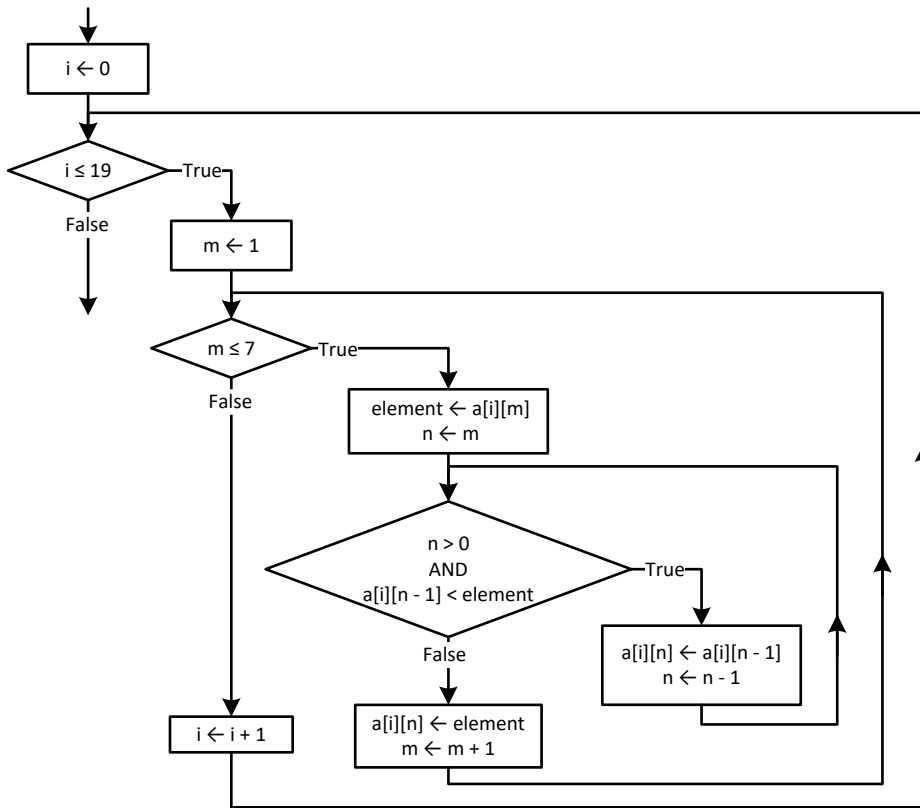
for (i = 0; i <= ARTISTS - 1; i++) {
    cout << artistNames[i] << ", " << total[i] << endl;
}
return 0;
}
```

22. Solution

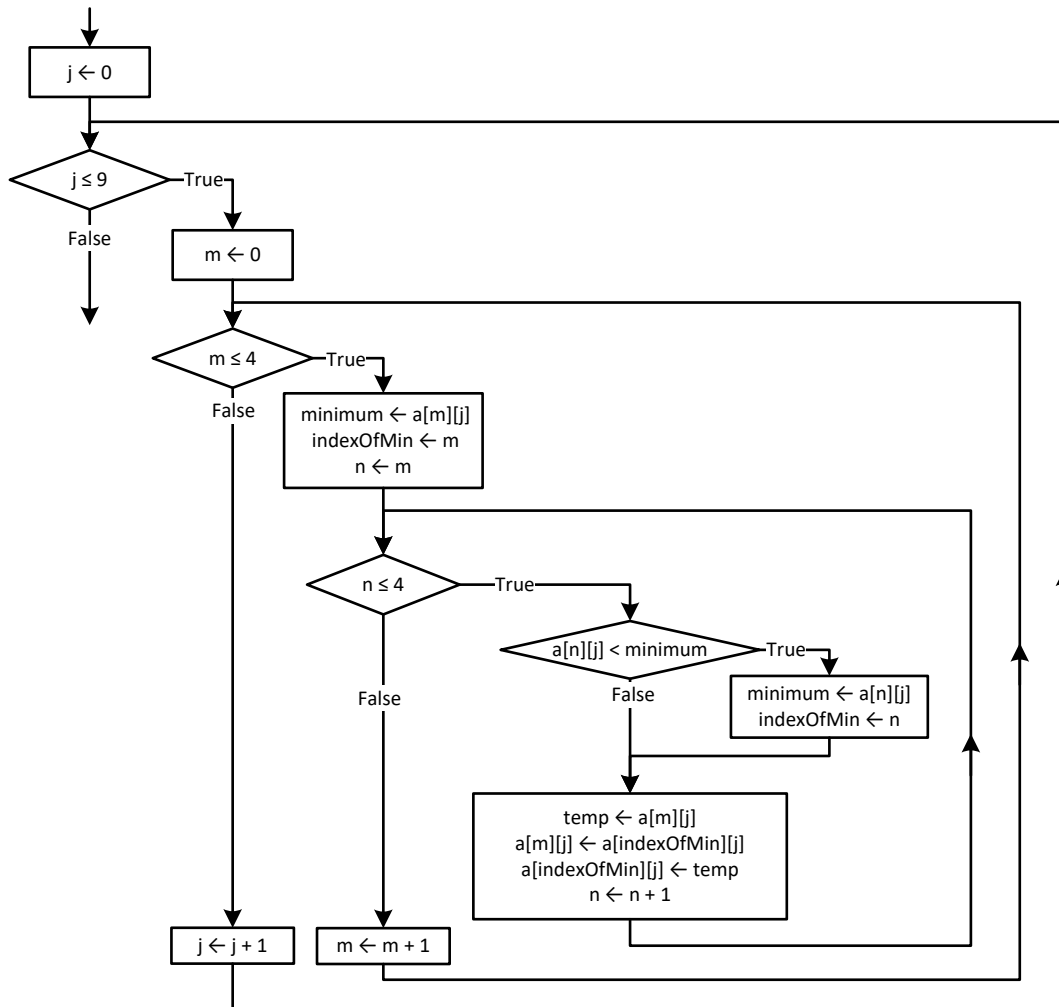
23. Solution



24. Solution



25. Solution



26. Solution

```

#include <iostream>
using namespace std;
const int PEOPLE = 10;
const int PUZZLES = 8;

int main() {
    int i, indexOfMin, j, m, n, hours, minutes, seconds, iTemp;
    double minimum, dTemp;
    string tempStr;

    string names[PEOPLE];
    int times[PEOPLE][PUZZLES];
    for (i = 0; i <= PEOPLE - 1; i++) {
        cin >> names[i];
        for (j = 0; j <= PUZZLES - 1; j++) {
            cin >> hours;

```

```

    cin >> minutes;
    cin >> seconds;
    times[i][j] = hours * 3600 + minutes * 60 + seconds;
}
}

for (i = 0; i <= PEOPLE - 1; i++) {
    for (m = 0; m <= PUZZLES - 1; m++) {
        minimum = times[i][m];
        indexOfMin = m;
        for (n = m; n <= PUZZLES - 1; n++) {
            if (times[i][n] < minimum) {
                minimum = times[i][n];
                indexOfMin = n;
            }
        }
        iTemp = times[i][m];
        times[i][m] = times[i][indexOfMin];
        times[i][indexOfMin] = iTemp;
    }
}

for (i = 0; i <= PEOPLE - 1; i++) {
    cout << names[i] << endl;
    for (j = 0; j <= 2; j++) {
        cout << times[i][j] << endl;
    }
}

double average[PEOPLE];
for (i = 0; i <= PEOPLE - 1; i++) {
    average[i] = 0;
    for (j = 0; j <= PUZZLES - 1; j++) {
        average[i] += times[i][j];
    }
    average[i] /= PUZZLES;
}

for (m = 0; m <= 2; m++) { //Perform only 3 iterations
    minimum = average[m];
    indexOfMin = m;
    for (n = m; n <= PEOPLE - 1; n++) {
        if (average[n] < minimum) {
            minimum = average[n];
            indexOfMin = n;
        }
    }
    dTemp = average[m];
    average[m] = average[indexOfMin];
    average[indexOfMin] = dTemp;

    tempStr = names[m];
}

```

```

    names[m] = names[indexOfMin];
    names[indexOfMin] = tempStr;
}

cout << names[0] << ", " << names[1] << ", " << names[2] << endl;
return 0;
}

```

27. Solution

```

#include <iostream>
using namespace std;
const int AREAS = 5;
const int HOURS = 48;

int main() {
    int i, j, m, m_i, m_j, n;
    double maximum, element1;
    string element2;

    string names[AREAS];
    double CO2[AREAS][HOURS];
    for (i = 0; i <= AREAS - 1; i++) {
        cin >> names[i];
        for (j = 0; j <= HOURS - 1; j++) {
            cin >> CO2[i][j];
        }
    }

    double averagePerHour[AREAS];
    for (i = 0; i <= AREAS - 1; i++) {
        averagePerHour[i] = 0;
        for (j = 0; j <= HOURS - 1; j++) {
            averagePerHour[i] += CO2[i][j];
        }
        averagePerHour[i] /= HOURS;
    }

    for (i = 0; i <= AREAS - 1; i++) {
        cout << names[i] << ", " << averagePerHour[i] << endl;
    }

    double averagePerCity[HOURS];
    for (j = 0; j <= HOURS - 1; j++) {
        averagePerCity[j] = 0;
        for (i = 0; i <= AREAS - 1; i++) {
            averagePerCity[j] += CO2[i][j];
        }
        averagePerCity[j] /= AREAS;
    }

    for (j = 0; j <= HOURS - 1; j++) {
        cout << averagePerCity[j] << endl;
    }
}

```



```
}

maximum = averagePerCity[0];
m_j = 0;
for (j = 1; j <= HOURS - 1; j++) {
    if (averagePerCity[j] > maximum) {
        maximum = averagePerCity[j];
        m_j = j;
    }
}
cout << m_j << endl;

maximum = CO2[0][0];
m_i = 0;
m_j = 0;
for (i = 0; i <= AREAS - 1; i++) {
    for (j = 0; j <= HOURS - 1; j++) {
        if (CO2[i][j] > maximum) {
            maximum = CO2[i][j];
            m_i = i;
            m_j = j;
        }
    }
}
cout << m_j << ", " << names[m_i] << endl;

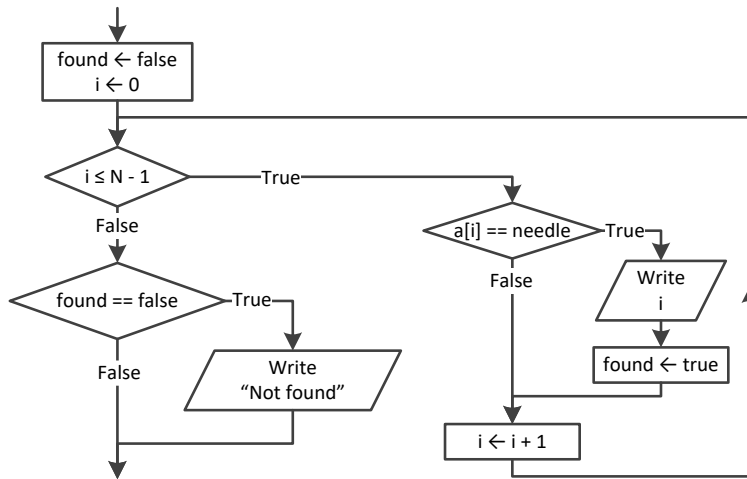
for (m = 1; m <= AREAS - 1; m++) {
    element1 = averagePerHour[m];
    element2 = names[m];

    n = m;
    while (n > 0 && averagePerHour[n - 1] < element1) {
        averagePerHour[n] = averagePerHour[n - 1];
        names[n] = names[n - 1];
        n--;
    }

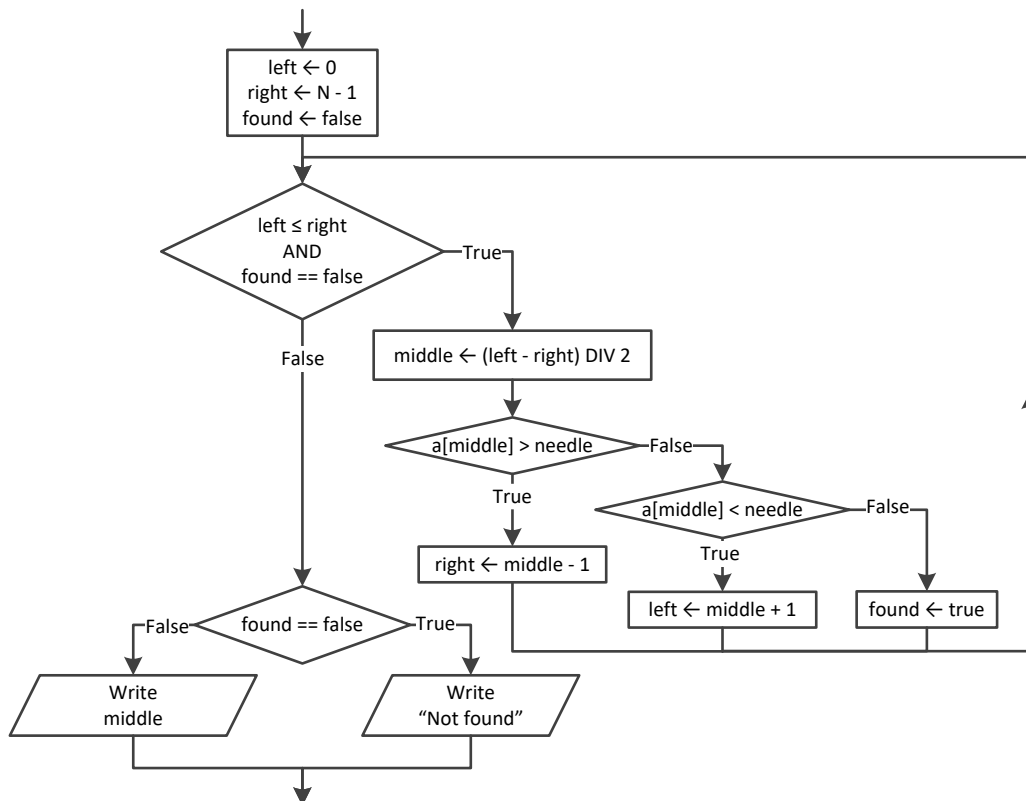
    averagePerHour[n] = element1;
    names[n] = element2;
}

cout << names[0] << ", " << names[1] << ", " << names[2] << endl;
return 0;
}
```

28. Solution



29. Solution



30. Solution

```

#include <iostream>
#include <boost/algorithm/string.hpp>
using namespace boost::algorithm;
using namespace std;

```

```

const int TEAMS = 20;
const int WEEKS = 12;

int main() {
    int i, j;
    string needle;
    bool found;

    string names[TEAMS];
    string results[TEAMS][WEEKS];
    for (i = 0; i <= TEAMS - 1; i++) {
        cout << "Enter name for team No. " << i + 1 << ": ";
        cin >> names[i];
        for (j = 0; j <= WEEKS - 1; j++) {
            cout << "Enter result for";
            cout << " week No. " << j + 1 << " for " << names[i] << ": ";
            cin >> results[i][j];
        }
    }

    //Get value to search and convert it to uppercase
    cout << "Enter a result to search: ";
    cin >> needle;
    needle = to_upper_copy(needle);

    for (i = 0; i <= TEAMS - 1; i++) {
        found = false;
        cout << "Found results for " << names[i] << endl;
        for (j = 0; j <= WEEKS - 1; j++) {
            if (to_upper_copy(results[i][j]) == needle) {
                cout << "Week " << j + 1 << endl;
                found = true;
            }
        }

        if (!found) {
            cout << "No results!" << endl;
        }
    }
    return 0;
}

```

31. Solution

```

#include <iostream>
using namespace std;
const int TEAMS = 10;
const int GAMES = 16;

int main() {
    int i, j, total;
    string needle;

```

```

string names[TEAMS];
int goalsScored[TEAMS][GAMES];
int goalsLetIn[TEAMS][GAMES];
for (i = 0; i <= TEAMS - 1; i++) {
    cout << "Enter team name: ";
    cin >> names[i];
    for (j = 0; j <= GAMES - 1; j++) {
        cout << "Enter goals scored: ";
        cin >> goalsScored[i][j];
        while (goalsScored[i][j] < 0) {
            cout << "Error! Enter goals scored: ";
            cin >> goalsScored[i][j];
        }

        cout << "Enter goals let in: ";
        cin >> goalsLetIn[i][j];
        while (goalsLetIn[i][j] < 0) {
            cout << "Error! Enter goals let in: ";
            cin >> goalsLetIn[i][j];
        }
    }
}

cout << "Enter a team to search: ";
cin >> needle;

i = 0;
while (i < TEAMS - 1 && names[i] != needle) {
    i++;
}

if (names[i] != needle) {
    cout << "This team does not exist" << endl;
}
else {
    total = 0;
    for (j = 0; j <= GAMES - 1; j++){
        if (goalsScored[i][j] > goalsLetIn[i][j]) {
            total += 3;
        }
        else if (goalsScored[i][j] == goalsLetIn[i][j]) {
            total += 1;
        }
    }
    cout << total << endl;
}
return 0;
}

```

32. Solution

```
#include <iostream>
```

```
using namespace std;
const int CLASS1 = 20;
const int CLASS2 = 25;

int main() {
    int i, left, m, middle, n, right;
    string temp, needle;
    bool found;

    cout << "Class 1" << endl;
    string names1[CLASS1];
    for (i = 0; i <= CLASS1 - 1; i++) {
        cout << "Enter name: ";
        cin >> names1[i];
    }
    cout << "Class 2" << endl;
    string names2[CLASS2];
    for (i = 0; i <= CLASS2 - 1; i++) {
        cout << "Enter name: ";
        cin >> names2[i];
    }

    //Bubble sort
    for (m = 1; m <= CLASS1 - 1; m++) {
        for (n = CLASS1 - 1; n >= m; n--) {
            if (names1[n] < names1[n - 1]) {
                temp = names1[n];
                names1[n] = names1[n - 1];
                names1[n] = temp;
            }
        }
    }
    for (m = 1; m <= CLASS2 - 1; m++) {
        for (n = CLASS2 - 1; n >= m; n--) {
            if (names2[n] < names2[n - 1]) {
                temp = names2[n];
                names2[n] = names2[n - 1];
                names2[n] = temp;
            }
        }
    }

    cout << "\nClass 1" << endl;
    for (i = 0; i <= CLASS1 - 1; i++) {
        cout << names1[i] << endl;
    }
    cout << "\nClass 2" << endl;
    for (i = 0; i <= CLASS2 - 1; i++) {
        cout << names2[i] << endl;
    }

    cout << "Enter a name to search: ";
```

```
cin >> needle;

left = 0;
right = CLASS1 - 1;
found = false;
while (left <= right && !found) {
    middle = (int)((left + right) / 2);

    if (needle < names1[middle]) {
        right = middle - 1;
    }
    else if (needle > names1[middle]) {
        left = middle + 1;
    }
    else {
        found = true;
    }
}

if (found) {
    cout << "Student found in Class No 1" << endl;
}
else {
    left = 0;
    right = CLASS2 - 1;
    while (left <= right && !found) {
        middle = (int)((left + right) / 2);

        if (needle < names2[middle]) {
            right = middle - 1;
        }
        else if (needle > names2[middle]) {
            left = middle + 1;
        }
        else {
            found = true;
        }
    }

    if (found) {
        cout << "Student found in Class No 2" << endl;
    }
    else {
        cout << "Student not found in either class" << endl;
    }
}
return 0;
}
```

33. Solution

```
cout << "Enter username: ";
```

```

cin >> usr;
usr = to_upper_copy(usr);
cout << "Enter password: ";
cin >> pwd;
pwd = to_upper_copy(pwd);

i = 0;
while (i < 99 && to_upper_copy(usernames[i]) != usr) {
    i++;
}

if (to_upper_copy(usernames[i]) == usr && to_upper_copy(passwords[i]) == pwd) {
    cout << "Login OK!" << endl;
}
else {
    cout << "Login Failed!" << endl;
}

```

34. Solution

```

cout << "Enter a value to search: ";
cin >> valueStr;

found = false;

for (i = 0; i <= 999; i++) {
    if (names[i] == valueStr) {
        cout << SSNs[i] << endl;
        found = true;
    }
}

if (!found) {
    value = stoi(valueStr);
    i = 0;
    while (i < 999 && SSNs[i] != value) {
        i++;
    }

    if (SSNs[i] == value) {
        found = true;
        cout << names[i] << endl;
    }
}

if (!found) {
    cout << "This value does not exist" << endl;
}

```

35. Solution

```

#include <iostream>
using namespace std;

```

```

const int STUDENTS = 12;
const int LESSONS = 6;

int main() {
    int i, j;
    bool found, failure;

    int grades[STUDENTS][LESSONS];
    for (i = 0; i <= STUDENTS - 1; i++) {
        for (j = 0; j <= LESSONS - 1; j++) {
            do {
                cin >> grades[i][j];
                failure = false;
                if (grades[i][j] < 0) {
                    cout << "Error! You entered a negative value" << endl;
                    failure = true;
                }
                else if (grades[i][j] > 100) {
                    cout << "Error! You entered a value grater than 100" << endl;
                    failure = true;
                }
            } while (failure);
        }
    }

    double average[STUDENTS];
    for (i = 0; i <= STUDENTS - 1; i++) {
        average[i] = 0;
        for (j = 0; j <= LESSONS - 1; j++) {
            average[i] += grades[i][j];
        }
        average[i] /= LESSONS;
    }

    found = false;
    for (i = 0; i <= STUDENTS - 1; i++) {
        if (average[i] < 70) {
            found = true;
            break;
        }
    }

    if (found) {
        cout << "There is at least one student that has an average value below 70" << endl;
    }
    return 0;
}

```

36. Solution

```

#include <iostream>
#include <boost/algorithm/string.hpp>

```



```

using namespace boost::algorithm;
#include <unordered_map>
using namespace std;
int main() {
    string word, letter;
    int i;

    unordered_map<string, string> morseAlphabet = {
        {"A", ".-"},
        {"B", "-..."},
        {"C", "-.-."},
        {"D", "-.."},
        {"E", "."},
        {"F", "..-"},
        {"G", "--."},
        {"H", "...."},
        {"I", ".."},
        {"J", ".---"},
        {"K", "-.-"},
        {"L", "-.-."},
        {"M", "--"},
        {"N", "-."},
        {"O", "---"},
        {"P", ".--"},
        {"Q", "--.-"},
        {"R", "-.-"},
        {"S", "..."},
        {"T", "-"},
        {"U", "..-"},
        {"V", "...-"},
        {"W", "--"},
        {"X", "-.-.-"},
        {"Y", "-.-.-"},
        {"Z", "--.-"},
        {" ", "/" }
    };

    cout << "Enter an English message: ";
    cin >> word;

    for (i = 0; i <= word.length() - 1; i++) {
        letter = word[i];
        cout << morseAlphabet[to_upper_copy(letter)] << " ";
    }
    return 0;
}

```

37. Solution

```

#include <iostream>
#include <unordered_map>

```

```
#include <boost/algorithm/string.hpp>
using namespace boost::algorithm;
using namespace std;
int main() {
    int countSpaces, countExistingLetters, countNonExistingLetters;
    int countUserProvidedCharacters, countNonAlphabeticCharacters;
    string sentence, letter;
    string alphabet = "ABCDEFGHIJKLMNOPQRSTUVWXYZ";

    //Create an unordered map to store the frequencies of each letter with initial
    //frequencies all set to zero.
    unordered_map<string, int> lettersFrequency;
    for (const auto& character : alphabet) {
        letter = character;
        lettersFrequency[letter] = 0;
    }

    cout << "Enter an English sentence: ";
    getline(cin, sentence);

    //Iterate through the characters of the user-provided sentence and if it is a letter,
    //update (increase) the corresponding frequency count in the lettersFrequency unordered map.
    //Also count number of space characters and existing letters
    countSpaces = 0;
    countExistingLetters = 0;
    for (const auto& character : to_upper_copy(sentence)) {
        letter = character;
        if (lettersFrequency.count(letter) == 1) {
            lettersFrequency[letter]++;
            countExistingLetters++;
        }
        else if (letter == " ") {
            countSpaces++;
        }
    }

    //Display the frequency of each existing letter
    for (const auto& element : lettersFrequency) {
        if (element.second > 0) {
            cout << element.first << ": " << element.second << endl;
        }
    }

    //Count and display all non existing letters
    countNonExistingLetters = 0;
    for (const auto& element : lettersFrequency) {
        if (element.second == 0) {
            countNonExistingLetters += 1;
            cout << element.first << endl;
        }
    }
}
```

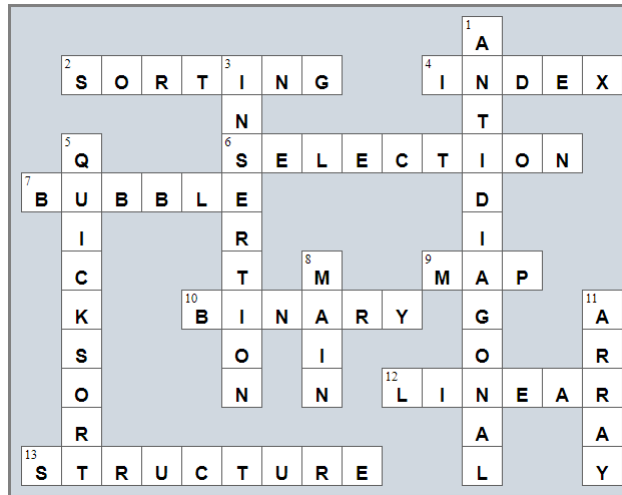
```
//Display percentage of letters that do not exist in relation to the letters of the English alphabet
cout << countNonExistingLetters * 100.0 / 26 << "%" << endl;

//Display percentage of non-alphabetic characters in relation to the characters of
//the user-provided sentence (excluding space characters)
countUserProvidedCharacters = sentence.length() - countSpaces;
countNonAlphabeticCharacters = countUserProvidedCharacters - countExistingLetters;
cout << countNonAlphabeticCharacters * 100.0 / countUserProvidedCharacters << "%" << endl;
return 0;
}
```

Review in "Data Structures in C#"

Review Crossword Puzzle

1.



Chapter 35

35.4 Review Questions: True/False

1. true
2. true
3. false
4. false
5. true
6. true
7. true
8. true
9. true
10. true
11. false
12. true

Chapter 36

36.8 Review Questions: True/False

- | | |
|-----------|-----------|
| 1. true | 18. true |
| 2. true | 19. false |
| 3. false | 20. true |
| 4. true | 21. true |
| 5. true | 22. true |
| 6. false | 23. true |
| 7. true | 24. true |
| 8. false | 25. false |
| 9. true | 26. true |
| 10. false | 27. false |
| 11. true | 28. true |
| 12. true | 29. false |
| 13. true | 30. true |
| 14. true | 31. true |
| 15. true | 32. true |
| 16. false | 33. true |
| 17. false | 34. false |

36.9 Review Exercises

1. Solution

```
int findMax(int a, int b) {
    int maximum;
    if (a > b) {
        maximum = a;
    }
    else {
        maximum = b;
    }
    return maximum;
}
```

2. Solution

Step	Statement	Main Code		Method sumDigits()		
		s	i	a	d1	d2
1	s = 0	0	?			
2	i = 25	0	25			
3	i <= 27	true				
4	s += sumDigits(i)			25	?	?
5	d1 = a % 10			25	5	?

6	d2 = (int)(a / 10)			25	5	2
7	return d1 + d2	7	25			
8	i++	7	26			
9	i <= 27	true				
10	s += sumDigits(i)			26	?	?
11	d1 = a % 10			26	6	?
12	d2 = (int)(a / 10)			26	6	2
13	return d1 + d2	15	26			
14	i++	15	27			
15	i <= 27	true				
16	s += sumDigits(i)			27	?	?
17	d1 = a % 10			27	7	?
18	d2 = (int)(a / 10)			27	7	2
19	return d1 + d2	24	27			
20	i++	24	28			
21	i <= 27	false				
22	cout << s << endl	It displays: 24				

3. Solution

Step	Statement	Main Code		Method sss()		
		s	i	a	total	k
1	i = 1	?	1			
2	s = 0	0	1			
3	while(i < 6)	true				
4	if (i % 2 == 1)	true				
5	s += 1	1	1			
6	i++	1	2			
7	while(i < 6)	true				
8	if (i % 2 == 1)	false				
9	s += sss(i)			2	?	?
10	total = 0			2	0	?
11	k = 1			2	0	1
12	k <= a			true		
13	total += k			2	1	1
14	k++			2	1	2
15	k <= a			true		
16	total += k			2	3	2

17	k++			2	3	3
18	k <= a			false		
19	return total	4	2			
20	i++	4	3			
21	while(i < 6)	true				
22	if (i % 2 == 1)	true				
23	s += 1	5	3			
24	i++	5	4			
25	while(i < 6)	true				
26	if (i % 2 == 1)	false				
27	s += sss(i)			4	?	?
28	total = 0			4	0	?
29	k = 1			4	0	1
30	k <= a			true		
31	total += k			4	1	1
32	k++			4	1	2
33	k <= a			true		
34	total += k			4	3	2
35	k++			4	3	3
36	k <= a			true		
37	total += k			4	6	4
38	k++			4	6	4
39	k <= a			true		
40	total += k			4	10	4
41	k++			4	10	5
42	k <= a			false		
43	return total	15	4			
44	i++	15	5			
45	while(i < 6)	true				
46	if (i % 2 == 1)	true				
47	s += 1	16	5			
48	i++	16	6			
49	while(i < 6)	false				
50	cout << s << endl	It displays: 16				

4. Solution

Step	Statement	Main Code				Method customDiv()	
		k	m	a	x	b	d
1	cin >> k	12	?	?	?		
2	m = 2	12	2	?	?		
3	a = 1	12	2	1	?		
4	while (a < 6)	true					
5	if (k % m != 0)	false					
6	x = a + m + customDiv(m, a)					2	1
7	return (int)((b + d) / 2)	12	2	1	4		
8	cout << m << " " << a << " " << x << endl	It displays: 2 1 4					
9	a += 2	12	2	3	4		
10	m++	12	3	3	4		
11	while (a < 6)	true					
12	if (k % m != 0)	false					
13	x = a + m + customDiv(m, a)					3	3
14	return (int)((b + d) / 2)	12	3	3	9		
15	cout << m << " " << a << " " << x << endl	It displays: 3 3 9					
16	a += 2	12	3	5	9		
17	m++	12	4	5	9		
18	while (a < 6)	true					
19	if (k % m != 0)	false					
20	x = a + m + customDiv(m, a)					4	5
21	return (int)((b + d) / 2)	12	4	5	13		
22	cout << m << " " << a << " " << x << endl	It displays: 4 5 13					
23	a += 2	12	4	7	13		
24	m++	12	5	7	13		
25	while (a < 6)	false					

5. Solution

Step	Statement	Main Code		void Method display()
		i	x	a
1	i = 1	1	?	
2	i <= 5	true		
3	cin >> x	1	3	

4	display(x)			3
5	if (a % 2 == 0)			false
6	cout << a << " is odd" << endl	It displays: 3 is odd		
7	i++	2	3	
8	i <= 5	true		
9	cin >> x	2	7	
10	display(x)			7
11	if (a % 2 == 0)			false
12	cout << a << " is odd" << endl	It displays: 7 is odd		
13	i++	3	7	
14	i <= 5	true		
15	cin >> x	3	9	
16	display(x)			9
17	if (a % 2 == 0)			false
18	cout << a << " is odd" << endl	It displays: 9 is odd		
19	i++	4	9	
20	i <= 5	true		
21	cin >> x	4	2	
22	display(x)			2
23	if (a % 2 == 0)			true
24	cout << a << " is even" << endl	It displays: 2 is even		
25	i++	5	2	
26	i <= 5	true		
27	cin >> x	5	4	
28	display(x)			4
29	if (a % 2 == 0)			true
30	cout << a << " is even" << endl	It displays: 4 is even		
31	i++	6	4	
32	i <= 5	false		

6. Solution

Step	Statement	Main Code		void Method division()	
		x	y	a	b
1	x = 20	20	?		
2	y = 30	20	30		
3	while (x % y < 30)	true			

4	division(y, x)			30	20
5	b = (int)(b / a)			30	0
6	cout << a * b << endl	It displays: 0			
7	x = 4 * y	120	30		
8	y++	120	31		
9	while (x % y < 30)	true			
10	division(y, x)			31	120
11	b = (int)(b / a)			31	3
12	cout << a * b << endl	It displays: 93			
13	x = 4 * y	124	31		
14	y++	124	32		
15	while (x % y < 30)	true			
16	division(y, x)			32	124
17	b = (int)(b / a)			32	3
18	cout << a * b << endl	It displays: 96			
19	x = 4 * y	128	32		
20	y++	128	33		
21	while (x % y < 30)	true			
22	division(y, x)			33	128
23	b = (int)(b / a)			33	3
24	cout << a * b << endl	It displays: 99			
25	x = 4 * y	132	33		
26	y++	132	34		
27	while (x % y < 30)	false			

7. Solution

Step	Statement	Main Code		void Method calculate ()		
		i	m	n	s	j
1	i = 1	1	?			
2	i <= 3	true				
3	cin >> m	1	2			
4	calculate(m)			2	?	?
5	s = 0			2	0	?
6	j = 2			2	0	2
7	j <= 2 * n			true		
8	s = s + pow(j, 2)			2	4	2

9	<code>j += 2</code>			2	4	4
10	<code>j <= 2 * n</code>			true		
11	<code>s = s + pow(j, 2)</code>			2	20	4
12	<code>j += 2</code>			2	20	6
13	<code>j <= 2 * n</code>			false		
14	<code>cout << s << endl</code>	It displays: 20				
15	<code>i++</code>	2	2			
16	<code>i <= 3</code>	true				
17	<code>cin >> m</code>	2	3			
18	<code>calculate(m)</code>			3	?	?
19	<code>s = 0</code>			3	0	?
20	<code>j = 2</code>			3	0	2
21	<code>j <= 2 * n</code>			true		
22	<code>s = s + pow(j, 2)</code>			3	4	2
23	<code>j += 2</code>			3	4	4
24	<code>j <= 2 * n</code>			true		
25	<code>s = s + pow(j, 2)</code>			3	20	4
26	<code>j += 2</code>			3	20	6
27	<code>j <= 2 * n</code>			true		
28	<code>s = s + pow(j, 2)</code>			3	56	6
29	<code>j += 2</code>			3	56	8
30	<code>j <= 2 * n</code>			false		
31	<code>cout << s << endl</code>	It displays: 56				
32	<code>i++</code>	3	3			
33	<code>i <= 3</code>	true				
34	<code>cin >> m</code>	3	4			
35	<code>calculate(m)</code>			4	?	?
36	<code>s = 0</code>			4	0	?
37	<code>j = 2</code>			4	0	2
38	<code>j <= 2 * n</code>			true		
39	<code>s = s + pow(j, 2)</code>			4	4	2
40	<code>j += 2</code>			4	4	4
41	<code>j <= 2 * n</code>			true		
42	<code>s = s + pow(j, 2)</code>			4	20	4
43	<code>j += 2</code>			4	20	6
44	<code>j <= 2 * n</code>			true		
45	<code>s = s + pow(j, 2)</code>			4	56	6

46	<code>j += 2</code>			4	56	8
47	<code>j <= 2 * n</code>				true	
48	<code>s = s + pow(j, 2)</code>			4	120	8
49	<code>j += 2</code>			4	120	10
50	<code>j <= 2 * n</code>				false	
51	<code>cout << s << endl</code>	It displays: 120				
52	<code>i++</code>	4	4			
53	<code>i <= 3</code>	false				

8. Solution

```
int findSum(int a, int b, int c) {
    return a + b + c;
}
```

9. Solution

```
double findAvg(double a, double b, double c, double d) {
    return (a + b + c + d) / 4;
}
```

10. Solution

```
double maximum(double a, double b, double c) {
    double m;

    m = a;
    if (b > m) {
        m = b;
    }
    if (c > m) {
        m = c;
    }
    return m;
}
```

11. Solution

```
void displayMax(double a, double b, double c, double d, double e) {
    double m;

    m = a;
    if (b > m) {
        m = b;
    }
    if (c > m) {
        m = c;
    }
    if (d > m) {
```

```
    m = d;
}
if (e > m) {
    m = e;
}
cout << m << endl;
}
```

12. Solution

```
double myRound(double x) {
    int digitToCheck;
    double returnValue;

    digitToCheck = (int)(x * 1000) % 10;
    if (digitToCheck >= 5) {
        returnValue = ((int)(x * 100) + 1) / 100.0;
    }
    else {
        returnValue = ((int)(x * 100)) / 100.0;
    }

    return returnValue;
}
```

13. Solution

```
#include <iostream>
using namespace std;

double findMin(double a, double b) {
    double minimum;

    minimum = a;
    if (b < minimum) {
        minimum = b;
    }
    return minimum;
}

int main() {
    double temp1, temp2, x1, x2, x3, x4;

    cout << "Enter four numbers: " << endl;
    cin >> x1 >> x2 >> x3 >> x4;

    //First approach
    temp1 = findMin(x1, x2);
    temp2 = findMin(x3, x4);
    cout << findMin(temp1, temp2) << endl;

    //Second approach
    cout << findMin(findMin(x1, x2), findMin(x3, x4)) << endl;
}
```

```
    return 0;
}
```

14. Solution

```
#include <iostream>
using namespace std;

double KelvinToFahrenheit(double kelvin) {
    return 1.8 * kelvin - 459.67;
}

double KelvinToCelsius(double kelvin) {
    return kelvin - 273.15;
}

int main() {
    double k;

    cout << "Enter a temperature in degrees Kelvin: ";
    cin >> k;
    cout << "Fahrenheit: " << KelvinToFahrenheit(k) << endl;
    cout << "Celsius: " << KelvinToCelsius(k) << endl;
    return 0;
}
```

15. Solution

```
#include <iostream>
#include <cmath>
using namespace std;

string bmi(double w, double h) {
    double b;
    string returnValue;

    b = w * 703 / pow(h, 2);
    if (b < 16) {
        returnValue = "You must add weight.";
    }
    else if (b < 18.5) {
        returnValue = "You should add some weight.";
    }
    else if (b < 25) {
        returnValue = "Maintain your weight.";
    }
    else if (b < 30) {
        returnValue = "You should lose some weight.";
    }
    else {
        returnValue = "You must lose weight.";
    }
}
```

```
    return returnValue;
}

int main() {
    double height, weight;
    int age;

    cout << "Enter your weight (in pounds): ";
    cin >> weight;
    while (weight < 0) {
        cout << "Error! Enter your weight (in pounds): ";
        cin >> weight;
    }

    cout << "Enter your age: ";
    cin >> age;
    while (age < 18) {
        cout << "Error! Enter your age: ";
        cin >> age;
    }

    cout << "Enter your height (in inches): ";
    cin >> height;
    while (height < 0) {
        cout << "Error! Enter your height (in inches): ";
        cin >> height;
    }

    cout << bmi(weight, height) << endl;
    return 0;
}
```

16. Solution

```
#include <iostream>
using namespace std;

void numOfDay(int year, int month) {
    int days;

    switch (month) {
        case 4:
        case 6:
        case 9:
        case 11:
            days = 30;
            break;
        case 2:
            if (year % 4 == 0 && year % 100 != 0 || year % 400 == 0) {
                days = 29;
            }
            else {
                days = 28;
            }
    }
}
```



```
    }
    break;
default:
    days = 31;
}

cout << days << endl;
}

int main() {
    int m, y;

    cout << "Enter a year: ";
    cin >> y;
    for (m = 1; m <= 12; m++) {
        numOfDays(y, m);
    }
    return 0;
}
```

17. Solution

```
#include <iostream>
using namespace std;

int numOfDays(int year, int month) {
    int days;

    switch (month) {
        case 4:
        case 6:
        case 9:
        case 11:
            days = 30;
            break;
        case 2:
            if (year % 4 == 0 && year % 100 != 0 || year % 400 == 0) {
                days = 29;
            }
            else {
                days = 28;
            }
            break;
        default:
            days = 31;
    }
    return days;
}

int main() {
    int y, m1, m2, m, total;

    cout << "Enter a year: ";
```

```
cin >> y;
cout << "Enter a month: ";
cin >> m1;
cout << "Enter a second month: ";
cin >> m2;

total = 0;
for (m = m1; m <= m2; m++) {
    total += numOfDay(y, m);
}
cout << total << endl;
return 0;
}
```

18. Solution

```
#include <iostream>
using namespace std;

void displayMenu() {
    cout << endl;
    cout << "1. Convert meters to miles" << endl;
    cout << "2. Convert miles to meters" << endl;
    cout << "3. Exit" << endl;
    cout << "Enter a choice: ";
}

void metersToMiles(double meters) {
    cout << meters << " meters equals " << meters / 1609.344 << " miles" << endl;
}

void milesToMeters(double miles) {
    cout << miles << " miles equals " << miles * 1609.344 << " meters" << endl;
}

int main() {
    int choice;
    double distance;

    displayMenu();
    cin >> choice;
    while (choice != 3) {
        cout << "Enter distance: ";
        cin >> distance;
        if (choice == 1) {
            metersToMiles(distance);
        }
        else {
            milesToMeters(distance);
        }

        displayMenu();
        cin >> choice;
    }
}
```

```
    }  
    return 0;  
}
```

19. Solution

```
#include <iostream>  
using namespace std;  
  
void amountToPay(int seconds) {  
    double extra, tax, total, totalWithoutTax;  
  
    if (seconds <= 600) {  
        extra = 0;  
    }  
    else if (seconds <= 1200) {  
        extra = (seconds - 600) * 0.01;  
    }  
    else {  
        extra = 600 * 0.01 + (seconds - 1200) * 0.02;  
    }  
  
    totalWithoutTax = 10 + extra;  
    tax = totalWithoutTax * 11 / 100;  
    total = totalWithoutTax + tax;  
  
    cout << "Total amount to pay: " << total << endl;  
}  
  
int main() {  
    int seconds;  
  
    cout << "Enter number of seconds: ";  
    cin >> seconds;  
    amountToPay(seconds);  
}
```

Chapter 37

37.9 Review Questions: True/False

- | | |
|-----------|-----------|
| 1. true | 13. true |
| 2. true | 14. false |
| 3. true | 15. true |
| 4. false | 16. true |
| 5. true | 17. false |
| 6. false | 18. true |
| 7. true | 19. true |
| 8. false | 20. false |
| 9. true | 21. true |
| 10. false | 22. true |
| 11. true | 23. true |
| 12. true | |

37.10 Review Exercises

1. *Solution*

It displays: 5

2. *Solution*

It displays: 14

3. *Solution*

It displays: 14

4. *Solution*

It displays: hellohellohello

5. *Solution*

It displays: 15

6. *Solution*

It displays: 11 4

7. *Solution*

It displays: 3

8. *Solution*

Within the function `getNumOfDigits()`, variable `x` eventually becomes 0, and since the variable `val` is passed to the function by reference, that 0 also reflects back to the main code. So, when the flow of execution returns to the main code, the value of variable `val` is zeroed!

To resolve this issue, all you have to do is remove the ampersand (&) sign from the beginning of the formal argument `x`. If you do so, the variable `val` is passed to the function by value, so that no matter what happens to variable `x` within the function, nothing can affect the value of the variable `val` of the main code.

9. Solution

```
#include <iostream>
using namespace std;

const int STUDENTS = 10;
const int LESSONS = 5;

void part1(string names[], int grades[][LESSONS]) {
    int i, j;

    for (i = 0; i <= STUDENTS - 1; i++) {
        cout << "Enter name for student No. " << i + 1 << ": ";
        cin >> names[i];
        for (j = 0; j <= LESSONS - 1; j++) {
            cout << "Enter grade for lesson No. " << j + 1 << ": ";
            cin >> grades[i][j];
        }
    }
}

double *part2(int grades[][LESSONS]) {
    static double average[STUDENTS];
    int i, j;

    for (i = 0; i <= STUDENTS - 1; i++) {
        average[i] = 0;
        for (j = 0; j <= LESSONS - 1; j++) {
            average[i] += grades[i][j];
        }
        average[i] /= LESSONS;
    }
    return average;
}

void part3(double average[], string names[]) {
    int m, n;
    double temp;
    string tempStr;

    for (m = 1; m <= STUDENTS - 1; m++) {
        for (n = STUDENTS - 1; n >= m; n--) {
            if (average[n] > average[n - 1]) {
                temp = average[n];
                average[n] = average[n - 1];
                average[n - 1] = temp;

                tempStr = names[n];
                names[n] = names[n - 1];
            }
        }
    }
}
```

```

        names[n - 1] = tempStr;
    }
    else if (average[n] == average[n - 1]) {
        if (names[n] < names[n - 1]) {
            tempStr = names[n];
            names[n] = names[n - 1];
            names[n - 1] = tempStr;
        }
    }
}
}
}

int main() {
    int i;

    string names[STUDENTS];
    int grades[STUDENTS][LESSONS];

    part1(names, grades);

    double *average = part2(grades);

    part3(average, names);

    for (i = 0; i <= STUDENTS - 1; i++) {
        cout << names[i] << "\t" << average[i] << endl;
    }
    return 0;
}

```

10. Solution

```

#include <iostream>
#include <boost/algorithm/string.hpp>
using namespace boost::algorithm;
using namespace std;

string part1() {
    string message;

    cout << "Enter a message: ";
    getline(cin, message);
    message = to_lower_copy(message);
    return message;
}

string part2(string message) {
    string letter, messageClean;
    int i;

    messageClean = "";
    for (i = 0; i <= message.length() - 1; i++) {
        letter = message[i];
    }
}

```

```

        if (letter != " " && letter != "," && letter != "." && letter != "?") {
            messageClean += letter;
        }
    }
    return messageClean;
}

bool part3(string messageClean) {
    int middlePos, i, j;
    bool palindrome;
    string leftLetter, rightLetter;

    middlePos = (int)((messageClean.length() - 1) / 2);
    j = messageClean.length() - 1;
    palindrome = true;
    for (i = 0; i <= middlePos; i++) {
        leftLetter = messageClean[i];
        rightLetter = messageClean[j];
        if (leftLetter != rightLetter) {
            palindrome = false;
            break;
        }
        j--;
    }
    return palindrome;
}

bool part4(string message) {
    string messageClean;
    bool palindrome;

    messageClean = part2(message);
    palindrome = part3(messageClean);
    return palindrome;
}

int main() {
    string message;
    bool palindrome;

    message = part1();
    palindrome = part4(message);
    if (palindrome) {
        cout << "The message is palindrome" << endl;
    }
    return 0;
}

```

11. Solution

```

#include <iostream>
using namespace std;
int main() {

```

```

int a, b, c, d, maximum;

cin >> a >> b >> c >> d;

maximum = a;
if (b > maximum) {
    maximum = b;
}
if (c > maximum) {
    maximum = c;
}
if (d > maximum) {
    maximum = d;
}

cout << maximum << endl;
return 0;
}

```

12. Solution

First approach

```

void f1(double a, double b, double c, double &total, double &average) {
    total = a + b + c;
    average = total / 3;
}

```

Second approach

```

void f1(double a, double b, double c, double returningArray[]) {
    returningArray[0] = a + b + c;
    returningArray[1] = returningArray[0] / 3;
}

```

13. Solution

```

double myRound(double x, int decimalPlaces = 2) {
    double returnValue;

    int digitToCheck = (int)((x * pow(10, decimalPlaces + 1)) % 10);
    if (digitToCheck >= 5) {
        returnValue = ((int)(x * pow(10, decimalPlaces)) + 1) / pow(10, decimalPlaces);
    }
    else {
        returnValue = ((int)(x * pow(10, decimalPlaces))) / pow(10, decimalPlaces);
    }
    return returnValue;
}

```

14. Solution

```

#include <iostream>
#include <boost/algorithm/string.hpp>
using namespace boost::algorithm;

```



```
using namespace std;

bool getInput() {
    string answer;

    do {
        cout << "Enter Yes or No: ";
        cin >> answer;
        answer = to_upper_copy(answer);
    } while (answer != "YES" && answer != "NO");

    return answer == "YES"; //This returns true or false
}

double findArea(double b, double h) {
    return b * h;
}

int main() {
    double b, h;

    do {
        cout << "Enter the base of the parallelogram: ";
        cin >> b;
        cout << "Enter the height of the parallelogram: ";
        cin >> h;

        cout << "Area = " << findArea(b, h) << endl;

        cout << "Would you like to repeat? " << endl;
    } while (getInput());
    return 0;
}
```

15. Solution

```
#include <iostream>
using namespace std;

const int STUDENTS = 100;

void getArrays(string names[], int grades[]) {
    int i;

    for (i = 0; i <= STUDENTS - 1; i++) {
        cout << "Enter name: ";
        cin >> names[i];
        cout << "Enter grade: ";
        cin >> grades[i];
    }
}

double getAverage(int grades[]) {
    int i, total = 0;
```

```

    for (i = 0; i <= STUDENTS - 1; i++) {
        total += grades[i];
    }
    return total / (double)STUDENTS;
}

void sortArrays(int grades[], string names[]) {
    int m, n, elementGrds;
    string elementNms;

    for (m = 1; m <= STUDENTS - 1; m++) {
        elementGrds = grades[m];
        elementNms = names[m];

        n = m;
        while (n > 0 && grades[n - 1] > elementGrds) {
            grades[n] = grades[n - 1];
            names[n] = names[n - 1];
            n--;
        }

        grades[n] = elementGrds;
        names[n] = elementNms;
    }
}

int main() {
    int i;
    double average;

    string names[STUDENTS];
    int grades[STUDENTS];

    getArrays(names, grades);
    average = getAverage(grades);
    sortArrays(grades, names);
    for (i = 0; i <= STUDENTS - 1; i++) {
        if (grades[i] < average) {
            cout << names[i] << endl;
        }
    }
    return 0;
}

```

16. Solution

```

#include <iostream>
using namespace std;

const int JUDGES = 10;

int getArray() {
    int score[JUDGES];

```

```

    int i;

    for (i = 0; i <= JUDGES - 1; i++) {
        cout << "Judge No " << i + 1 << ". Enter score: ";
        cin >> score[i];
    }
    return score;
}

void findMinMax(int score[], int &minimum, int &maximum) {
    int i;
    minimum = score[0];
    maximum = score[0];
    for (i = 1; i <= JUDGES - 1; i++) {
        if (score[i] > maximum) {
            maximum = score[i];
        }
        if (score[i] < minimum) {
            minimum = score[i];
        }
    }
}

int main() {
    string name;
    int total, i, points, minimum = 0, maximum = 0;

    cout << "Enter artist's name: ";
    cin >> name;
    int score[] = getArray();
    findMinMax(score, minimum, maximum);

    total = 0;
    for (i = 0; i <= JUDGES - 1; i++) {
        total += score[i];
    }

    points = total - minimum - maximum;
    cout << "Artist " << name << " got " << points << " points" << endl;
    return 0;
}

```

17. Solution

```

#include <iostream>
using namespace std;

int sumRecursive(int n) {
    if (n == 1) {
        return 1;
    }
    else {
        return sumRecursive(n - 1) + n;
    }
}

```

```
    }  
}  
  
int main() {  
    int num;  
    cin >> num;  
    cout << sumRecursive(num) << endl;  
    return 0;  
}
```

18. Solution

```
#include <iostream>  
using namespace std;  
  
double woc(int index) {  
    if (index == 1) {  
        return 1;  
    }  
    else {  
        return 2 * woc(index - 1);  
    }  
}  
  
int main() {  
    double total;  
    int i;  
  
    total = 0;  
    for (i = 1; i <= 64; i++) {  
        total += woc(i);  
    }  
    cout << total << endl;  
    return 0;  
}
```

19. Solution

```
#include <iostream>  
using namespace std;  
  
int fib(int n) {  
    if (n <= 1) {  
        return n;  
    }  
    else {  
        return fib(n - 1) + fib(n - 2);  
    }  
}  
  
int main() {  
    int num;
```

```

cin >> num;
cout << fib(num - 1) << endl;
return 0;
}

```

20. Solution

```

int tribonacci(int n) {
    if (n == 0) {
        return 0;
    }
    else if (n == 1 || n == 2) {
        return 1;
    }
    else {
        return tribonacci(n - 1) + tribonacci(n - 2) + tribonacci(n - 3);
    }
}

```

21. Solution

```

double myPow(double n, int p) {
    if (p == 0)
        return 1;
    else if (p < 0)
        return 1 / (n * myPow(n, -p - 1));
    else
        return n * myPow(n, p - 1);
}

```

22. Solution

```

#define _USE_MATH_DEFINES //This is necessary for Visual Studio Community IDE
#include <iostream>
#include <cmath>
using namespace std;

double factorial(int value) {
    if (value == 1) {
        return 1;
    }
    else {
        return value * factorial(value - 1);
    }
}

double myCos(double x, int i = 40) {
    if (i == 0) {
        return 1;
    }
    else {
        return myCos(x, i - 4) + pow(x, i) / factorial(i) - pow(x, i - 2) / factorial(i - 2);
    }
}

```

```
    }  
  }  
  
  int main() {  
    cout << myCos(M_PI / 4) << endl;  
  }
```

Chapter 38

38.3 Review Exercises

1. Solution

```
#include <iostream>
using namespace std;

void displayMenu() {
    cout << "1. Convert USD to Euro (EUR)" << endl;
    cout << "2. Convert USD to British Pound Sterling (GBP)" << endl;
    cout << "3. Convert USD to Japanese Yen (JPY)" << endl;
    cout << "4. Convert USD to Canadian Dollar (CAD)" << endl;
    cout << "5. Exit" << endl;
    cout << "-----" << endl;
    cout << "Enter a choice: ";
}

double USD_to_EU(double value) {
    return value * 0.94;
}

double USD_to_GBP(double value) {
    return value * 0.81;
}

double USD_to_JPY(double value) {
    return value * 149.11;
}

double USD_to_CAD(double value) {
    return value * 1.36;
}

int main() {
    int choice;
    double amount;

    displayMenu();
    cin >> choice;
    while (choice != 5) {
        cout << "Enter an amount in US dollars: ";
        cin >> amount;
        switch (choice) {
            case 1:
                cout << amount << " USD = " << USD_to_EU(amount) << " Euro" << endl;
                break;
            case 2:
                cout << amount << " USD = " << USD_to_GBP(amount) << " GBP" << endl;
                break;
            case 3:
```

```

        cout << amount << " USD = " << USD_to_JPY(amount) << " JPY" << endl;
        break;
    case 4:
        cout << amount << " USD = " << USD_to_CAD(amount) << " CAD" << endl;
        break;
    }

    displayMenu();
    cin >> choice;
}
return 0;
}

```

2. Solution

```

#include <iostream>
using namespace std;

void displayMenu() {
    cout << "-----" << endl;
    cout << "1. Convert USD to Euro (EUR)" << endl;
    cout << "2. Convert USD to British Pound Sterling (GBP)" << endl;
    cout << "3. Convert EUR to USD" << endl;
    cout << "4. Convert EUR to GBP" << endl;
    cout << "5. Convert GBP to USD" << endl;
    cout << "6. Convert GBP to EUR" << endl;
    cout << "7. Exit" << endl;
    cout << "-----" << endl;
    cout << "Enter a choice: ";
}

double USDToEUR(double value) {
    return value * 0.94;
}

double USDToGBP(double value) {
    return value * 0.81;
}

int main() {
    int choice;
    double amount;

    displayMenu();
    cin >> choice;
    while (choice != 7) {
        cout << "Enter an amount: ";
        cin >> amount;
        switch (choice) {
            case 1:
                cout << amount << " USD = " << USD_to_EUR(amount) << " Euro" << endl;
                break;
            case 2:

```



```

        cout << amount << " USD = " << USD_to_GBP(amount) << " GBP" << endl;
        break;
    case 3:
        cout << amount << " EUR = " << 1 / USD_to_EUR(1 / amount) << " USD" << endl;
        break;
    case 4:
        cout << amount << " EUR = " << USD_to_GBP(1 / USD_to_EUR(1 / amount)) << " GBP" << endl;
        break;
    case 5:
        cout << amount << " GBP = " << 1 / USD_to_GBP(1 / amount) << " USD" << endl;
        break;
    case 6:
        cout << amount << " GBP = " << USD_to_EUR(1 / USD_to_GBP(1 / amount)) << " EUR" << endl;
        break;
    }

    displayMenu();
    cin >> choice;
}
return 0;
}

```

3. Solution

```

#define _USE_MATH_DEFINES //This is necessary for Visual Studio Community IDE
#include <iostream>
#include <cmath>
using namespace std;

const double ACCURACY = 0.000000001;

double factorial(int n) {
    int i;

    double returnValue = 1;
    for (i = 1; i <= n; i++) {
        returnValue *= i;
    }
    return returnValue;
}

double mySin(double x) {
    int i, sign;
    double sinus, sinusPrevious;
    sign = 1;
    sinus = 0;
    i = 1;
    do {
        sinusPrevious = sinus;
        sinus += sign * pow(x, i) / factorial(i);

        sign = -sign;
        i += 2;
    }
}

```

```
    } while (abs(sinus - sinusPrevious) > ACCURACY);
    return sinus;
}

double degreesToRad(double degrees) {
    return 2 * M_PI * degrees / 360;
}

int main() {
    int i;

    for (i = 0; i <= 360; i++) {
        cout << "sin(" << i << ") ~= " << mySin(degreesToRad(i)) << endl;
    }
    return 0;
}
```

4. Solution

```
#include <iostream>
using namespace std;

bool isLeap(int year) {
    bool returnValue = false;
    if (year % 4 == 0 && year % 100 != 0 || year % 400 == 0) {
        returnValue = true;
    }
    return returnValue;
}

int numOfDay(int year, int month) {
    int days;

    switch (month) {
        case 4:
        case 6:
        case 9:
        case 11:
            days = 30;
            break;
        case 2:
            if (isLeap(year)) {
                days = 29;
            }
            else {
                days = 28;
            }
            break;
        default:
            days = 31;
    }

    return days;
}
```

```
}  
  
bool checkDate(int day, int month, int year) {  
    bool returnValue = true;  
    if (month < 1 || month > 12) {  
        returnValue = false;  
    }  
    else if (day < 1 || day > numOfDays(year, month)) {  
        returnValue = false;  
    }  
    return returnValue;  
}  
  
int main() {  
    int day, month, year, total, i;  
  
    cout << "Enter day: ";  
    cin >> day;  
    cout << "Enter month: ";  
    cin >> month;  
    cout << "Enter year: ";  
    cin >> year;  
    while (!checkDate(day, month, year)) {  
        cout << "Error!" << endl;  
        cout << "Enter day: ";  
        cin >> day;  
        cout << "Enter month: ";  
        cin >> month;  
        cout << "Enter year: ";  
        cin >> year;  
    }  
  
    total = 0;  
    for (i = 1; i <= month - 1; i++) {  
        total += numOfDays(year, i);  
    }  
    total += day;  
  
    cout << total << endl;  
    return 0;  
}
```

5. Solution

```
#include <iostream>  
#include <ctime>  
#include <cstdlib>  
using namespace std;  
  
int dice() {  
    return 1 + rand() % 6;  
}
```

```

int main() {
    int dice1, dice2, i, player, total, totalPlayer1 = 0, totalPlayer2 = 0;
    string key, names[2];

    srand(time(NULL));

    cout << "Player1 - Enter name: ";
    cin >> names[0];
    cout << "Player2 - Enter name: ";
    cin >> names[1];

    for (player = 0; player <= 1; player++) {
        total = 0;
        for (i = 1; i <= 10; i++) {
            cout << names[player] << ", hit enter to roll the dice!" << endl;
            getline(cin, key); //This statement just waits the user to hit the enter key

            dice1 = dice();
            dice2 = dice();
            cout << dice1 << " " << dice2 << endl;
            total += dice1 + dice2;
        }
        if (player == 1) {
            totalPlayer1 = total;
        }
        else {
            totalPlayer2 = total;
        }
    }

    if (totalPlayer1 == totalPlayer2) {
        cout << "Tie!" << endl;
    }
    else if (totalPlayer1 > totalPlayer2) {
        cout << names[0] << " wins" << endl;
    }
    else {
        cout << names[1] << " wins" << endl;
    }
    return 0;
}

```

6. Solution

```

#include <iostream>
using namespace std;

const int GAS = 1;
const int DIESEL = 2;
const int HYBRID = 3;
const double TAX_RATE = 0.10;
const int CARS = 40;

```

```
int getChoice() {
    int choice;
    cout << "1. Gas" << endl;
    cout << "2. Diesel" << endl;
    cout << "3. Hybrid" << endl;
    cout << "Enter type of the car: ";
    cin >> choice;
    return choice;
}

int getDays() {
    int days;
    cout << "Enter total number of rental days: ";
    cin >> days;
    return days;
}

double getCharge(int carType, int rentalDays) {
    double charge;

    if (carType == GAS) {
        if (rentalDays <= 5) {
            charge = rentalDays * 24;
        }
        else if (rentalDays <= 8) {
            charge = 5 * 24 + (rentalDays - 5) * 22;
        }
        else {
            charge = 5 * 24 + 3 * 22 + (rentalDays - 8) * 18;
        }
    }
    else if (carType == DIESEL) {
        if (rentalDays <= 5) {
            charge = rentalDays * 28;
        }
        else if (rentalDays <= 8) {
            charge = 5 * 28 + (rentalDays - 5) * 25;
        }
        else {
            charge = 5 * 28 + 3 * 25 + (rentalDays - 8) * 21;
        }
    }
    else {
        if (rentalDays <= 5) {
            charge = rentalDays * 30;
        }
        else if (rentalDays <= 8) {
            charge = 5 * 30 + (rentalDays - 5) * 28;
        }
        else {
            charge = 5 * 30 + 3 * 28 + (rentalDays - 8) * 23;
        }
    }
}
```

```

    }
    charge = charge * (1 + TAX_RATE); //This is equivalent to charge += charge * TAX_RATE;
    return charge;
}

int main() {
    int count, i;
    double charge, total;

    int rentedCarTypes[CARS];
    int rentedDays[CARS];

    for (i = 0; i <= CARS - 1; i++) {
        rentedCarTypes[i] = getChoice();
        rentedDays[i] = getDays();
    }

    total = 0;
    for (i = 0; i <= CARS - 1; i++) {
        charge = getCharge(rentedCarTypes[i], rentedDays[i]);
        cout << "Car No " << i + 1 << ": " << charge << endl;
        total += charge;
    }

    count = 0;
    for (i = 0; i <= CARS - 1; i++) {
        if (rentedCarTypes[i] == HYBRID) {
            count++;
        }
    }

    cout << "Hybrids rented: " << count << endl;
    cout << "Net profit: " << total / (1 + TAX_RATE) << endl;
    return 0;
}

```

7. Solution

```

#include <iostream>
using namespace std;

const int CHANNELS = 10;
const int DAYS = 7;
const string dayNames[] = {"Monday", "Tuesday", "Wednesday",
                           "Thursday", "Friday", "Saturday", "Sunday"};

//Note that in C++, in order to pass multidimensional arrays to functions,
//the arrays in the formal argument list must have bounds for all dimensions except the first
void getData(string names[], int viewers[][DAYS]) {
    int i, j;

    for (i = 0; i <= CHANNELS - 1; i++) {
        cout << "Enter name for channel No. " << i + 1 << ": ";
    }
}

```

```
    cin >> names[i];
    for (j = 0; j <= DAYS - 1; j++) {
        cout << "Enter the number of viewers of the main news program on " << dayNames[j] <<
            " for channel " << names[i] << ": ";
        cin >> viewers[i][j];
    }
}

double getAverage(int a[]) {
    int total ,i;

    total = 0;
    for (i = 0; i <= 4; i++) {
        total += a[i];
    }
    return total / 5.0;
}

int main() {
    int i, j;
    double weekend;
    bool increasing;

    string names[CHANNELS];
    int viewers[CHANNELS][DAYS];
    getData(names, viewers);

    int temporaryArray[5];
    for (i = 0; i <= CHANNELS - 1; i++) {
        for (j = 0; j <= 4; j++) {
            temporaryArray[j] = viewers[i][j];
        }
        weekend = (viewers[i][DAYS - 2] + viewers[i][DAYS - 1]) / 2;
        if (weekend >= 1.2 * getAverage(temporaryArray)) {
            cout << names[i] << endl;
        }
    }

    for (i = 0; i <= CHANNELS - 1; i++) {
        increasing = true;
        for (j = 1; j <= DAYS - 1; j++) {
            if (viewers[i][j] <= viewers[i][j - 1]) {
                increasing = false;
            }
        }
        if (increasing) {
            cout << names[i] << endl;
        }
    }
    return 0;
}
```

8. Solution

```
#include <iostream>
using namespace std;

const int CITIZENS = 300;

void inputData(long long SSNs[], string answers[]) {
    int i;

    for (i = 0; i <= CITIZENS - 1; i++) {
        cout << "Enter SSN: ";
        cin >> SSNs[i];
        cout << "Enter answer: ";
        cin >> answers[i];
    }
}

void sortArrays(long long SSNs[], string answers[]) {
    int m, n, indexOfMin;
    long long minimum, temp;
    string tempStr;

    for (m = 0; m <= CITIZENS - 1; m++) {
        minimum = SSNs[m];
        indexOfMin = m;
        for (n = m; n <= CITIZENS - 1; n++) {
            if (SSNs[n] < minimum) {
                minimum = SSNs[n];
                indexOfMin = n;
            }
        }
        temp = SSNs[m];
        SSNs[m] = SSNs[indexOfMin];
        SSNs[indexOfMin] = temp;
        tempStr = answers[m];
        answers[m] = answers[indexOfMin];
        answers[indexOfMin] = tempStr;
    }
}

int searchArray(long long SSNs[], long long SSN) {
    int left, right, middle;
    bool found;

    left = 0;
    right = CITIZENS - 1;
    found = false;
    while (left <= right && !found) {
        middle = (int)((left + right) / 2);

        if (SSN < SSNs[middle]) {
            right = middle - 1;
        }
    }
}
```



```
    }
    else if (SSN > SSNs[middle]) {
        left = middle + 1;
    }
    else {
        found = true;
    }
}

if (!found) {
    cout << "SSN not found!" << endl;
    return -1;
}
else {
    return middle;
}
}

int countAnswers(string answers[], string answer) {
    int count, i;

    count = 0;
    for (i = 0; i <= CITIZENS - 1; i++) {
        if (answers[i] == answer) {
            count++;
        }
    }
    return count;
}

int main() {
    long long SSNs[CITIZENS];
    long long SSN;
    string answers[CITIZENS];
    int index, count;
    string answer;

    do {
        inputData(SSNs, answers);
        sortArrays(SSNs, answers);

        cout << "Enter an SSN to search: ";
        cin >> SSN;

        index = searchArray(SSNs, SSN);
        if (index != -1) {
            answer = answers[index];
            cout << answer << endl;

            count = countAnswers(answers, answer);
            cout << count * 100 / (double)CITIZENS << endl;
        }
        cout << "Repeat? ";
    }
```

```

    cin >> answer;
} while (answer == "yes");
return 0;
}

```

9. Solution

```

#include <iostream>
using namespace std;

const int TEAMS = 8;
const int GAMES = 12;

//Note that in C++, in order to pass multidimensional arrays to functions,
//the arrays in the formal argument list must have bounds for all dimensions except the first
void inputData(string names[], string results[][GAMES]) {
    int i, j;

    for (i = 0; i <= TEAMS - 1; i++) {
        cout << "Enter team name: ";
        cin >> names[i];
        for (j = 0; j <= GAMES - 1; j++) {
            cout << "Enter result (W, L, T): ";
            cin >> results[i][j];
        }
    }
}

void displayResult(string names[], string results[][GAMES]) {
    string result;
    int i, j;
    bool found;

    cout << "Enter a result to search (W, L, T): ";
    cin >> result;
    for (i = 0; i <= TEAMS - 1; i++) {
        cout << "Team: " << names[i] << endl;
        found = false;
        for (j = 0; j <= GAMES - 1; j++) {
            if (results[i][j] == result) {
                cout << "Week: " << j + 1 << endl;
                found = true;
            }
        }
    }
    if (!found) {
        cout << "Nothing found" << endl;
    }
}

int findTeam(string names[]) {
    string name;
    int i, returnValue;

```

```

    cout << "Enter a name to search: ";
    cin >> name;

    i = 0;
    while (i < TEAMS - 1 && names[i] != name) {
        i++;
    }

    if (names[i] != name) {
        returnValue = -1;
    }
    else {
        returnValue = i;
    }
    return returnValue;
}

int main() {
    string names[TEAMS];
    string results[TEAMS][GAMES];
    int j, index, total;

    inputData(names, results);
    displayResult(names, results);

    index = findTeam(names);
    while (index != -1) {
        total = 0;
        for (j = 0; j <= GAMES - 1; j++) {
            if (results[index][j] == "W") {
                total += 3;
            }
            else if (results[index][j] == "T") {
                total += 1;
            }
        }
        cout << "Points: " << total << endl;
        index = findTeam(names);
    }

    if (index == -1) {
        cout << "Team not found" << endl;
    }
    return 0;
}

```

10. Solution

```

#include <iostream>
using namespace std;

bool hasDuplicateDigits(int num) {

```

```
int digit;

//Initialize an array to store the count of each digit
int digitCount[] = {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0};

while (num > 0) {
    digit = num % 10; //Extract the last digit
    if (digitCount[digit] > 0) {
        return true; //If this digit has been seen before, return true
    }
    digitCount[digit]++; //Increment the count of this digit
    x = (int)(x / 10); //Move to the next digit
}

return false; //No duplicate digits found
}

int main() {
    int num;

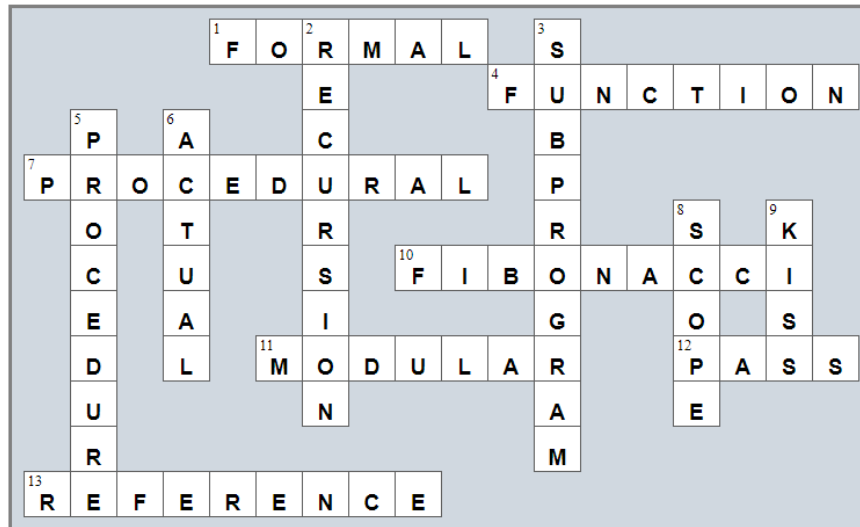
    cout << "Enter an integer: ";
    cin >> num;
    while (num < 11) {
        cout << "Wrong number! Enter an integer greater than 10: ";
        cin >> num;
    }

    if (hasDuplicateDigits(num)) {
        cout << "The integer contains duplicate digits" << endl;
    }
    else {
        cout << "The integer does not contain duplicate digits" << endl;
    }
    return 0;
}
```

Review in "Subprograms"

Review Crossword Puzzle

1.



Chapter 39

39.8 Review Questions: True/False

- | | | |
|----------|-----------|-----------|
| 1. false | 7. false | 13. true |
| 2. true | 8. true | 14. false |
| 3. true | 9. true | 15. true |
| 4. false | 10. false | 16. false |
| 5. false | 11. true | 17. false |
| 6. false | 12. true | |

39.9 Review Exercises

1. Solution

```
#include <iostream>
using namespace std;

class Geometry {
public:
    double rectangleArea(double b, double h) {
        return b * h;
    }

    double triangleArea(double b, double h) {
        return b * h / 2;
    }
};

int main() {
    double sqrSide, rctnglBase, rctnglHeight, trnglBase, trnglHeight;
    Geometry gmtr;

    cout << "Enter square side: ";
    cin >> sqrSide;

    cout << "Enter rectangle base: ";
    cin >> rctnglBase;
    cout << "Enter rectangle height: ";
    cin >> rctnglHeight;

    cout << "Enter triangle base: ";
    cin >> trnglBase;
    cout << "Enter triangle height: ";
    cin >> trnglHeight;

    cout << gmtr.rectangleArea(sqrSide, sqrSide) << endl;
    cout << gmtr.rectangleArea(rctnglBase, rctnglHeight) << endl;
    cout << gmtr.triangleArea(trnglBase, trnglHeight) << endl;
    return 0;
}
```

2. Solution

```
#include <iostream>
using namespace std;

class Pet {
public:
    string kind;
    int legsNumber;

    void startRunning() {
        cout << "Pet is running" << endl;
    }

    void stopRunning() {
        cout << "Pet stopped" << endl;
    }
};

int main() {
    Pet pet1;
    pet1.kind = "dog";
    pet1.legsNumber = 4;

    Pet pet2;
    pet2.kind = "monkey";
    pet2.legsNumber = 2;

    pet1.startRunning();
    pet2.startRunning();
    pet1.stopRunning();
    return 0;
}
```

3. Solution

```
#include <iostream>
using namespace std;

class Pet {
private:
    string _kind;
    int _legsNumber;

public:
    //Define the constructor
    Pet(string kind, int legsNumber) {
        this->setKind(kind);
        this->setLegsNumber(legsNumber);
    }

    //Define the getter
    string getKind() {
```

```
    return this->_kind;
}

//Define the setter
void setKind(string value) {
    if (value != "") {
        this->_kind = value;
    }
    else {
        throw runtime_error("Cannot be empty");
    }
}

//Define the getter
int getLegsNumber() {
    return this->_legsNumber;
}

void setLegsNumber(int value) {
    if (value >= 0) {
        this->_legsNumber = value;
    }
    else {
        throw runtime_error("Cannot be negative");
    }
}

void startRunning() {
    cout << "Pet is running" << endl;
}

void stopRunning() {
    cout << "Pet stopped" << endl;
}
};

int main() {
    Pet pet1("dog", 4);

    pet1.startRunning();
    pet1.stopRunning();

    pet1.setKind(""); //This will throw an error
    pet1.setLegsNumber(-1); //This will throw an error
    return 0;
}
```

4. Solution

```
#include <iostream>
using namespace std;
const int BOXES = 30;
```



```
class Box {
private:
    double _width = 0;
    double _length = 0;
    double _height = 0;
public:
    //Define an empty constructor
    Box() { }

    //Define the constructor
    Box(double w, double l, double h) {
        //Initialize fields
        this->_width = w;
        this->_length = l;
        this->_height = h;
    }

    void displayVolume() {
        cout << "Volume: " << this->_width * this->_length * this->_height << endl;
    }

    void displayDimensions() {
        cout << this->_width << " x " << this->_length << " x " << this->_height << endl;
    }
};

int main() {
    int i;
    double w, l, h;

    Box listOfObj[BOXES]; //Create an array of objects. It calls the first constructor (the empty one).

    for (i = 0; i <= BOXES - 1; i++) {
        cout << "Enter width: ";
        cin >> w;
        cout << "Enter length: ";
        cin >> l;
        cout << "Enter height: ";
        cin >> h;

        //Add each new object to the array
        listOfObj[i] = Box(w, l, h); //It calls the second constructor
    }

    for (i = 0; i <= BOXES - 1; i++) {
        listOfObj[i].displayDimensions();
        listOfObj[i].displayVolume();
    }
    return 0;
}
```

5. Solution

```
#include <iostream>
using namespace std;
const int BOXES = 30;

class Box {
private:
    double _width;
    double _length;
    double _height;

public:
    //Define an empty constructor
    Box() { }

    //Define the constructor
    Box(double w, double l, double h) {
        //Initialize fields
        this->setWidth(w);
        this->setLength(l);
        this->setHeight(h);
    }

    //Define the getter
    double getWidth() {
        return this->_width;
    }

    //Define the setter
    void setWidth(double value) {
        if (value > 0) {
            this->_width = value;
        }
        else {
            throw runtime_error("Cannot be negative or zero");
        }
    }

    //Define the getter
    double getLength() {
        return this->_length;
    }

    //Define the setter
    void setLength(double value) {
        if (value > 0) {
            this->_length = value;
        }
        else {
            throw runtime_error("Cannot be negative or zero");
        }
    }
}
```

```
}

//Define the getter
double getHeight() {
    return this->_height;
}

//Define the setter
void setHeight(double value) {
    if (value > 0) {
        this->_height = value;
    }
    else {
        throw runtime_error("Cannot be negative or zero");
    }
}

void displayVolume() {
    cout << "Volume: " << this->getWidth() * this->getLength() * this->getHeight() << endl;
}

void displayDimensions() {
    cout << this->getWidth() << " x " << this->getLength() << " x " << this->getHeight() << endl;
}
};

int main() {
    int i;
    double w, l, h;

    Box listOfObj[BOXES]; //Create an array of objects. It calls the first constructor (the empty one).

    for (i = 0; i <= BOXES - 1; i++) {
        cout << "Enter width: ";
        cin >> w;
        cout << "Enter length: ";
        cin >> l;
        cout << "Enter height: ";
        cin >> h;

        //Add each new object to the array
        listOfObj[i] = Box(w, l, h); //It calls the second constructor
    }

    for (i = 0; i <= BOXES - 1; i++) {
        listOfObj[i].displayDimensions();
        listOfObj[i].displayVolume();
    }
    return 0;
}
```

6. Solution

```
#include <iostream>
#include <cmath>
using namespace std;

class Cube {
private:
    double _edge;

public:
    //Define the constructor
    Cube(double edge) {
        this->_edge = edge;
    }

    void displayVolume() {
        cout << "Volume: " << pow(this->_edge, 3) << endl;
    }

    void displayOneSurface() {
        cout << "One surface: " << pow(this->_edge, 2) << endl;
    }

    void displayTotalSurface() {
        cout << "Total surface: " << 6 * pow(this->_edge, 2) << endl;
    }
};

int main() {
    double edge;

    cout << "Enter edge length of a cube: ";
    cin >> edge;

    Cube cubel(edge);

    cubel.displayVolume();
    cubel.displayOneSurface();
    cubel.displayTotalSurface();
    return 0;
}
```

7. Solution

```
#include <iostream>
#include <cmath>

using namespace std;
class Cube {
private:
    double _edge;

public:
```

```

//Define the constructor
Cube(double edge) {
    this->setEdge(edge);
}

//Define the getter
double getEdge() {
    return this->_edge;
}

//Define the setter
void setEdge(double value) {
    if (value > 0) {
        this->_edge = value;
    }
    else {
        throw runtime_error("Cannot be negative or zero");
    }
}

void displayVolume() {
    cout << "Volume: " << pow(this->getEdge(), 3) << endl;
}

void displayOneSurface() {
    cout << "One surface: " << pow(this->getEdge(), 2) << endl;
}

void displayTotalSurface() {
    cout << "Total surface: " << 6 * pow(this->getEdge(), 2) << endl;
}
};

int main() {
    double edge;

    cout << "Enter edge length of a cube: ";
    cin >> edge;

    Cube cubel(edge);

    cubel.displayVolume();
    cubel.displayOneSurface();
    cubel.displayTotalSurface();
    return 0;
}

```

8. Solution

```

#include <iostream>
#include <cmath>
using namespace std;

class Circle {

```

```
private:
    double _radius = -1;

public:
    //Define the getter
    double getRadius() {
        if (this->_radius > 0) {
            return this->_radius;
        }
        else {
            throw runtime_error("Radius is not set");
        }
    }

    //Define the setter
    void setRadius(double value) {
        if (value > 0) {
            this->_radius = value;
        }
        else {
            throw runtime_error("Cannot be negative or zero");
        }
    }

    double getDiameter() {
        return 2 * this->getRadius();
    }

    double getArea() {
        return 3.14 * pow(this->getRadius(), 2);
    }

    double getPerimeter() {
        return 2 * 3.14 * this->getRadius();
    }
};

void displayMenu() {
    cout << "1. Enter radius" << endl;
    cout << "2. Display radius" << endl;
    cout << "3. Display diameter" << endl;
    cout << "4. Display area" << endl;
    cout << "5. Display perimeter" << endl;
    cout << "6. Exit" << endl;
}

int main() {
    int choice;
    double radius;

    Circle circle1;

    do {
```

```
    displayMenu();
    cout << "Enter a choice: ";
    cin >> choice;

    if (choice == 1) {
        cout << "Enter radius: ";
        cin >> radius;
        circle1.setRadius(radius);
    }
    else if (choice == 2) {
        cout << "Radius: " << circle1.getRadius() << endl;
    }
    else if (choice == 3) {
        cout << "Diameter: " << circle1.getDiameter() << endl;
    }
    else if (choice == 4) {
        cout << "Area: " << circle1.getArea() << endl;
    }
    else if (choice == 5) {
        cout << "Perimeter: " << circle1.getPerimeter() << endl;
    }
} while (choice != 6);
return 0;
}
```

9. Solution

```
#include <iostream>
#include <boost/algorithm/string.hpp>
using namespace boost::algorithm;
using namespace std;

class Info {
private:
    string _userText;
public:
    //Define the getter
    string getUserText() {
        return this->_userText;
    }

    //Define the setter
    void setUserText(string value) {
        if (value != "") {
            this->_userText = value;
        }
        else {
            throw runtime_error("Cannot be set to empty");
        }
    }
}
```

```
int getSpacesCount() {
    int i, count = 0;
    string character;
    string text = this->getUserText();

    for (i = 0; i <= text.length() - 1; i++) {
        character = text[i];
        if (character == " ") {
            count += 1;
        }
    }
    return count;
}

int getWordsCount() {
    return this->getSpacesCount() + 1;
}

int getVowelsCount() {
    int i, count = 0;
    char character;
    string text = this->getUserText();
    string vowels = "aeiou";

    for (i = 0; i <= text.length() - 1; i++) {
        character = to_lower_copy(text)[i];
        if (vowels.find(character) != -1) {
            count += 1;
        }
    }
    return count;
}

int getLettersCount() {
    string text = this->getUserText();

    return text.length() - this->getSpacesCount();
}
};

int main() {
    string text;

    Info inf;

    cout << "Enter a text: ";
    getline(cin, text);
    inf.setUserText(text);

    cout << "Text: " << inf.getUserText() << endl;
    cout << "Spaces: " << inf.getSpacesCount() << endl;
    cout << "Words: " << inf.getWordsCount() << endl;
    cout << "Vowels: " << inf.getVowelsCount() << endl;
```



```

    cout << "Total number of letters: " << inf.getLettersCount() << endl;
    return 0;
}

```

10. Solution

```

#include <iostream>
using namespace std;

class EncryptDecrypt {
    const string alphabet = " abcdefghijklmnopqrstuvwxyz"; //space is a valid character!
private:
    int _encrDecrKey = -1;
public:
    //Define the getter
    int getEncrDecrKey() {
        if (this->_encrDecrKey != -1) {
            return this->_encrDecrKey;
        }
        else {
            throw runtime_error("Key is not set");
        }
    }

    //Define the setter
    void setEncrDecrKey(int value) {
        if (value >= 1 && value <= 26) {
            this->_encrDecrKey = value;
        }
        else {
            throw runtime_error("Must be between 1 and 26");
        }
    }

    string encrypt(string message) {
        string returnValue = "";
        char character, newLetter;
        int i, index, newIndex;

        for (i = 0; i <= message.length() - 1; i++) {
            character = message[i];
            index = alphabet.find(character);
            newIndex = index + this->getEncrDecrKey();
            if (newIndex >= 27) {
                newIndex -= 27;
            }
            newLetter = alphabet[newIndex];
            returnValue += newLetter;
        }
        return returnValue;
    }
}

```

```
string decrypt(string encMessage) {
    string returnValue = "";
    char character, newLetter;
    int i, index, newIndex;

    for (i = 0; i <= encMessage.length() - 1; i++) {
        character = encMessage[i];
        index = alphabet.find(character);
        newIndex = index - this->getEncrDecrKey();
        if (newIndex < 0) {
            newIndex += 27;
        }
        newLetter = alphabet[newIndex];
        returnValue += newLetter;
    }
    return returnValue;
};

void displayMenu() {
    cout << "1. Enter encryption/decryption key" << endl;
    cout << "2. Encrypt a message" << endl;
    cout << "3. Decrypt a message" << endl;
    cout << "4. Exit" << endl;
}

int main() {
    string text;
    int choice, encrDecrKey;

    EncryptDecrypt ed;

    displayMenu();
    cout << "Enter a choice: ";
    cin >> choice;

    while (choice != 4) {
        //When there is a cin statement before a getline() function
        //you need to write the following statement between them
        //otherwise the next getline() function won't work
        cin.ignore(1, '\n');

        if (choice == 1) {
            cout << "Enter encryption/decryption key: ";
            cin >> encrDecrKey;
            ed.setEncrDecrKey(encrDecrKey);
        }
        else if (choice == 2) {
            cout << "Enter message to encrypt: ";
            getline(cin, text);
            cout << "Encrypted message: " << ed.encrypt(text) << endl;
        }
    }
}
```

```

    else if (choice == 3) {
        cout << "Enter message to decrypt: ";
        getline(cin, text);
        cout << "Decrypted message: " << ed.decrypt(text) << endl;
    }

    displayMenu();
    cout << "Enter a choice: ";
    cin >> choice;
}
return 0;
}

```

11. Solution

```

#include <iostream>
using namespace std;

class Vehicle {
public:
    int numberOfWheels;
    string color;
    double length, width, height;

    //Define the constructor
    Vehicle(int numberOfWheels, string color, double length, double width, double height) {
        this->numberOfWheels = numberOfWheels;
        this->color = color;
        this->length = length;
        this->width = width;
        this->height = height;
    }

    void startEngine() {
        cout << "The engine started" << endl;
    }

    void stopEngine() {
        cout << "The engine stopped" << endl;
    }
};

class Car : public Vehicle {
public:
    int bootCapacity;

    //Define the constructor
    Car(int numberOfWheels, string color, double length, double width, double height)
        : Vehicle(numberOfWheels, color, length, width, height) {
        this->bootCapacity = 0;
    }

    void turnWindshieldWipersOn() {

```

```

        cout << "The windshield wipers have been turned on!" << endl;
    }
};

class Motorcycle : public Vehicle {
public:
    bool hasLuggage;

    //Define the constructor
    Motorcycle(int numberOfWheels, string color, double length, double width, double height)
        : Vehicle(numberOfWheels, color, length, width, height) {
        this->hasLuggage = false;
    }

    void doAWheelie() {
        cout << "I am doing a wheelie!!!" << endl;
    }
};

int main() {
    Car car1(4, "Red", 5, 2, 1.5);
    car1.bootCapacity = 300;
    car1.startEngine();
    car1.turnWindshieldWipersOn();
    car1.stopEngine();

    Car car2(4, "Green", 4.5, 2.2, 1.4);
    car2.bootCapacity = 400;
    car2.startEngine();
    car2.turnWindshieldWipersOn();
    car2.stopEngine();

    Motorcycle motorcycle1(2, "Blue", 2, 0.9, 1.3);
    motorcycle1.hasLuggage = true;
    motorcycle1.startEngine();
    motorcycle1.doAWheelie();
    motorcycle1.stopEngine();
    return 0;
}

```

12. Solution

```

#include <iostream>
using namespace std;

class SchoolMember {
private:
    string _name;
    int _age;
public:
    SchoolMember(string name, int age) {
        this->setName(name);
    }
};

```

```
        this->setAge(age);
        cout << "A school member was initialized" << endl;
    }

    string getName() {
        return this->_name;
    }

    void setName(string value) {
        if (value != "")
            this->_name = value;
        else
            throw runtime_error("Name cannot be empty");
    }

    int getAge() {
        return this->_age;
    }

    void setAge(int value) {
        if (value > 0)
            this->_age = value;
        else
            throw runtime_error("Age cannot be negative or zero");
    }
};

class Teacher : public SchoolMember {
private:
    double _salary;
public:
    Teacher(string name, int age, double salary) : SchoolMember(name, age) {
        this->setSalary(salary);
        cout << "A teacher was initialized" << endl;
    }

    void displayValues() {
        cout << "Name: " << this->getName() << endl;
        cout << "Age: " << this->getAge() << endl;
        cout << "Salary: " << this->getSalary() << endl;
    }

    double getSalary() {
        return this->_salary;
    }

    void setSalary(double value) {
        if (value >= 0)
            this->_salary = value;
        else
            throw runtime_error("Salary cannot be negative");
    }
};
```

```
};

class Student : public SchoolMember {
private:
    string _finalGrade;
public:
    Student(string name, int age, string finalGrade) : SchoolMember(name, age) {
        this->setFinalGrade(finalGrade);
        cout << "A student was initialized" << endl;
    }

    void displayValues() {
        cout << "Name: " << this->getName() << endl;
        cout << "Age: " << this->getAge() << endl;
        cout << "Final grade: " << this->getFinalGrade() << endl;
    }

    string getFinalGrade() {
        return this->_finalGrade;
    }

    void setFinalGrade(string value) {
        if (value == "A" || value == "B" || value == "C" || value == "D" || value == "E" || value == "F")
            this->_finalGrade = value;
        else
            throw runtime_error("Final grade must be in the range of 'A' to 'F'");
    }
};

int main() {
    Teacher teacher1("Mr. John Scott", 43, 35000);
    Teacher teacher2("Mrs. Ann Carter", 5, 32000);

    Student student1("Mark Nelson", 14, "A");
    Student student2("Mary Morgan", 13, "B");

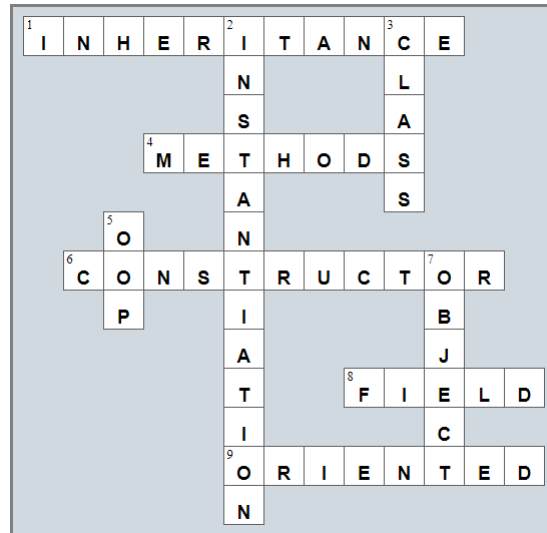
    teacher1.displayValues();
    teacher2.displayValues();

    student1.displayValues();
    student2.displayValues();
    return 0;
}
```

Review in "Object Oriented Programming"

Review Crossword Puzzle

1.



Chapter 40

40.8 Review Questions: True/False

- | | | |
|----------|-----------|-----------|
| 1. false | 9. true | 17. false |
| 2. false | 10. false | 18. false |
| 3. true | 11. true | 19. false |
| 4. false | 12. false | 20. true |
| 5. false | 13. false | 21. true |
| 6. false | 14. false | 22. true |
| 7. false | 15. true | |
| 8. false | 16. true | |

40.9 Review Exercises

1. Solution

```
#include <iostream>
#include <fstream>
using namespace std;

const string PATH = "c:/temp/";

int main() {
    string days[] = {"Sunday", "Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday"};

    fstream f(PATH + "days_of_week.txt", fstream::out);
    for (const auto& d : days) {
        f << d << endl;
    }
    f.close();
    return 0;
}
```

2. Solution

```
#include <iostream>
#include <fstream>
using namespace std;

const string PATH = "c:/temp/";

int main() {
    int i;
    string days[7];

    fstream f(PATH + "days_of_week.txt", fstream::in);
    for (i = 0; i <= 6; i++) {
        getline(f, days[i]);
    }
}
```



```
f.close();

for (i = 6; i >= 0; i--) {
    cout << days[i] << endl;
}
return 0;
}
```

3. Solution

```
#include <iostream>
#include <fstream>
using namespace std;

const string PATH = "c:/temp/";

int main() {
    fstream f(PATH + "days_of_week.txt", fstream::app);
    f << "*** End of File ***" << endl;
    f.close();

    return 0;
}
```

4. Solution

```
#include <iostream>
#include <fstream>
#include <ctime>
#include <cstdlib>
using namespace std;

const string PATH = "c:/temp/";

int main() {
    int i;

    srand(time(NULL));

    fstream f(PATH + "randoms.txt", fstream::out);
    for (i = 0; i <= 49; i++) {
        f << 1 + rand() % 100 << endl;
    }
    f.close();
    return 0;
}
```

5. Solution

```
#include <iostream>
#include <fstream>
#include <ctime>
#include <cstdlib>
using namespace std;
```

```

const string PATH = "c:/temp/";

int main() {
    int i;

    srand(time(NULL));

    for (i = 1; i <= 10; i++) {
        fstream f(PATH + "file" + to_string(i) + ".txt", fstream::out);
        f << 100 + rand() % (10000 - 100 + 1) << endl;
        f.close();
    }

    return 0;
}

```

6. Solution

```

#include <iostream>
#include <fstream>
using namespace std;

const string PATH = "c:/temp/";

int main() {
    int i, j;

    fstream f(PATH + "multiplication_table.txt", fstream::out);

    for (i = 1; i <= 10; i++) {
        for (j = 1; j <= 4; j++) {
            f << i << " x " << j << " = " << i * j << endl;
        }
    }

    f.close();
    return 0;
}

```

7. Solution

```

#include <iostream>
#include <fstream>
using namespace std;

const string PATH = "c:/temp/";

int main() {
    string line;
    fstream f(PATH + "a_file.txt", fstream::in);

    while (!f.eof()) {
        getline(f, line);
        cout << line.length() << endl;
    }
}

```

```
    }  
    f.close();  
  
    return 0;  
}
```

8. Solution

First approach

```
#include <iostream>  
#include <fstream>  
using namespace std;  
  
const string PATH = "c:/temp/";  
const string PUNCTUATION_CHARS = ",.!";  
  
int main() {  
    int i;  
    string line;  
    fstream f(PATH + "a_file.txt", fstream::in);  
  
    i = 1;  
    while (!f.eof()) {  
        getline(f, line);  
        for (const auto& character : line) {  
            if ((int)PUNCTUATION_CHARS.find(character) > -1) {  
                cout << "There is a punctuation mark on line No " << i << endl;  
                break;  
            }  
        }  
        i++;  
    }  
    f.close();  
  
    return 0;  
}
```

Second approach

```
#include <iostream>  
#include <fstream>  
using namespace std;  
  
const string PATH = "c:/temp/";  
  
int main() {  
    int i;  
    string line;  
    fstream f(PATH + "a_file.txt", fstream::in);  
  
    i = 1;  
    while (!f.eof()) {  
        getline(f, line);  
        if ((int)line.find(",") > -1 || (int)line.find(".") > -1 || (int)line.find("!") > -1) {  
            cout << "There is a punctuation mark on line No " << i << endl;  
        }  
    }  
}
```

```
    }  
    i++;  
  }  
  f.close();  
  return 0;  
}
```

Chapter 41

41.2 Review Exercises

1. Solution

```
#include <iostream>
#include <fstream>
using namespace std;

const string PATH = "c:/temp/";

int main() {
    string values;
    int i, total, count, number;

    fstream fin(PATH + "f_data41.2-1.txt", fstream::in);
    getline(fin, values);
    fin.close();

    total = 0;
    count = 0;
    for (i = 0; i < 10; i++) {
        number = stoi(values.substr(i * 3, 2));
        if (number > 50) {
            total += number;
            count += 1;
        }
    }

    if (count > 0) {
        cout << total / (double)count << endl;
    }
    return 0;
}
```

2. Solution

```
#include <iostream>
#include <fstream>
using namespace std;

const string PATH = "c:/temp/";

int main() {
    string values;
    int i, total, count, number;

    fstream fin(PATH + "f_data41.2-2.txt", fstream::in);
    getline(fin, values);
    fin.close();

    total = 0;
```

```

count = 0;
i = 0;
while (i < values.length() / 4) {
    number = stoi(values.substr(i * 4, 3));
    if (number >= 300 && number <= 500) {
        total += number;
        count += 1;
    }
    i++;
}

if (count > 0) {
    cout << total / (double)count << endl;
}
return 0;
}

```

3. Solution

```

#include <iostream>
#include <fstream>
using namespace std;

const string PATH = "c:/temp/";

int main() {
    string maxName, minName, name, line;
    int maximum, minimum, grade, commaPosition;

    fstream fin(PATH + "f_data41.2-3.txt", fstream::in);

    //Read the first line
    getline(fin, line);

    commaPosition = line.find(',');
    grade = stoi(line.substr(0, commaPosition));
    name = line.substr(commaPosition + 1);

    maximum = minimum = grade;
    maxName = minName = name;

    //Read the rest of the lines
    while (!fin.eof()) {
        getline(fin, line);

        commaPosition = line.find(',');
        grade = stoi(line.substr(0, commaPosition));
        name = line.substr(commaPosition + 1);

        if (grade > maximum) {
            maximum = grade;
            maxName = name;
        }

        if (grade < minimum) {

```

```

        minimum = grade;
        minName = name;
    }
}

fin.close();

cout << maxName << endl;
cout << minName << endl;
return 0;
}

```

4. Solution

```

#include <iostream>
#include <fstream>
#include <boost/algorithm/string.hpp>
using namespace boost::algorithm;
using namespace std;

const string PATH = "c:/temp/";

int main() {
    string description, maximumDescription, keyword, stringInfo1, stringInfo2;
    double width, length, height, volume, total;
    double maximum;

    cout << "Enter a keyword to search: ";
    cin >> keyword;
    keyword = to_lower_copy(keyword);

    fstream fin(PATH + "f_data41.2-4.txt");

    maximum = total = 0;
    stringInfo1 = stringInfo2 = "";
    while (!fin.eof()) {
        fin >> width;
        fin >> length;
        fin >> height;
        getline(fin, description);
        description = trim_copy(description);

        if ((int)to_lower_copy(description).find(keyword) != -1) {
            stringInfo1 += description + " - Dimensions: " +
                to_string(width) + " x " + to_string(length) + " x " + to_string(height) + "\n";
        }

        volume = width * length * height / 1728;
        stringInfo2 += description + ": Volume = " + to_string(volume) + " cubic feet\n";

        total += volume;

        if (volume > maximum) {
            maximum = volume;
        }
    }
}

```

```

        maximumDescription = description;
    }
}

fin.close();

if (stringInfo1 != "") {
    cout << "Keyword '" << keyword << "' found!" << endl;
    cout << stringInfo1 << endl;
}

cout << "Volume of each item:" << endl;
cout << stringInfo2 << endl;
cout << "Total volume: " << total << endl;
cout << "Greatest box: " << maximumDescription << endl;
return 0;
}

```

5. Solution

First approach

```

#include <iostream>
#include <fstream>
using namespace std;

int main() {
    string filename1, filename2, s, contents;

    cout << "Enter filename No 1: ";
    cin >> filename1;

    if (filename1.substr(filename1.length() - 4) != ".txt") {
        cout << "Wrong filename" << endl;
    }
    else {
        cout << "Enter filename No 2: ";
        cin >> filename2;
        if (filename2.substr(filename2.length() - 4) != ".txt") {
            cout << "Wrong filename" << endl;
        }
        else {
            fstream fin;

            contents = "";
            fin.open(filename2, fstream::in);
            while (!fin.eof()) {
                getline(fin, s);
                contents += s + "\n";
            }
            fin.close();

            fin.open(filename1, fstream::in);
            while (!fin.eof()) {

```



```

        getline(fin, s);
        contents += s + "\n";
    }
    fin.close();

    fstream fout("final.txt", fstream::out);
    fout << contents;
    fout.close();
}
}
return 0;
}

```

Second approach

```

#include <iostream>
#include <fstream>
using namespace std;

int main() {
    string filename1, filename2, s, contents;

    cout << "Enter filename No 1: ";
    cin >> filename1;

    if (filename1.substr(filename1.length() - 4) != ".txt") {
        cout << "Wrong filename" << endl;
    }
    else {
        cout << "Enter filename No 2: ";
        cin >> filename2;
        if (filename2.substr(filename1.length() - 4) != ".txt") {
            cout << "Wrong filename" << endl;
        }
        else {
            fstream fin1(filename2, fstream::in);
            fstream fin2(filename1, fstream::in);
            fstream fout("final.txt", fstream::out);

            contents = "";
            while (!fin1.eof()) {
                getline(fin1, s);
                contents += s + "\n";
            }

            while (!fin2.eof()) {
                getline(fin2, s);
                contents += s + "\n";
            }

            fout << contents;

            fin1.close();
            fin2.close();

```

```

        fout.close();
    }
}
return 0;
}

```

6. Solution

```

#include <iostream>
#include <fstream>
using namespace std;

const string PATH = "c:/temp/";
const int ELEMENTS = 15;

int main() {
    int i, m, n;
    double numbers[ELEMENTS];
    double temp;
    string line;

    fstream fin(PATH + "f_data41.2-6.txt", fstream::in);
    for (i = 0; i < ELEMENTS; i++) {
        getline(fin, line);
        numbers[i] = stod(line);
    }
    fin.close();

    for (m = 1; m <= ELEMENTS - 1; m++) {
        for (n = ELEMENTS - 1; n >= m; n--) {
            if (numbers[n] < numbers[n - 1]) {
                temp = numbers[n];
                numbers[n] = numbers[n - 1];
                numbers[n - 1] = temp;
            }
        }
    }

    fstream fout(PATH + "f_data41.2-6.txt", fstream::app);
    fout << "\n***** Sorted numbers *****";
    for (const auto& number : numbers) {
        fout << number << endl;
    }
    fout.close();
    return 0;
}

```

7. Solution

```

#include <iostream>
#include <fstream>
#include <algorithm>
using namespace std;

```

```

const string PATH = "c:/temp/";
const int NUMBER_OF_CITIES = 8;

int main() {
    int i;
    bool onCityLine;
    double total, average, maximum;
    string line;

    string cities[NUMBER_OF_CITIES];
    double temperatures[NUMBER_OF_CITIES];

    fstream fin(PATH + "f_data41.2-7.txt", fstream::in);

    //Split read values into two arrays (cities and temperatures)
    i = 0;
    onCityLine = true;
    while (!fin.eof()) {
        getline(fin, line);

        if (onCityLine) {
            cities[i] = line;
        }
        else {
            temperatures[i++] = stod(line);
        }
        onCityLine = !onCityLine; //true becomes false, and false becomes true
    }

    fin.close();

    total = 0;
    for (i = 0; i < NUMBER_OF_CITIES; i++) {
        total += temperatures[i];
    }

    average = total / NUMBER_OF_CITIES;
    cout << average << endl;

    maximum = *max_element(temperatures, end(temperatures));
    cout << "Highest temperature: " << maximum << endl;
    for (i = 0; i < NUMBER_OF_CITIES; i++) {
        if (temperatures[i] == maximum) {
            cout << cities[i] << endl;
        }
    }
    return 0;
}

```

8. Solution

```

#include <iostream>
#include <fstream>

```

```

using namespace std;

const string PATH = "c:/temp/";

string abbreviate(string word) {
    if (word.length() > 10) {
        return word[0] + to_string(word.length() - 2) + word[word.length() - 1];
    }
    else {
        return word;
    }
}

int main() {
    string line, word;
    int spaceIndex;
    fstream fin(PATH + "f_data41.2-8.txt", fstream::in);

    while (!fin.eof()) {
        getline(fin, line);

        spaceIndex = (int)line.find(" "); //Find the first space
        while (spaceIndex != -1) {
            word = line.substr(0, spaceIndex); //Get the word and
            line = line.substr(spaceIndex + 1); //remove it from line

            cout << abbreviate(word) << " ";

            spaceIndex = (int)line.find(" ");
        }
        //Display the last word remained in the string line
        cout << abbreviate(line) << endl;
    }

    fin.close();
    return 0;
}

```

9. Solution

```

#include <iostream>
#include <fstream>
using namespace std;

const string PATH = "c:/temp/";

string pigLatinTranslator(string word) {
    int i, firstVowelIndex;
    string pigLatinWord, vowels = "aeiou";

    if ((int)vowels.find(word[0]) != -1) { //If first character is vowel
        pigLatinWord = word + "way";
    }
    else {

```

```

    //Find the index of the first vowel
    firstVowelIndex = -1;
    for (i = 0; i <= word.length() - 1; i++) {
        if ((int)vowels.find(word[i]) != -1) {
            firstVowelIndex = i;
            break;
        }
    }

    //If at least one vowel found
    if (firstVowelIndex != -1) {
        //Move the consonants to the end
        word = word.substr(firstVowelIndex) + word.substr(0, firstVowelIndex);
    }
    pigLatinWord = word + "ay";
}
return pigLatinWord;
}

int main() {
    string line, word;
    int spaceIndex;
    ifstream fin(PATH + "f_data41.2-9.txt", ifstream::in);
    ofstream fout(PATH + "pig_latin_translation.txt", ofstream::out);

    while (!fin.eof()) {
        getline(fin, line);

        spaceIndex = (int)line.find(" "); //Find the first space
        while (spaceIndex != -1) {
            word = line.substr(0, spaceIndex); //Get the word and
            line = line.substr(spaceIndex + 1); //remove it from line

            fout << pigLatinTranslator(word) << " ";

            spaceIndex = (int)line.find(" ");
        }
        //Write the last word remained in the string line
        fout << pigLatinTranslator(line) << endl;
    }

    fin.close();
    fout.close();
}

```

10. Solution

```

#include <iostream>
#include <fstream>
#include <boost/algorithm/string.hpp>
using namespace boost::algorithm;
using namespace std;

```

```

int main() {
    const string PATH = "c:/temp/";

    const string X = " ABCDEFGHIJKLMNOPQRSTUVWXYZ"; //The space character remains as is
    const string Y = " JKWCTAMEDXSLFBYUNGRZOIQVHP";

    string initialMessage, encryptedMessage;
    int i;

    cout << "Enter a message to encrypt: ";
    getline(cin, initialMessage);
    initialMessage = to_upper_copy(initialMessage);

    encryptedMessage = "";
    for (const auto& letter : initialMessage) {
        //Search for letter in const X
        for (i = 0; i < 27; i++) {
            if (letter == X[i]) {
                //Create encrypted message using letters from const Y
                encryptedMessage += Y[i];
                break;
            }
        }
    }

    ofstream fout(PATH + "encrypted.txt", ofstream::out);
    fout << encryptedMessage;
    fout.close();
    return 0;
}

```

11. Solution

```

#include <iostream>
#include <fstream>
using namespace std;

int main() {
    const string PATH = "c:/temp/";

    const string X = " ABCDEFGHIJKLMNOPQRSTUVWXYZ"; //The space character remains as is
    const string Y = " JKWCTAMEDXSLFBYUNGRZOIQVHP";

    string initialMessage, encryptedMessage;
    int i;

    ifstream fin(PATH + "encrypted.txt", ifstream::in);
    getline(fin, encryptedMessage);
    fin.close();

    initialMessage = "";
    for (const auto& letter : encryptedMessage) {
        //Search for letter in const Y
        for (i = 0; i < 27; i++) {

```

```

    if (letter == Y[i]) {
        //Create decrypted message using letters from const X
        initialMessage += X[i];
        break;
    }
}

fstream fout(PATH + "decrypted.txt", fstream::out);
fout << initialMessage;
fout.close();
return 0;
}

```

12. Solution

First approach

```

void copyFile(string source, string destination) {
    string s, x;

    fstream fin(source, fstream::in);
    x = "";
    while (!fin.eof()) {
        getline(fin, s);
        x += s + "\n";
    }
    fin.close();

    fstream fout(destination, fstream::out);
    fout << x;
    fout.close();
}

```

Second approach

```

void copyFile(string source, string destination) {
    string s;

    fstream fin(source, fstream::in);
    fstream fout(destination, fstream::out);
    while (!fin.eof()) {
        getline(fin, s);
        fout << s << endl;
    }
    fin.close();
    fout.close();
}

```

13. Solution

```

#include <iostream>
#include <fstream>
#include <cmath>
using namespace std;

```

```
const string PATH = "c:/temp/";

class Triangle {
private:
    double _sideA, _sideB, _sideC;

public:
    //Define the constructor
    Triangle() {
        string line;

        ifstream fin(PATH + "f_data41.2-13.txt", ifstream::in);
        getline(fin, line);
        this->_sideA = stod(line);
        getline(fin, line);
        this->_sideB = stod(line);
        getline(fin, line);
        this->_sideC = stod(line);
        fin.close();
    }

    bool canBeTriangle() {
        if (this->_sideA > 0 && this->_sideB > 0 && this->_sideC > 0 &&
            this->_sideA + this->_sideB > this->_sideC &&
            this->_sideB + this->_sideC > this->_sideA &&
            this->_sideC + this->_sideA > this->_sideB) {
            return true;
        }
        else {
            return false;
        }
    }

    void displayLengths() {
        cout << "Side A: " << this->_sideA << endl;
        cout << "Side B: " << this->_sideB << endl;
        cout << "Side C: " << this->_sideC << endl;
        if (this->canBeTriangle()) {
            cout << "Can be lengths of the three sides of a triangle!" << endl;
        }
        else {
            cout << "Cannot be lengths of the three sides of a triangle!" << endl;
        }
    }

    void displayArea() {
        double s, area;

        if (this->canBeTriangle()) {
            s = (this->_sideA + this->_sideB + this->_sideC) / 2;
            area = sqrt(s * (s - this->_sideA) * (s - this->_sideB) * (s - this->_sideC));
            cout << "Area: " << area << endl;
        }
    }
};
```

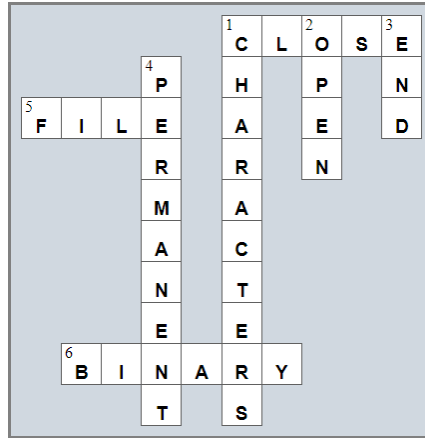


```
    }  
}  
  
void displayPerimeter() {  
    double perimeter;  
  
    if (this->canBeTriangle()) {  
        perimeter = this->_sideA + this->_sideB + this->_sideC;  
        cout << "Perimeter: " << perimeter << endl;  
    }  
}  
};  
  
int main() {  
    Triangle tr;  
  
    tr.displayLengths();  
    tr.displayArea();  
    tr.displayPerimeter();  
    return 0;  
}
```

Review in "Files"

Review Crossword Puzzle

1.



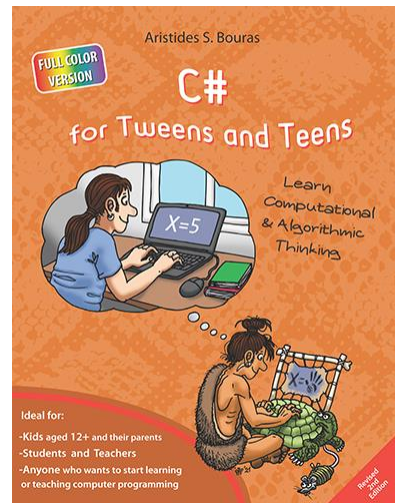
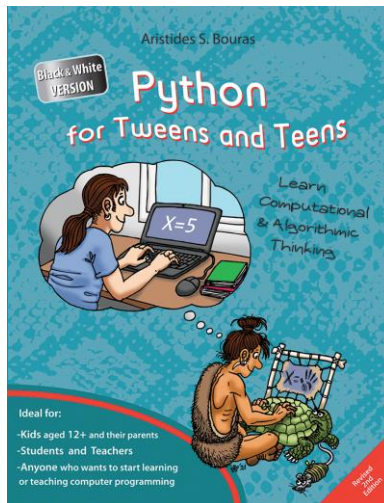
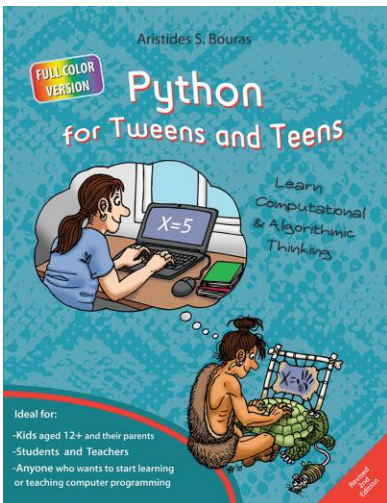
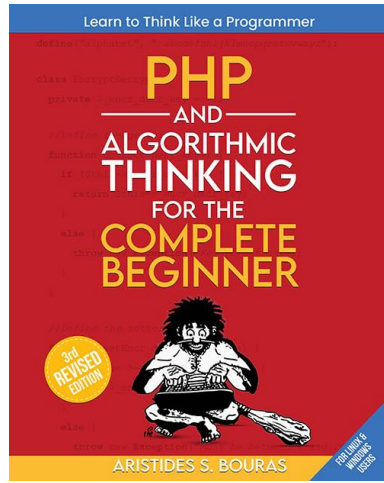
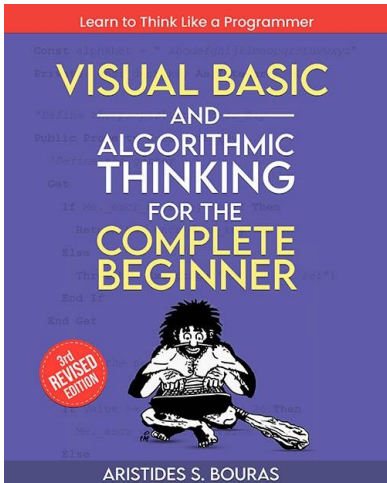
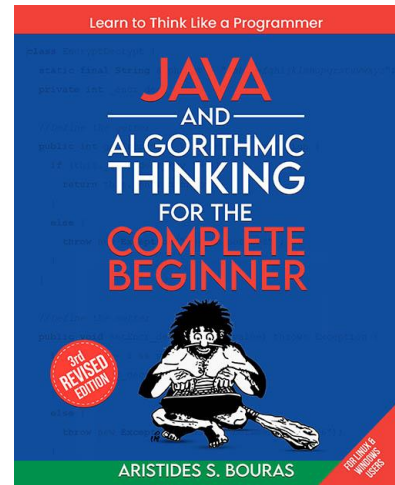
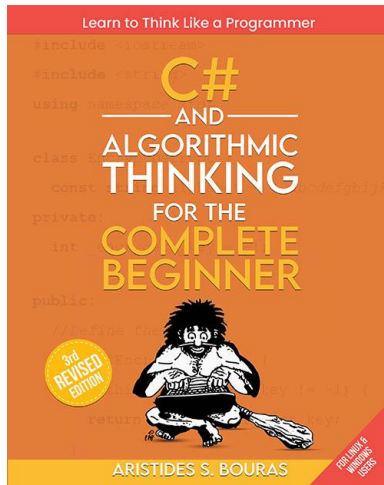
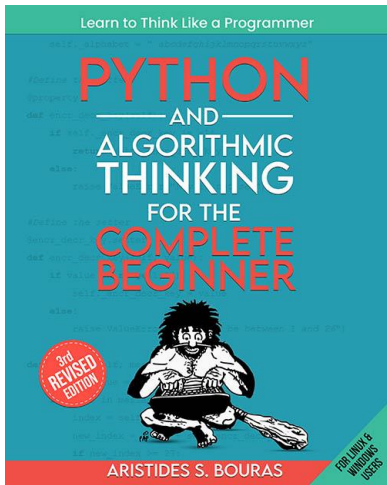
Some Final Words from the Author

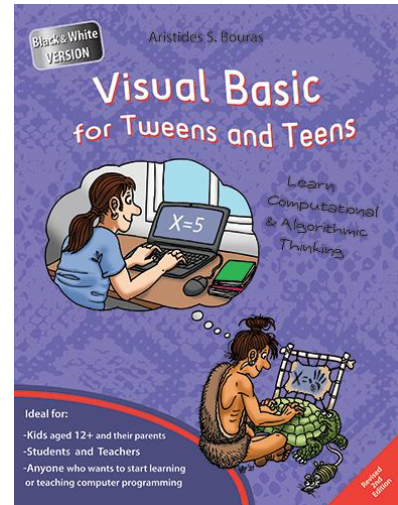
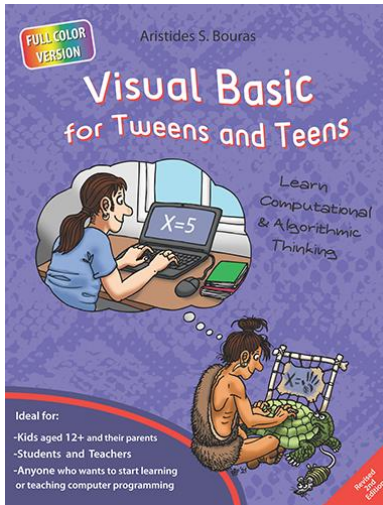
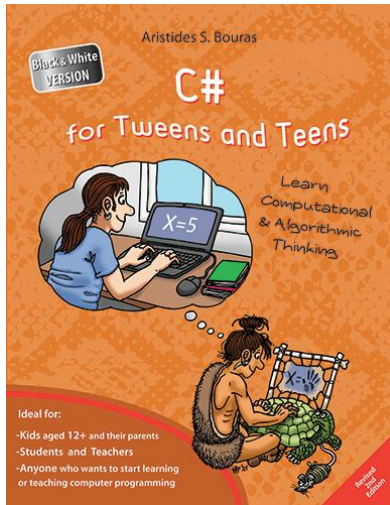
I hope you thoroughly enjoyed reading this book. I made every possible effort to ensure it is beneficial and comprehensible, even for people who may have no prior experience in programming.

If you found this book valuable, please consider visiting the web store where you purchased it, as well as [goodreads.com](https://www.goodreads.com), to show your appreciation by writing a positive review and awarding as many stars as you think appropriate. By doing so, you will motivate me to keep writing and, of course, you'll be assisting other readers in discovering my work.

And always remember: Learning is a lifelong, continuous process that begins at birth and extends throughout your lifetime!

Some of my Books





For more information about my books visit my website:

<https://www.bouraspage.com>

