Chapter 5 Smith Exercises & Labs

Exercise 5-1.

1. x = 6;

y = ++x;

After this code executes, what is the value of **x** and **y**?

x = 7

y = 7

1. x = 6;

y = x++;

After this code executes, what is the value of **x** and **y**?

x = 7

y = 6

1. x = 6;

y = --x;

After this code executes, what is the value of **x** and **y**?

x = 5

y = 5

1. x = 6;

y = x--;

After this code executes, what is the value of **x** and **y**?

x = 5

y = 6

Exercise 5-2.

1. What is the loop control variable? What is the sentinel value?

Variable is numTimes, sentinel value is 8

1. What is the output?

There is no output because numTimes can never be less than NUM\_LOOPS

1. What is the output if the code is changed to **while(numTimes++ <= NUM\_LOOPS)**?

Value of numTimes is 9

1. What is the output if the code is changed to **while(++numTimes <= NUM\_LOOPS)**?

There is no output because numTimes can never be less than or equal to NUM\_LOOPS

Exercise 5-3.

1. What is the value of **number1** when the loop exits?

6

1. What is the value of **number2** when the loop exits?

21

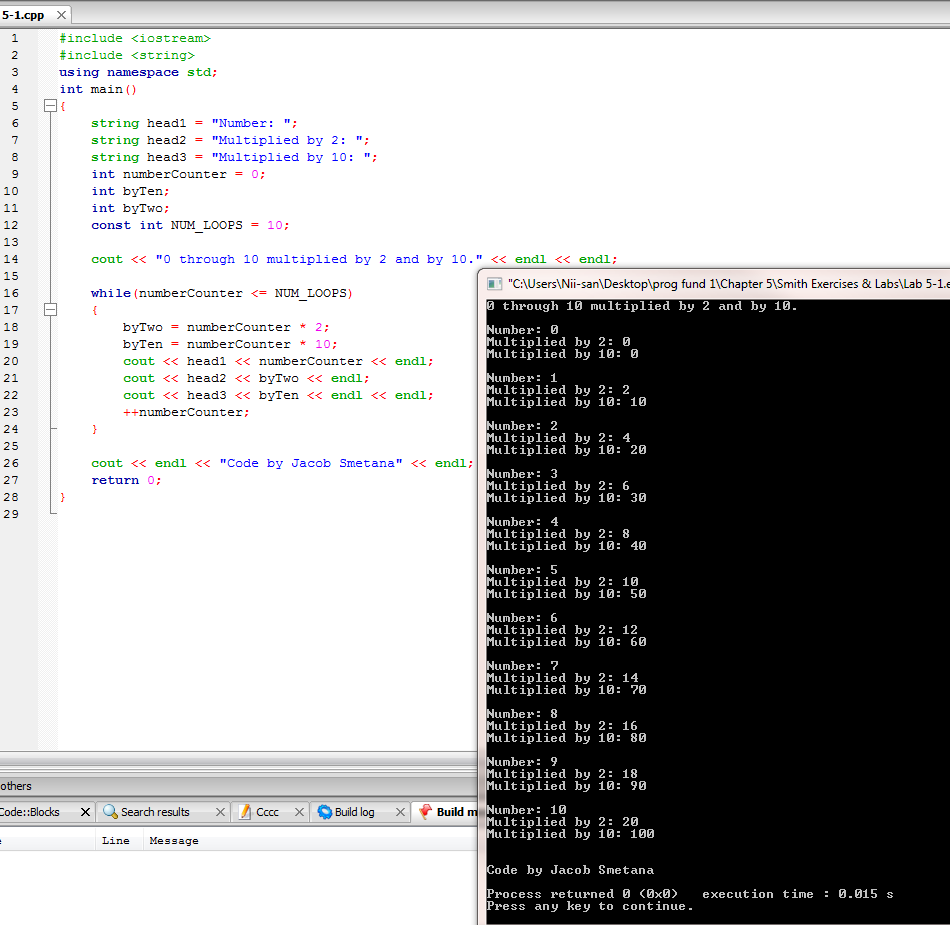
1. If the statement **number1++** is changed to **++number1**, what is the value of **number1** when the loop exits?

6

1. What could you do to force the value of **number2** to be **21** when the loop exits?

Make number2 = 21

Lab 5-1.



Exercise 5-4.

1. What is the output if the user enters a **3**?

Page Number 1

Page Number 2

Page Number 3

Value of counter is 4

1. What is the problem with this code, and how can you fix it?

The counter variable ends as one more than the total number of pages to print.

Add **counter--** after the while loop

1. Assuming you fix the problem, if the user enters **50** as the number of pages to print, what is the value of **counter** when the loop exits?

50

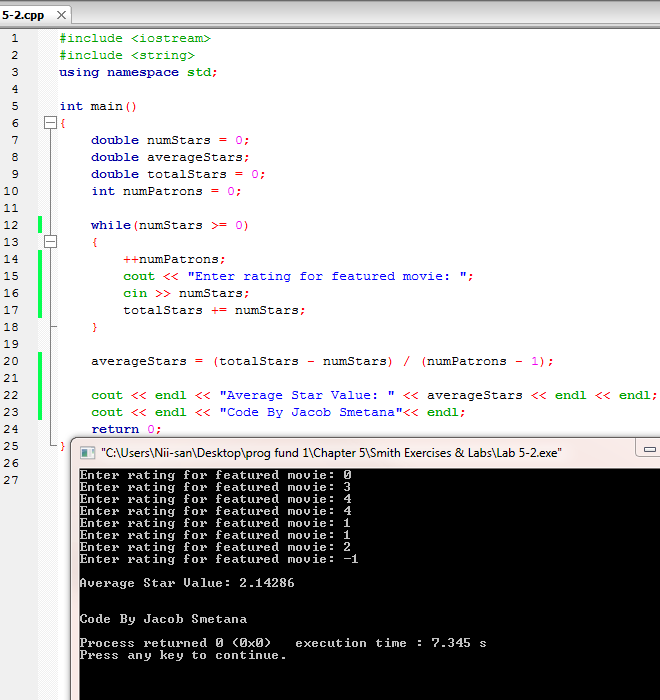
1. Assuming you fix the problem, if the user enters **0** as the number of pages to print, how many pages will print?

0

1. What is the output if the curly braces are deleted?

Deleting the curly braces results in an infinite loop unless 0 is entered instead.

Lab 5-2.



Exercise 5-5.

1. The loop executes 12 times.

False

1. This loop could be written as a **while** loop.

True

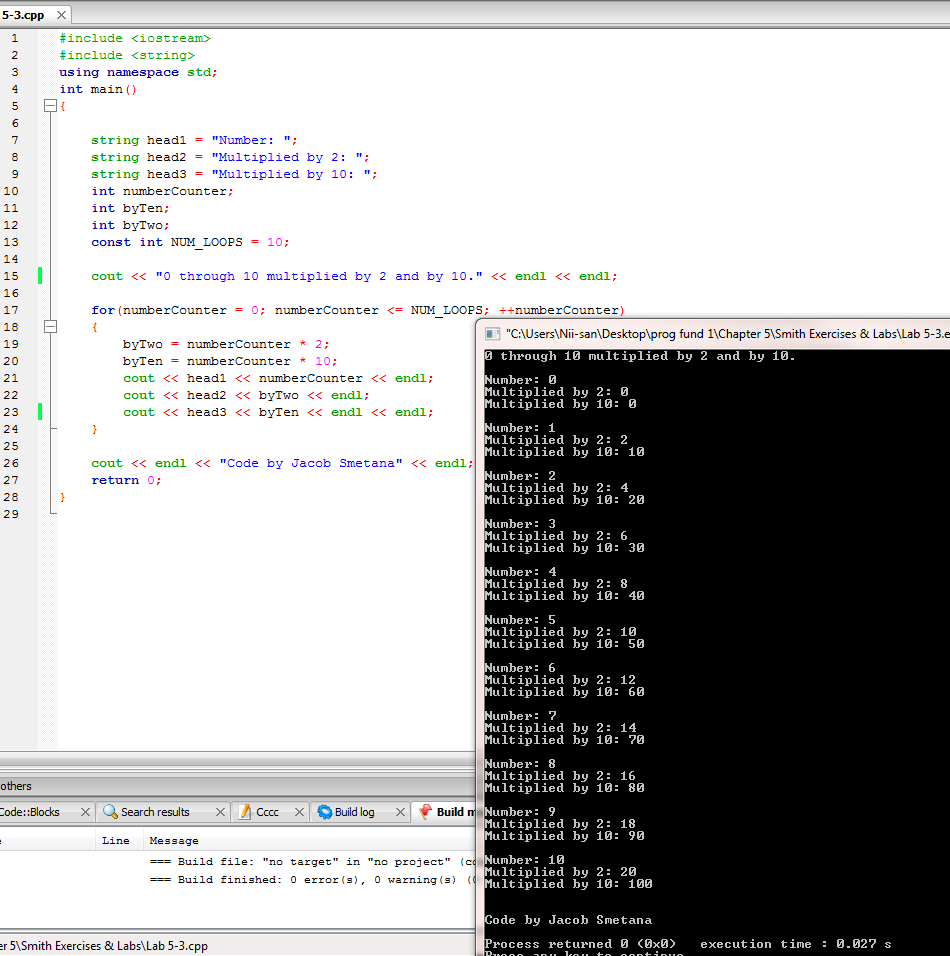
1. Changing the **<=** operator to **<** will make no difference in the output.

True

1. This loop executes 6 times.

True

Lab 5-3.



Exercise 5-6.

1. How many times does this loop execute?

3 times

1. What is the output of this program?

Strike 1

Strike 2

Strike 3

1. Is the output different if you change the order of the statements in the body of the loop, so that **counter++** comes after the output statement?

Yes. The output becomes: Strike 0

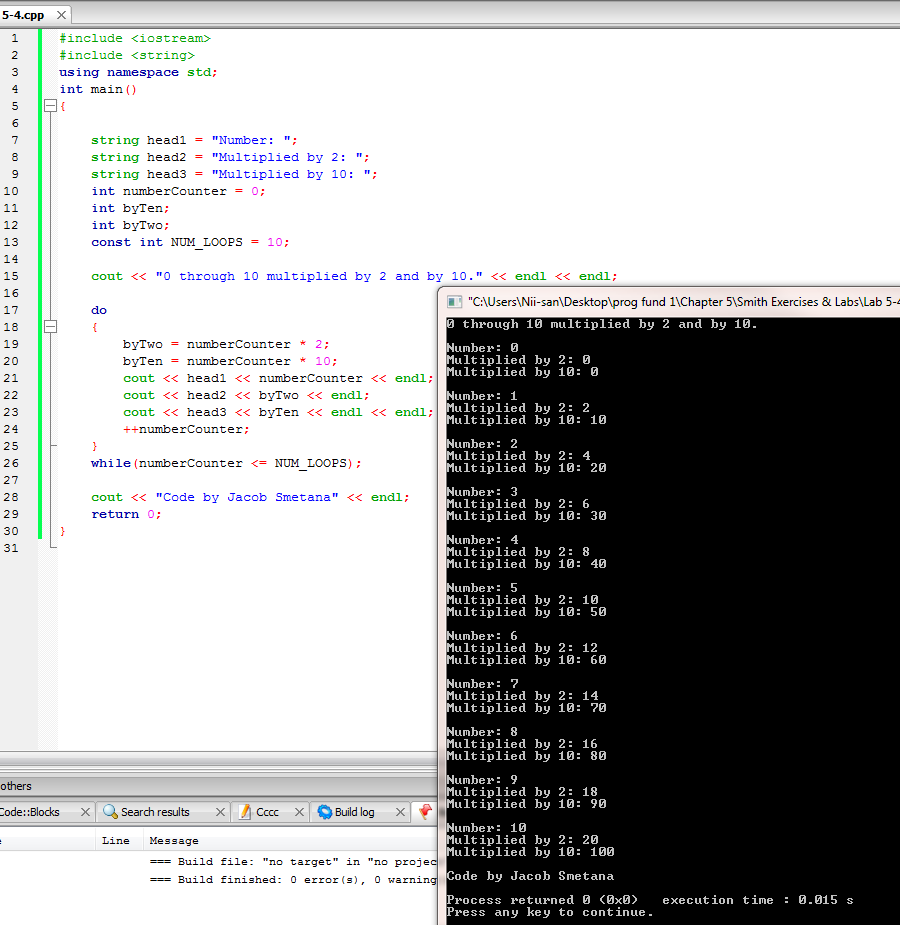
Strike 1

Strike 2

1. What is the loop control variable?

**counter**

Lab 5-4.



Exercise 5-7.

1. How many times does the outer loop execute?

5 times

1. How many times does the inner loop execute?

7 times

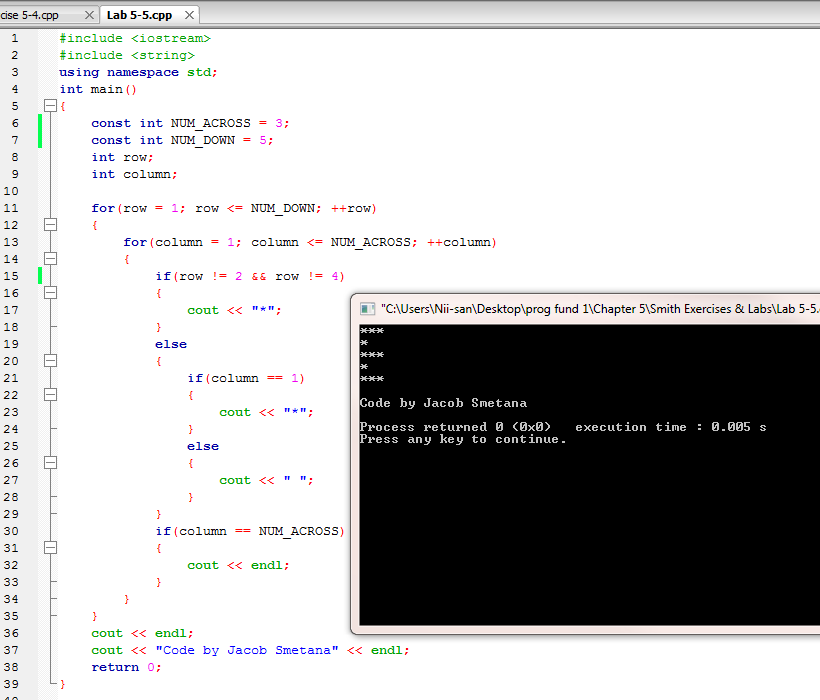
1. What is the value of **sum** printed by **cout**?

175

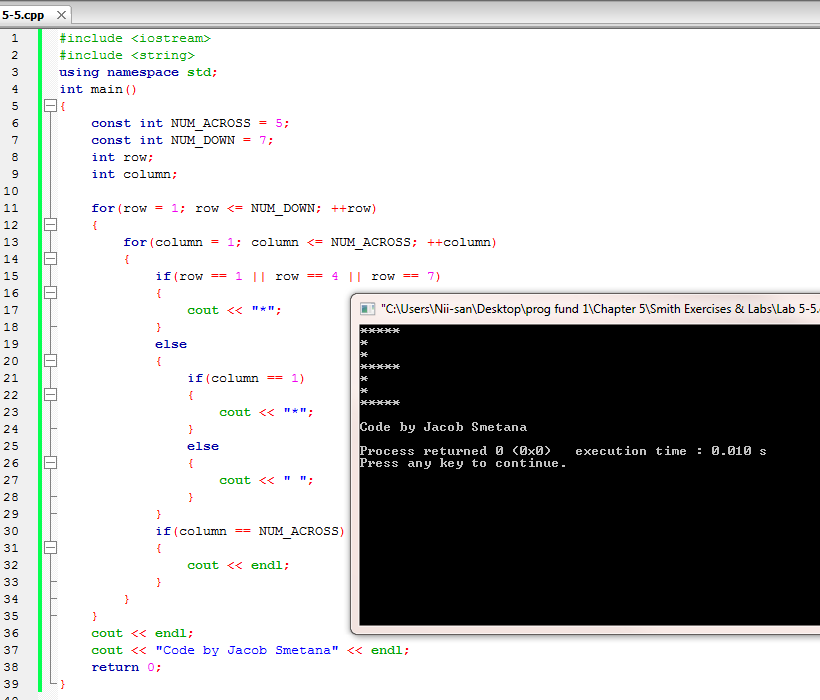
1. What would happen if you changed **rows++** and **columns++** to **++rows** and **++columns**?

Nothing changes

Lab 5-5.



Lab 5-5 Step 7.



Exercise 5-8.

1. What happens when you compile this program if the variable **sum** is not initialized with the value **0**?

Nothing changes

1. Could you replace **sum += rainfall;** with **sum = sum + rainfall;** ?

Yes, you could

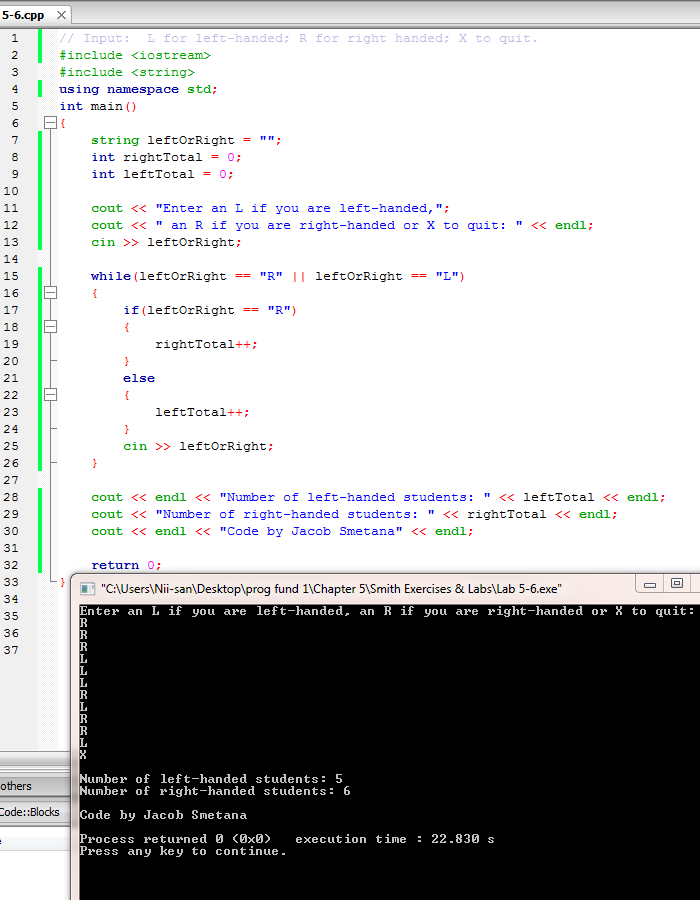
1. The variables **sum**, **rainfall**, and **average** should be declared to be what data type to calculate the most accurate average rainfall?

**double**

1. Could you replace **DAYS\_IN\_WEEK** in the statement **average = sum / DAYS\_IN\_WEEK;** with the variable named **counter** and still get the desired result? Explain.

You would not get the desired result because **counter** would equal **8** at the end of the program.

Lab 5-6.



Exercise 5-9.

1. You plan to use the following statement in a C++ program to validate user input: **while(inputString == “”)**

What would your user enter to cause this test to be true?

Nothing. User just presses Enter.

1. You plan to use the following statement in a C++ program to validate user input: **while(userAnswer == “N” || userAnswer == “n”)**

What would a user enter to cause this test to be true?

User could enter either an uppercase or lowercase *n*.

1. You plan to use the following statement in a C++ program to validate user input: **while(userAnswer < 1 || userAnswer > 10)**

What would a user enter to cause this test to be true?

User could enter any number less than 1 or greater than 10 but none including or in between 1 and 10.

Lab 5-7.

