Increment and Decrement Operators, First Loops

https://csci-1301.github.io/about#authors

January 12, 2023 (03:45:00 PM)

Contents

This lab serves multiple goals:

- To introduce increment and decrement operators,
- To understand the difference between prefix and postfix operators,
- To help you implementing your first `while` loops,
- (Optional) to write your first complex loops.

1 Increment and Decrement Operators

1.1 Preamble

Start by answering the following, assuming an `int` variable `i` has been initialized and that its current value is 5.

1. What does `i++` do to `i`?
2. What does `i--` do to `i`?
3. In your own words, can you explain what is the difference between `++i` and `i++`?

1.2 Observe and Confirm

Download, extract, open in your IDE and read the following code[1]. Then, compile and execute it, and study the output carefully to make sure you understand the mechanism of the increment and decrement operators.

Compare your answers from previous section to what you observe in the output. Do your answers match with what you observe in the output?

---

[1]labs/IncrementDecrement/IncrementExample.zip

1.3 Exercise

For each of the following, determine the final value of n.

```
int x = 5;
int n = x++;

int x = 5;
int n = ++x;

int x = 5;
int n = x++ + x++;

int x = 5;
int n = ++x + ++x;

int x = 5, y = 6;
int n = x++ * ++y;

int x = 5;
int n = x++ + --x;
```
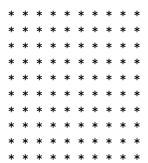
You can test your answers using your IDE.

2 First While Loops

The following is asking to write a series of **while** loops performing simple tasks. The solution to the first question is in this archive[2], but it is recommended to try on your own first.

1. Write a **while** loop that displays the integers between 1 and 100 on the screen, with a space between them.
2. Write a **while** loop that displays the integers between 100 and −100 on the screen, in decreasing order, with a space between them.
3. Write a **while** loop that displays the ∗ (asterisk symbol) character 100 times on the screen.
4. Modify your previous loop, so that a new line character is displayed on the screen every time 10 asterisk symbols have been displayed on the screen.

That is, your program should display this on the screen (this example has a space after each asterisk symbol for display purposes):
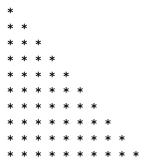
```
* * * * * * * * * *
* * * * * * * * * *
* * * * * * * * * *
* * * * * * * * * *
* * * * * * * * * *
* * * * * * * * * *
* * * * * * * * * *
* * * * * * * * * *
* * * * * * * * * *
* * * * * * * * * *
```

---

[2]labs/IncrementDecrement/FirstLoop.zip

3 Pushing Further (Optional)

Here are additional pattern problems, similar to the last one from the previous section. Every time, you are asked to generate those patterns using a **while** loop.

    1. Triangle:

```
*
* *
* * *
* * * *
* * * * *
* * * * * *
* * * * * * *
* * * * * * * *
* * * * * * * * *
* * * * * * * * * *
```

    1. Triangle of numbers

```
1
222
33333
4444444
555555555
```

    1. Upside-down binary triangle

```
1010101
10101
101
1
```