
Loops

CS150 Introduction to Computer Science 1 1

Multiple Assignments

C++ allows statements such as:

```
a = b = c = d = 45;
```

- What is stored in a, b, c, and d?
- What is the associativity of the assignment operator?

2

Combined Assignments

- We have seen that the same variable can be used on the left hand side of the assignment and on the right hand side

```
notes = notes / 20;  
note = notes % 20;
```

- These are common in programming, so the two operators can be combined as follows:

```
notes /= 20;  
note %= 20;
```

3

Combined Assignments

- Combined assignments can be combined with arithmetic operators

```
y -= a * 2;
```

```
a /= b + c;
```

```
c %= d - 3;
```

- What is the long form of these statements?

4

Increment and Decrement Operators

- C++ provides a shortcut to increment or decrement a variable by 1
 - Always by 1

```
int x = 99;
```

```
x++; // this is equivalent to x += 1
```

```
x--; // this is equivalent to x -= 1
```

CS150 Introduction to Computer Science 1

5

Prefix and Postfix

Prefix

```
k = --x;
```

```
k = ++x;
```

Increment/decrement x
then assign value of x to k

Postfix

```
k = x--;
```

```
k = x++;
```

Assign value of x to k,
then increment or
decrement x

6

What is the Output?

```
int y = 0, x = 0, z = 0;
x = y++;
cout << x << " " << y << " "
    << z << endl;
y = ++z;
cout << x << " " << y << " "
    << z << endl;
z = x++ + 1;
cout << x << " " << y << " "
    << z << endl;
```

7

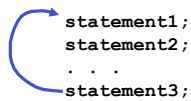
Loop!

- So far, we can
 - Get input
 - Produce output
 - Calculate
 - Conditionally execute statements

8

Loops

- Perform the same bit of code many times



- Why might we want to do this?

9

while Loop

- while the expression is true, loop!

```
while (expression)
{
    statement1;
    statement2;
    . . .
    statement3;
}
statement4;
```

¹Test the expression
²Either
Perform the statements
in the loop if expression is true
³Repeat
or
Move past the loop if
expression is false

Example: What happens?

```
int count = 0;
while (count < 4)
{
    cout << "I'm looping! ";
    cout << count << endl;
    count ++;
}
cout << endl << "I'm done looping. ";
cout << count << endl;
```

Counters

- Counter: A variable that is incremented or decremented each time a loop

```
int theCounter = 0; // initialize the counter
while (theCounter < 5) // test the counter
{
    cout << "*****" << endl;
    theCounter ++; // update the counter
}
```

Key Ingredients of `while` loops

- Initialize
- Test
- Update

If any one of these is missing or incorrect, your loop won't run properly--not at all, too many/ few times or infinitely.

What is the Output?

```
int theCounter = 0;    // initialize the counter

while (theCounter < 5) // test the counter
{
    cout << "*****" << endl;
}
```

Practice

- Write a snippet of code that will print all the integers from 0 to 20 using a while loop.
- Write a snippet of code that will print a table containing the integers from 0 to 20 and their squares.
