COMPOUND OPERATORS, WHILE LOOPS, FOR LOOPS

LOOPS ~ The Roundabouts of Code

Loops

- Think of loops like doing laps around a viewing pullout on a bike trail
    - You are still trying to get from the beginning to the end of main, but sometimes you have to go over a place many times to see what you want to see

- Loops are very useful - they:
    - make code shorter and easier to read
    - make code simpler
    - make code easier to modify

Types of Loops

while loop

for loop

do while loop

While Loop Syntax

```
while (thisExpressionIsTrue) {
   //Code that repeats each time through the loop
}
```

//NOTE: Any variables declared inside of a loop can only be used in the loop

Counted vs Sentential Loops

Try Figuring out these problems by yourself then check the next slide for the answer!

Sentential

```
int i= 0;
cin >> i;
while (i != -1) {
  cout << “No -1 yet!”
   << endl;
  cin >> i;
}
```

What does this print if the user inputs 0 1 20 -5 -1?

Counted

```
int i = 7;
while (i <= 11) {
  cout << “~ ”;
  ++i;
}
```

What does this print?

Counted vs Sentential Loops

Try Figuring out these problems by yourself then check the next slide for the answer!

Sentential

```
int i= 0;
cin >> i;
while (i != -1) {
  cout << “No -1 yet!”
   << endl;
  cin >> i;
}
```

Answer:
No -1 yet!
No -1 yet!
No -1 yet!

What does this print if the user inputs 0 1 20 -5 -1?

Counted

```
int i = 7;
while (i <= 11) {
  cout << “~ ”;
  ++i;
}
```

What does this print?

Answer:
~~~~~~
~~~~~~

Common Loop Problems

- infinite loop:
  - never do a while(true) loop!!!

```
int x = 4;
while (x < 10)
  x--;
```

- off by one:

```
//print 10 *s
int x = 10;
while (x >= 0) {
  cout << “*”;
  x--;
}
```

show on xCode

While Loop Practice Problems

What prints?

```
int x = 10;
while (x != 2.5) {
  cout << x << endl;
  x /= 2;
}
```

Try coding these to find the answer!

What prints?

```
int sum = 0;
int i = 0;
while (i != −1)  {
  cin >> i;
  sum += i;
}
cout  <<  sum << “ “
   << i;
```

User enters: 2 5 -4 6 -1

For Loops!!

- The loop to use when you want to do a task a specific number of times.
- If you find yourself writing a huge list of if (check), you might want to try a for loop:
  - ex: void printTriangeleNums(int val);
- Loops help simplify and clarify code
- Be careful when determining the boundaries of a loop

Loop Syntax

For Loop

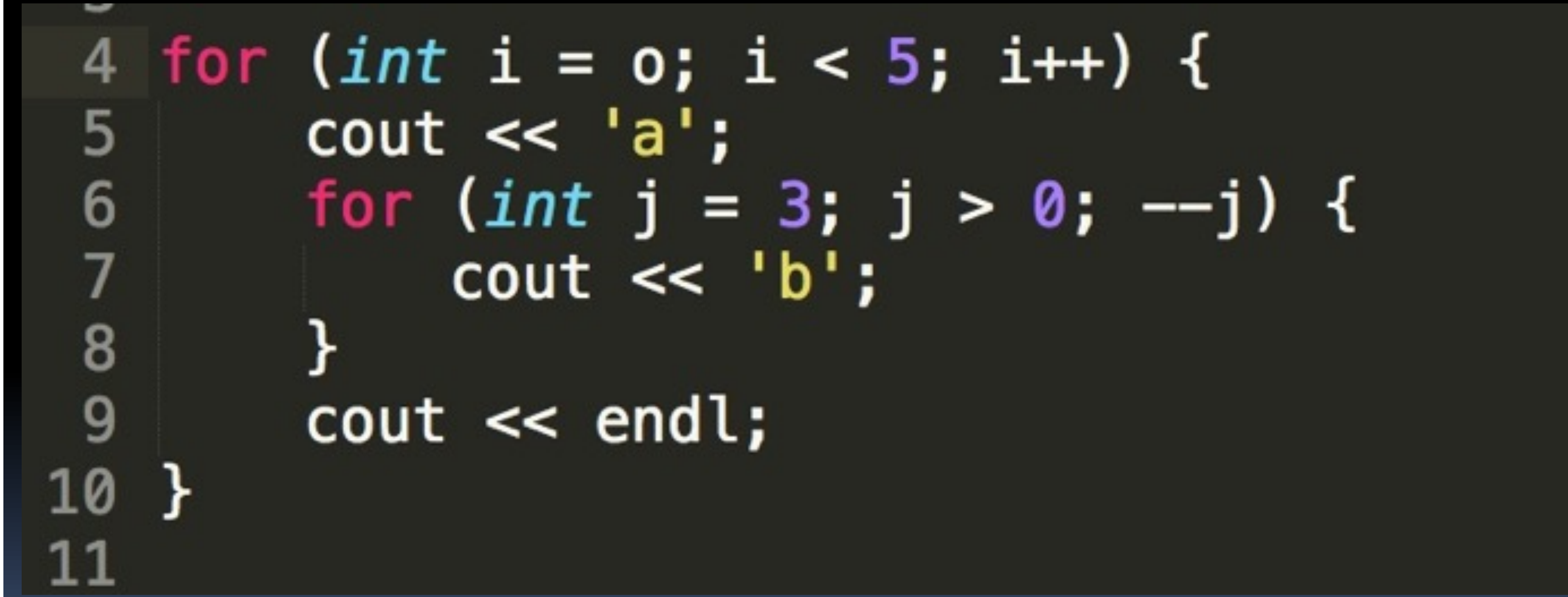
```
for (initialization; condition; update)  {
        //Loop body
}
```

vs

While Loop

```
initialization
while (condition) {
    //Loop Body
   update
}
```

For Loop Examples:



```
for (int i = o; i < 5; i++) {
    cout << 'a';
    for (int j = 3; j > 0; --j) {
        cout << 'b';
    }
    cout << endl;
}
```

Another For Loop: What does this print?

```
bool fun = true;
int count = 8;
for (int i = 3; i < count; ++i, --count)  {
  cout << count << " " << i << endl;
  for (int k = i; (k > 0) && fun; --k) {
    cout << "FUN!" << endl;
    if (k = 3) {          //NOTICE THE SINGLE EQUAL SIGN…
      fun = false;
      }
   }
  fun = true;
}
```

Writing a for loop function:

```
//Requires:  n >= 0
//Modifies: Nothing
//Effects: prints the first n Triangle numbers.
//                    A Triangle number is defined as any
//                 number of the form m(m+1)/2,
//                  where m is a constant.  eg: 1, 3, 6,
 //                     10.....

void printTriangleNums(int n);
```