CS 111

for loops

Model for for loop

```
for (INITIALIZE COUNTER; IS THE COUNTER VALID?; ADJUST COUNTER){
        ACTION;
}
```

- There are three steps involving the counter, all contained within the for loop parentheses
  - Initialization
  - Check to see if counter is still valid
  - If counter is valid, adjust counter value

while loop vs. for loop

While Loop	For Loop

int counter = 1, num;
cout << "Enter a number: ";
cin >> num;
while(counter <= 20){
  cout << num * counter << " ";
  counter++;
}
	int num;
	cout << "Enter a number: ";
	cin >> num;
	for (int counter = 1; counter <= 20; counter++){
	  cout << num * counter << " ";
	}

Example 1

```
#include<iostream>
using namespace std;
int main(){
    for(int c = 5; c < 8; c++){
        cout << c;
    }
    return 0;
}
```

Example 2

```
#include<iostream>
using namespace std;
int main(){
    for(int c = 5; c < 12; c = c + 3){
        if(c < 7){
            cout << "C";
        } else {
            cout << "+";
       }
    }
    return 0;
}
```

Three things to decide

1. Where does the counter begin?
2. How do we test if it is valid?
3. How do we adjust it?

for (Item 1; Item 2; Item 3)

9.2 Pseudocode

// Count number of odd inputs from user

Declare integer variables to store user input n, and a counter for the odd numbers

Read in value from user

Copy the value n from user into a separate variable

Using a for loop receive n more numbers from user

    Read in value from user

    Increment the odd counter for each odd entry

Print the number of odd entries