CECS 174 – Lecture 19 – Methods Continued

Return Type Method – a method that returns a value back to the location from where it was called.

Method Definition – Similar in structure to the method definition for a void method (which does not return a value), these methods return a value back to where they were called. The only difference is that instead of a return type of void, you need to specify what the return type will be (ex. int, double, String).

```
public static int methodName( int par1, int par2 ) {
     int value = 0;
     //statements
     return value;
}
```

Method Call – The method is called using its name and any parameters that were defined. However, since the method now has a specified return type, we must have a place to store that returning value. The return type of the method, and the variable being used to store the returning value should match.

```
int num = methodName( p1, p2 );
```

Example 1: return method with no parameters

```
public class TestMethod {
     static Scanner in = new Scanner(System.in);
     public static void main (String [] args) {
          int x = addFive( );
          System.out.println("X is " + x);

     }
     //adds 5 to the user's input, then returns it
     public static int addFive( ) {
          int num1 = in.nextInt();
          int result = num1 + 5;
          return result;
     }
}

/* Output
X is 8
*/
```

The method is called from main and the method computes the sum of the two given variables. The result is returned back to the main method and the value of the sum is placed into the variable x.

Example 2: return method with parameters

```
public class TestMethod {
     public static void main (String [] args) {
          int a = 3;
          int b = 5;
          int x = sum ( a, b );
          System.out.println("X is " + x);
     }
     //Passes in 2 ints then returns the sum
     public static int sum(int num1, int num2){
          return num1 + num2;
     }
}

/* Output
X is 8
*/
```

The method is called from main with two parameters. The method assigns the two parameters to the two variables and then computes the sum of the two variables. The result is returned and the value of the sum is placed into the variable x.

Example 3: a method calling another method

```
public class TestMethod {
     public static void main (String [] args) {
          int v = mult ( 3, 5 );
          System.out.println("Value is " + v);
     }
     //Passes in 2 ints then returns the sum
     public static int sum(int num1, int num2){
          return num1 + num2;
     }

     //Passes in 2 ints then returns the product
     public static int mult(int num1, int num2){
          int product = 0;
          for(int x=0; x < num2; x++){
               product = sum(product, num1);
          }
          return product;
     }
}

/* Output
Value is 15
*/
```

CECS 174 – Lecture 19 Worksheet – Write the following methods:

1. Write a method called maxOfThree that passes in three integer parameters and returns the maximum value.

2. Write a method that passes in a string. Have the method return true if the word is a palindrome, false otherwise.

3. Write a method called checkInput that takes in two integers specifying the bounds of a range. Have the method repeatedly prompt the user for input until they enter a valid integer that is within the specified range, at which point it returns that value.