5.3 Parameter Passing

Today you're going to look at parameter passing. When a method is called, variables are created for receiving the method's arguments. These variables are called **parameter variables**. The values that are supplied to the method when it is called are the **arguments** of the call. Each parameter variable is initialized with the corresponding argument.

For example. Let's look back at the example from last week – Volume of a Cube.

In order to pass a parameter and calculate the volume, we need to declare a variable called `result1` – `result1` is the **parameter variable**. Then assign it to the `cubeVolume` **method** and supply the method with an **argument** of **2**.

```
double result1 = cubeVolume(2);
```

Let's look at the entire program Cubes.java:

```
1     public class Cubes {
2           public static void main(String[] args) {
3                 double result1 = cubeVolume(2);
4                 double result2 = cubeVolume(10);
5                 System.out.println("A cube with side length 2 has volume " + result1);
6                 System.out.println("A cube with side length 10 has volume " + result2);
7           }
8
9           /** Computes the volume of a cube.
10           * @param sideLength the side length of the cube
11           * @return the volume
12           */
13
14          public static double cubeVolume(double sideLength) {
15                double volume = sideLength * sideLength * sideLength;
16                return volume;
17          }
18    }
```

Here's a look at how the program runs:

- The parameter variable `sideLength` of the `cubeVolume` method is created when the method is called.
- The parameter variable is initialized with the value of the argument that was passed in the call. (`sideLength` is set to 2).
- The method computers the expression `sideLength * sideLength * sideLength`, which has the value of 8. That value is stored in the variable volume.
- The method returns. All of it's variables are removed. The return value is transferred to the *caller*, that is, the method calling the `cubeVolume` method.

AP Computer Science A

Name: _____ Date: _____

5.3 Parameter Passing

Answer the following questions. Problem #5 needs to be uploaded to GitHub.

1. What does this program print?

```
public class Main {
	public static double mystery(int x, int y) {
		double z = x + y;
		z = z / 2.0;
		return z;
	}

	public static void main(String[] args) {
		int a = 5;
		int b = 7;
		System.out.println(mystery(a, b));
	}
}
```

Answer #1: _____

2. What does this program print?

```
public class Main {
	public static int mystery(int x) {
		int y = x * x;
		return y;
	}

	public static void main(String[] args) {
		int a = 4;
		System.out.println(mystery(a + 1));
	}
}
```

Answer #2: _____

3. What does this program print?

```
public class Main {
	public static int mystery(int n) {
		n++;
		n++;
		return n;
	}

	public static void main(String[] args) {
		int a = 5;
		System.out.println(mystery(a));
	}
}
```

Answer #3: _____

4. Consider these methods:

```
public static double f(double x) { return g(x) + Math.sqrt(h(x)); }
public static double g(double x) { return 4 * h(x); }
public static double h(double x) { return x * x + k(x) - 1; }
public static double k(double x) { return 2 * (x + 1); }
```

Without actually compiling and running a program, determine the results of the following method calls.

```
a. double x1 = f(2);
```

Answer 4a: _____

```
b. double x2 = g(h(2));
```

Answer 4b: _____

```
c. double x3 = k(g(2) + h(2));
```

Answer 4c: _____

```
d. double x4 = f(0) + f(1) + f(2);
```

Answer 4d: _____

```
e. double x5 = f(-1) + g(-1) + h(-1) + k(-1);
```

Answer 4e: _____

5. It is a well-known phenomenon that most people are easily able to read a text who's words have two characters flipped, provided the first and last letter of each word are not changed. For example,

I dn'ot gvie a dman for a man taht can olny sepll a wrod one way. (Mrak Taiwn)

Write a method String scramble(String word) that constructs a scrambled version of a given word, randomly flipping two characters other than the first and last one. Then write a program that reads words and prints the scrambled words.