Overview:

In this homework you will write several methods. Each method below should operate as specified. In particular, pay attention to the number and type of input parameters, the return value type, and the name of the function. Not following these specifications will cause you to lose points. For each function, I have provided several examples of possible input(s) and expected output. However, these few examples are not necessarily sufficient to ascertain that your function is entirely correct. You should test your methods as much as needed by placing calls to in in your `main` method.

Getting started:

1. Log in to a lab machine, and open a terminal.

2. At the prompt in the terminal, move into your cs170 directory, and create a new directory for this homework.
   ```
   cd cs170
   mkdir hw2
   ```

3. open gedit by typing
   ```
   gedit &
   ```

4. Name your class `Homework2`:
   ```
   public class Homework2 {  }
   ```

5. Add your `main` method:
   ```
   public static void main(String[] args) {  }
   ```

6. Save your file and then compile and run it to make sure you have everything working:
   ```
   javac Homework2.java java Homework2
   ```

7. One by one, add the functions below. It is highly recommended that you work on one function at a time. Write, compile, and test frequently. Do NOT try to write all 6 functions at once.

Method Specifications:

1. Write a function named `initial` which takes no parameters and does not return any values. This function should use the `System.out.println` function to print the first character of your first name. The initial should be at least 12 lines high. The example below is done with the number 8 simply to demonstrate the concept. You may use any symbols in your letter.

```
        ****
       ******
      **    **
     **        **
      **    **
       ******
       ******
      **    **
     **        **
      **    **
       ******
        ****
```

2. A Caesar Cipher is a cipher which "shifts" a letter of the alphabet to a new letter and allows a message to be "encoded". You can read more about Caesar Ciphers here: wikipedia. Implement a slightly different Caesar Cipher based on the ASCII table. Write a function named `asciiCipher` which takes a String and an integer (in that order) as input parameters and returns an "encrypted" version of the String in which each character in the original has been shifted by the integer value specified by the parameter. Each input String will have exactly 4 characters. You may assume the specified inputs would never cause you to have an ASCII value less than 0 or greater than 127.
   Examples:

   `asciiCipher("book"), 15)` returns `"q~~z"`   `asciiCipher("look", -32)` returns `"LOOK"`

3. A furlong is an old measure of distance dating back to medieval times, when it was used to measure the length of a plot of farm land. Today it is primarily used to measure the distance of horse racing tracks. The track at Belmont Park is 12 furlongs from start to finish. To covert to meters, 1 furlong is equal to 201.168 meters. Write a function named `furlongsToMeters` which takes an integer representing the number of furlongs as input and returns a double representing the equivalent number of miles.

   `furlongsToMeters(1)` returns `201.168` `furlongsToMeters(12)` returns `2414.016`

4. In the standard edition of the board game Monopoly, the money is given in denominations of $1, 5, 10, 20, 50, 100 and 500. Write a function named `makeChange` which takes in an integer representing the number of dollars. The method should print the optimal number of each type of denomination needed for the banker to make change for the given amount in the form below.
   Examples:

   ```
   makeChange(745)
       745 dollars is 1 ($500), 2 ($100), 0 ($50), 2 ($20), 0 ($10), 1 ($5), 0 ($1)
   ```

5. You were the lucky winner of magical seeds that grow into kittens. You are really excited, however, you have limited space to keep kittens. Therefore, you wish to know how many kittens a certain amount of magical seeds will grow. You know that one seed can grow 3 kittens at a minimum and 7 at a maximum. Write a function named `magicKittenSeeds` that accepts an integer (number of magical seeds) as a parameter and that will print the minimum and maximum amount of kittens you could get. You may assume your function is always invoked with a number greater than or equal to 0. Examples:

```
magicKittenSeeds(5)
    You can get between 15 and 35 kittens with 5 seeds.
magicKittenSeeds(8)
    You can get between 24 and 57 kittens with 5 seeds.
```

6. Suppose that you have a savings account that adds interest each month. If the interest rate of the account is 5% per month, and at the beginning of the month the account has $100, then at the end of the month, the account will have
$$100(1 + 0.05) = 105.$$

If nothing else is added to the account, then after the second month, the account will have

$$105(1 + 0.05) = 110.25$$

Write a function named `futureValue` that accepts a double (the initial amount in the account) and an integer (the percent interest rate of the account). The function should return the value of the account after 4 months.
Examples:

```
futureValue(100, 5)
    The value of the account after 4 months is $121.55
futureValue(342.10, 8)
    The value of the account after 4 months is $465.42
```

Submission:
Don't forget the required honor code statement (see the syllabus if you are unsure about this). Also don't forget that every function that you write should have "high level" comments describing (at minimum) the required inputs, expected output, and purpose of each function.

Make sure you're in your cs170/hw2 directory and type the turn-in command
    `/home/cs17000a/turnin Homework2.java hw2`

Make sure you see a message that says "+++Allowed". This means that your HW was turned in correctly. If you submit more than once, you will see a prompt asking you if you are sure you want to overwrite a previous submission. You will need to type "y" to complete the submission!