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 it in your main method.

Getting started:
1. Log in to a lab machine and open a terminal. (If you're unsure how to do this, see labs 1&2 or visit the TA office hours.
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 Homework 2 {   }
5. Add your main method:
   public static void main(String[] args) {   }
6. Save your file and then compile and run to make sure you have everything working:
   javac Homework2.java
   java Homework2
7. One by one, add the functions below. It is highly recommended 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 letter 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](https://en.wikipedia.org/wiki/ Caesar_cipher). 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 parameter and returns an “encrypted” version of the String in which each character in the original String 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. In the classic computer game *Zork*, the Great Underground Empire had its own system of measurements. *Bloits* were a unit of distance measurement. Assume 1 bloit is equal to 1.3 miles. Write a function named `bloitsToMiles` which takes an integer representing the number of bloits as input and returns a double representing the equivalent number of miles.

```
bloitsToMiles(3) returns 3.9
bloitsToMiles(1) returns 1.3
```

Don't worry if the answer your code returns has extra digits of precision like 1.30000000000000001 or similar.

4. Write a function named `makeChange` which takes an integer representing the number of cents as an input parameter. This method should print the optimal number of dollars, quarters, dimes, nickels, and pennies needed to make change for the given amount in the formal below. There are:

- 100 cents in a dollar
- 25 cents in a quarter
10 cents in a dime
5 cents in a dime
1 cent in a penny

Examples:
makeChange(25)
25 cents is 0 dollars, 1 quarters, 0 dimes, 0 nickels, and 0 pennies
makeChange(24)
24 cents is 0 dollars, 0 quarters, 2 dimes, 0 nickels, and 4 pennies

You were the lucky winner of magical seeds that grow into puppies. You are really excited, however, you have limited space (dorm room!) to keep puppies. Therefore, you wish to know how many puppies a certain amount of magical seeds will grow. You know that one seed can grow 2 puppies at a minimum and 5 at a maximum. Write a function named magicPuppySeeds that accepts an integer (number of magical seeds) as a parameter and that will print the minimum and maximum amount of puppies you could get. You may assume your function is always invoked with a number greater than or equal to 0.

Examples:
magicPuppySeeds(5)
You can get between 10 and 25 puppies with 5 seeds.
magicPuppySeeds(8)
You can get between 16 and 40 puppies with 8 seeds.

6. A 2015 Super Bowl advertisement cost $150,000 for 1 second of viewing time. You are the CEO of the “Selfie Toaster” corporation and want to spend all your company’s venture capital money on a Super Bowl ad. You can only buy ads in multiples of 30 seconds (ie, ads that are 30, 60, 90... seconds) long. Write a function named superBowlAd which takes an integer representing the number of dollars you have to spend. You function should return the number of seconds of airtime you can buy with your money. You may assume the function is always invoked with a number greater than or equal to 0.

Examples:
superBowlAd(4500000) returns 30
superBowlAd(4500100) returns 30

Submission:
Don't forget the required collaboration statement (see the syllabus if you're unsure about this). Also don't forget that every function 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/cs170001/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!