Your Name:

Lab#9: Week of Nov 16, 2020

Due by midnight on Nov 20, 2020.

Question 1. Consider searching for the value `x` in the sorted list `ls` shown below.

```
String[] ls = {-56,   -23,    -10,    -5,     -2,    -1,   0}
index              0     1       2      3       4       5     6
```

(a). How many steps does it take to search for the value -10 in the list using binary search? Show the steps in the chart below:

```
low    mid   high    ls[mid]
----------------    -------


---    ---   ---       ---


---    ---   ---       ---


---    ---   ---       ---


---    ---   ---       ---


---    ---   ---       ---
```

(b). How many steps does it take to search for the value 3 in the list using binary search? Show the steps in the chart below:

```
low    mid   high    ls[mid]
----------------    -------


---    ---   ---       ---


---    ---   ---       ---


---    ---   ---       ---


---    ---   ---       ---


---    ---   ---       ---
```

(c). Write a function that takes a String as input and returns a new String which doubles each letter. For example, if the user inputs "apple", the function returns "aappppllee". If the user inputs "fun", the function returns "ffuunn". Test your function in setup with the inputs "apple", "fun", and "bumble bee". Print the results to the console to check your work. You can submit your code to dropbox or write the code here.

```
// Function: doubleLetters
// Input: phrase (String), the phrase to modify
// Output: (String), the phrase with all letters doubled
```

```
void setup() {

    // todo: test your function and print the results to the console

}
```

(d). Write a function that takes a String as input and counts the number of times a given letter occurs in it. Your function should iterate through each character in the string and use an accumulator variable to count the number of occurances (e.g. don't use methods defined in String, except for "length()" and "chatAt()"). For example, if the user inputs "apple" and the letter "p", the function should return 2. Test your function in setup. You can submit your code to dropbox or write the code here.

```
// Function: count
// Input: phrase (String), the phrase to count letters within
// Input: letter (char), the letter to count in phrase
// Output: (int), the number of times letter appears in phrase
```

```
void setup() {

    // todo: test your function and print the results to the console
    // test: "apple", "p"
    // test: "apple", "z"
    // test: "bumble bee", "b"

}
```