String Lab

1) Consider the following method. What value is returned by the call `eval("jamjarjax", "ja")`?

```
int eval(String str, String check) {
  int m = str.length()/2;
  String a = str.substring(0, m);
  String b = str.substring(m);
  return a.indexOf(check) + b.indexOf(check);
}
```

2) Consider the following code. What will be printed?

```
void mystery(String str) {
  if (str.length() < 4) {
    println("D");
  }
  else {
    print(str.substring(0, 1));
    mystery(str.substring(1));
    println(str.substring(0, 1));
  }
}

mystery("BELLE");
```

3) Write a function `int lastIndexOf(String str, String substr)`, which returns the starting index of the last occurrence of `substr` in `str`.

4) Write a function `stringSum` that splits the numbers in its parameter string, converts them to floats, and returns the total. You may assume that all numbers are comma+single-space separated and they are all floats. However, your code should work for arbitrarily many numbers. For example, when called with `myNums` as below, `stringSum` should return 13.002846.

```
String myNums = "1.2345, 2.3, .345, 4.0, 5.123345678";
println(stringSum(myNums));
```

5) Write a function `countMatches`, which takes a String array and a target string, and returns the number of times the target string appears in the array. For example, the following calls to `countMatches` will print 3 and 0.

```
String[] strs = {"A", "B", "C", "D", "A", "F", "C", "A"};
println(countMatches(strs, "A"));
println(countMatches(strs, "E"));
```

6) Write a recursive function `boolean palindrome(String str)` that takes a `String` argument `str` and returns `true` if `str` is a palindrome and `false` otherwise.