Uttara Lab - Strings:

First compile and run TestStringMethods.java given as example. Go through the code to recap the methods on String we discussed in class and verify if you understand how it is working. In case you have any doubt, ask the Lab Instructor.

Methods that you need to use are given in the bottom of the document as well. You can also go through given .java files(TestStringMethods.java, TestMath.java, AddArrayElements.java)

1) Write a main() program to test methods of string to perform the following (directly create a string in main() like String str = “abcdef”):
a) check its length
b) print all the chars in string one at a time
c) convert string to array of chars and print chars
d) convert string to uppercase and lowercase and print
e) take 2 strings and check if they are equal (create 2 strings directly in main())
f) take 2 strings and check which is bigger or lesser (alphabetical comparison)
g) take 2 strings and find out if one string occurs in other. Print the first occurring index

2) WAM to swap first and last chars of a passed string and return it.
char at length-1 + substring from 1, length-1 + char at 0

```
str.charAt(str.length()-1) + str.substr(1,str.length()-1) +str.charAt(0)
```

3) If you have not done this in the prior lab, you can do this else go to the next question.
Test passing reference as parameter. Create a Fish class with String name in it. Create a Croc class with String name and a method called eat().Accept a Fish reference as parameter. In the eat(), print both the crocs name and fishs name (public void eat(Fish f)). In TestPass class, code a main() where you should create a croc object, a fish object, assign names “cocky” and “fishy” and invoke eat() and pass fish ref as parameter.Print the fishs name before calling eat() and after calling eat(). Run and test.

Test the following as well:

a) in eat(), after SOP, assign f to null. Check what happens.
b) in eat(), assign f to new Fish object and assign name “Flippy”. Now check the SOPs
c) in eat(), assign f.name to “Flippy” and observe outputs. What happens here? How are references getting passed?
d) pass null to eat() and check what happens

4) WAP to create an array with 10 size and of int holding ability (See TestArray.java). Store values 1-10 in it. Using the length variable, access each element of the array and print it to the console. Create the array in 2 ways - one using the new operator and then storing the values individually, two by creating the array with the values directly in the array.
Loop over the array using an index variable and the enhanced for as well.

5) WAM to test swapping of 2 int nums without temp var

6) WAM to test swapping of 2 strings without temp var

<u>Important problems asked in interviews on Strings:</u>

7) WAM to test whether a given string is a palindrome

8) WAM to accept 2 strings as input and return number of occurrences of second string in first

9) WAM to print only the unique chars in a given string

10) WAP to print all the substrings of a string

11) WAM to convert int to a binary representation and return as a String. Build another method that accepts a string and returns an int in base 10.

Methods to use:

```
1) int len = str.length()
2) for (int i = 0 ; i < str.length() ; i++)
        char c = str.charAt(i);
3) char[] arr = str.toCharArray(); // this gives you an array which you can loop over
4) String s1 = str.toUpperCase(); // str.toLowerCase();
5) boolean result = s1.equals(s2)
6) int pos = s1.indexOf(s2,0) // second param is int position
7) boolean result = s1.contains(s2)
9) String[] sa = str.split(" "); // splits the string on space token
10) int[] arr = {10,20,30}; // int literal array
11) String[] arr = new String[]{"rosey","posey”}; // String literal array
12) for(String s : arr)
        SOP(s);
13) for(int i = 0 ; i < arr.length ; i++)
        SOP(arr[i]);
```

Things to remember:

1) String is a class in Java (capital S)
2) length is a variable in an array and method in a string (arr.length , str.length())
3) You have to capture the returned value from a method to use it: String name = per.getName(); just calling per.getName() will not give you the value.
4) method names of String are fixed. You have to use proper naming convention (camel case).