String Problems PsuedoCodes

24 September 2024

Breaking string to tokens:

```
-----------------------------------
String[] tokens = s1.split("[\\s!?,._'@]+");

print each token using for loop
```

Integer to Binary

```
-----------------------
String output = String.format("%8s", Integer.toBinaryString(num)).replace(' ', '0');

in python: f"{num:08b}
```

FizzBuzz

```
---------------------
int c = word.length();
    if (word.charAt(0) == 'f' && word.charAt(c - 1) == 'b')
       System.out.println("FizzBuzz");
    else {
       if (word.charAt(0) == 'f')
          System.out.println("Fizz");
       else if (word.charAt(c - 1) == 'b')
          System.out.println("Buzz");
       else
          System.out.println(word);
    }
```

Multiply strings based on n Characters

```
-----------------------------------------------------------
FUNCTION multiply(str, n)

  rep = str.substring(length(str) - n)
  result = ""

  FOR i FROM 0 TO n - 1 DO
     result = result + rep
  END FOR

  RETURN result
END FUNCTION

FUNCTION main()
  input = scan.nextLine()

  TRY

     parts = input.split(" ")

     str = parts[0] + " " + parts[1]

     num = INTEGER_PARSE(parts[2])

     PRINT multiply(str, num)
  CATCH Exception e
     PRINT "Error: Input should have exactly 2 strings separated by a space"
  END TRY

END FUNCTION
```

NEW STRING FROM CHARACTER ARRAY

```
-------------------------------------------------------------------------------------------
FUNCTION main()

  n = scanner.nextInt()

  CREATE array a of size n

  FOR i FROM 0 TO n - 1 DO
     a[i] = scanner.next().charAt(0)
  END FOR

     // Read the index and length for the substring
     index = scanner.nextInt()
     length = scanner.nextInt()

     // Extract the substring from the character array
    str = String.copyValueOf(a, index, length)

     PRINT "New String: " + str
```

SUM OF DIGITS IN THE STRING:

```
-------------------------------------------------
FUNCTION sumOfDigits(stng)
  l = LENGTH(stng)
  sum = 0

  FOR i FROM 0 TO l - 1 DO

     IF CHARACTER_IS_DIGIT(stng[i]) THEN
        tmp = SUBSTRING(stng, i, i + 1)
        sum = sum + INTEGER_PARSE(tmp)
     END IF
  END FOR

  RETURN sum
END FUNCTION
```

END FUNCTION

REMOVE T FROM THE GIVEN STRING EXCEPT FIRST AND LAST POSITION

```
---------------------------------------------------------------------------
FUNCTION remove(s)
   result = ""
   l = LENGTH(s)

   FOR i FROM 0 TO l DO
      temp = s.CHAR_AT(I)

      IF  (i > 0 AND i < l - 1 AND (temp == 't' OR temp == 'T')) ==FALSE)
         result = result + temp

      END IF
   END FOR

   RETURN result
END FUNCTION
```

TWIN CHARACTERS

```
-----------------------------------------
FUNCTION appeartwice(s, c)
   // Find the index of the first occurrence of character c in the string s
   i = INDEX_OF(s, c)

   IF i == -1 THEN
      RETURN false
   END IF

   // If the character is at the last index, it cannot be consecutive
   IF i + 1 >= LENGTH(s) THEN
      RETURN false
   END IF

   RETURN  s.substring(i + 1, i + 2).equals(String.valueOf(c));
END FUNCTION

FUNCTION main()

   str1 = scan.nextLine().trim()
   c = scan.next().charAt(0)

   IF s.appeartwice(str1, c) THEN
      PRINT "The given character '" + c + "' appears consecutively in the string '" + str1 + "'"
   ELSE
      PRINT "No consecutive character present"
   END IF
END FUNCTION
```

REVERSE ODD LENGTH WORDS

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

public static String test(String str)
 {
  String[] words = str.split(" ");
  for (int i = 0; i < words.length; i++)
  {
    if (words[i].length() % 2 != 0)
    {
      StringBuilder rev = new StringBuilder(words[i]);
      words[i] = rev.reverse().toString();
    }
  }
  return String.join(" ", words);
```

Partial Array Sorting

```
-----------------------------------
Use Arrays.sort(a,start,end);
```

Remove words from text

```
String result = text.replace(word, "");  //use replace
```

Repeated Characters:

```
public static int morethantwice(String t) {
    int c = 0;
    while (t.length() > 0) {
        if (t.length() - t.replaceAll(t.charAt(0)+"", "").length() > 2)   //charAt(0)+"" is to convert each character to string since, replaceAll works with only strings
        {
            c++;
        }
        t = t.replaceAll(t.charAt(0) + "", "");
    }
    return c;
}
```

if input is : abaa: first character is a, after replacing all a string becomes b. original length is 4-1=3, which means a is appearing more than twice.

Check for second in first

```
public static boolean test(String first, String second) {
    if (second.length() > first.length())
        return false;
    if (second.isEmpty())
        return false;
    for (int i = 0; i < second.length(); i++) {
        if (!first.contains(String.valueOf(second.charAt(i))))
            return false;
    }
    return true;
}
```