Cairo University
Faculty of Engineering
Computer Engineering
Fall 2018

Design and Analysis of Algorithms
Lab 5

Strings

1. Given two strings, write a function to decide if one is a permutation of the other. Take case sensitivity into consideration.

2. Given two strings s1 and s2, write a function to determine if s2 is a rotation of s1 using one function call only of the *substring* library function.

3. Given a string, print all unique characters inside the string in reverse order of appearance inside it. For example, given 'aafbacbedf', the output is 'fdebca'.

4. Assuming all integers are concatenated after each other, write a function that would return the digit at the given index in such long string of integers ('12345678910111213141 5…99100101…'). Do not construct the string, just compute the digit at the specified location. Assume indexing starts from 1. Examples: $9 \rightarrow 9$, $10 \rightarrow 1$, $11 \rightarrow 0$, $189 \rightarrow 9$ and $190 \rightarrow 1$.

5. Given an input string and an ordering string, check whether the characters in the input string follows the strict order in the ordering string or not. For example:

    a. Given input string 'cairo university' and the ordering string 'civs', the function should return False since 'i' appears after 'v' and 's' in the input string.
    b. Given input string 'faculty of engineering!' and the ordering string 'acer!', the function should return True.

6. Given a pattern string and a text string, check whether the text string matches inside the pattern string or not. The pattern string can have either small letters, '*' or '?'. '?' represents one character, '*' represents zero or more characters. Examples: 'non' matches 'n?n', 'xyz' matches 'x*?z', 'algorithm' matches '*a?g*t*hm'.