Algorithms and Data Structures: Stacks & Queues

Assignment - 1

Dr Nagarajan Ganapathy

Version 1

Datum: 12.09.2023

Write the most efficient algorithm for the following problems in C++ programming language and mention the Time and Space Complexity of your algorithms in the comments (at the end):

1. **Create your own stack using class or struct, implement the following functions:[10 pts]**

   - **Top**
   - **Push**
   - **Peek**
   - **isempty**
   - **pop**
   - **size**

2. **Create your own queue using class or struct, implement the following functions: [10 pts]**

   - **Front**
   - **Push**
   - **isempty**
   - **pop**
   - **size**
   - **emplace**

3. **Develop a code that evaluate a string to be beautiful by considering following condition: [20 pts]**

   - **It contains these '{' ; '}' ; '[' ; ']' ; '(' ; ')' characters only.**
   - **The order of the brackets should be valid i.e. all the open brackets should be followed by closed brackets.**
   - **For a type of open bracket the same type of closed bracket should be followed to make the string valid**.

   Thus, now help harry in verifying if the string is beautiful or not. Return true for a beautiful string else false.

Testcases:

a) "{[()]}" is a beautiful string as the order of brackets is valid.
b) "{()}" is a beautiful string as closed flower bracket present.
c) "{}(" isn't a beautiful string as the last character is an open bracket.

4. **Larry is a classmate of Harry, he has been given an array of numbers and his instructor has asked to compute an array of numbers that contain the next greater element of every element in the orignal array of numbers. For the last element and the greatest element the next greater element should be -1. [20 pts]**

   Testcases:

   a) Input array = {1,2,3}

      Output array = {2,3,-1}

   b) Input array = {1,3,2,4}

      Output array = {3,4,4,-1}

   c) Input array = {1,7,4,2,5}

      Output array = {7,-1,5,5,-1}

   d) Input array = {3,3,3,3}

      Output array = {-1,-1,-1,-1}

5. **Harry's Instructor has taught him that the stacks and queues belong to the same family of datastructres and they can be replicated by each other.Hence, Harry got curious to know how stacks and queue can replicate each other. So now help Harry in: [20 pts]**

   - **Implementing queue using stacks**
   - **Implementing stacks using queues**

6. **Larry is fond of strings and he loves to play with them. Hence he took a string and decided remove all the adjacent duplicate characters untill the string has no adjacent duplicate characters, the output string can be empty also. So now help Larry in removing the adjacent duplicate characters you can use any suitable datastructure. [20 pts]**

   Eg:

   a) Input string : abcd

      Output string : abcd

      Explanation: No adjacent duplicate characters.

b) Input string : beebforkeeek

Output string : for

Explanation: first all adjacent e's were removed string became bbforkk

Now adjacent g's and k's are removed string became for.

c) Input string: ggggg

Output string :

Explanation: all characters are duplicate thus empty string is the output.