CSC-361 Homework 4: Linear Data Structures and Algorithms

You must work in a group of 2 or 3 on these exercises; submit only one document for the group.

Algorithms should be described in high-level English. Only provide pseudocode when it adds clarity and / or precision.

Required Problems

1. Suppose that a client performs an intermixed sequence of Stack `push` and `pop` operations. The `push` operations put the integers 0 through 9 in order onto the stack while `pop` operations print out the return values. Which of the following sequences could *not* occur.

   (a) `4 3 2 1 0 9 8 7 6 5`
   (b) `4 6 8 7 5 3 2 9 0 1`
   (c) `2 5 6 7 4 8 9 3 1 0`
   (d) `4 3 2 1 0 5 6 7 8 9`
   (e) `1 2 3 4 5 6 9 8 7 0`
   (f) `0 4 6 5 3 8 1 7 2 9`
   (g) `1 4 7 9 8 6 5 3 0 2`
   (h) `2 1 4 3 6 5 8 7 9 0`

2. Suppose that a client performs an intermixed sequence of Queue `enqueue` and `dequeue` operations. The `enqueue` operations put the integers 0 through 9 in order onto the queue while `dequeue` operations print out the return values. Which of the following sequences could *not* occur.

   (a) `0 1 2 3 4 5 6 7 8 9`
   (b) `4 6 8 7 5 3 2 9 0 1`
   (c) `2 5 6 7 4 8 9 3 1 0`
   (d) `4 3 2 1 0 5 6 7 8 9`

Choose 1 of the Following Problems

1. Give a linear time algorithm to separate the odd and even integers in an integer array: evens on the left and odds on the right.

2. Give an algorithm to sort a stack using a temporary stack.

3. Give a one-pass algorithm to identify the *dominators* in an array of integers. An element is a dominator if it is greater than all elements to its right in the array.

Choose 1 of the Following Problems

1. The `powerset` is the set of all subsets of a given set. For example,

   $$\texttt{powerset}(\{a,b,c\}) = \{\{\},\{a\},\{b\},\{c\},\{a,b\},\{a,c\},\{b,c\},\{a,b,c\}\}.$$

   Give an algorithm to compute the powerset of a given set.

2. Give pseudocode for an algorithm that will verify if an input string is in the language of balanced and nested parentheses: returns true if all parentheses are balanced and false otherwise. For example, the following expressions are all legal, balanced parenthetical expressions: the empty expression, `()`, `(())`, `((())())`, and `()()(()())`. The following expressions are not balanced: `(`, `()())`, `((((()))`, `)(`.