Data Structures (CS201)

Lab Assignment 2 (Graded)

August 21, 2020

Instructor: Anil Shukla
Time: 11 am—11 : 59 pm Total Marks: 10

Note: Graded means the marks will be counted for the final grading. Place proper comments in your source code. Write in C only. C++ is not allowed.
Note: Plagiarism is strictly prohibited. An appropriate disciplinary action will be taken if you are found to be involved in plagiarism.
Note: The instructions for submiting the assignment is mentioned in the google classroom. Carefully read the same and follow the instructions.
Note: At the end, find some test cases for each problem.

(1) Write a linked list implementation of the **unregular list** in C. An **unregular list** is like a regular list, except that all insertions are performed at the front of the list, and when an element is accessed by a FIND (i.e, search), it is moved to the front of the list without changing the relative order of the other items. (5 marks)
After implementing the unregular lists, solve the following problems. Just write the solutions as a comment at the end in your .c source code.

(1a) Suppose we have inserted $n, n-1, n-2, \ldots, 2, 1$ in this order in an empty unregular list. What is the time complexity of inserting the same in an empty unregular lists implemented above by you? (1 mark)

(1b) Assume that the content of our nonregular list is $1, 2, \ldots, n$. Consider the sequence of m operations that repeatedly accesses (searches, for simplicity) n and $n-1$. What is the time complexity for the same. (2 marks)

(1c) Assume that the content of our nonregular list is $1, 2, \ldots, n$. Consider the sequence of operations that accesses i, $(k+n-i)$ times, for each $i \in [n]$, where k is some non-negative integer. What is the time complexity for the same? (2 marks)

(2) Problems related to stacks:

(2a) Write a C-program to implement two stacks using only one array. Your C-program should not declare an overflow unless every slot in the array is used. Your program should first ask user for the size n of the array. (7 mark)

(2b) Show how to implement 3 stacks in one array. Provide the solution to this problem at the end of your .c source code for the problem (2a) above as a comment. No need to implement your ideas for this problem. (3 marks)

Test Cases:

Test Cases for problem (1):
Input: Insert(1), Insert(2), Insert(3), Insert(4), Insert(5)
Display: $5, 4, 3, 2, 1$
Search(2)
Display: $2, 5, 4, 3, 1$

Test Cases for problem (2a):
For different sizes of arrays, we check your program for the overflow and underflow conditions.