Department of Scientific Computing, Modeling and Simulation

SC - 504 Coputation Lab

C programming test - 3

M.Sc. Scientific Computing

Time: 10:15 to 11:45 AM | Date: September 12, 2024 | Max mark: 30

Note:

1. Use dynamic memory allocation whenever dealing with 1D, 2D array.
2. Manage your time efficiently to attempt all questions.

1. **Attempt any two (use while loop):**

   (a) Given an array of integers **nums** and an integer **target**, print indices of the two numbers such that they add up to target. You may assume that each input would have exactly one solution, and you may not use the same element twice. You can print the answer in any order. (3)

   Example :
   Input: *nums* = [2,7,11,15], target = 9, Output = [0,1]
   Explanation: nums[0] + nums[1] == 9, we print [0, 1].

   (b) Given an integer array **nums**, return true if any value appears at least twice in the array, and return false if every element is distinct. (3)

   (c) Given a non-empty array of integers **nums**, every element appears twice except for one. Find that single one. (3)

   Example :
   Input: *nums* = [2,2,1], Output = 1

2. **Attempt ALL:**

   (a) Given an integer array **nums**, return true if any value appears at least twice in the array, and return false if every element is distinct. (4)

   (b) Given a sorted array **nums** of distinct integers and a **target** value, return the index of the target if it is present in the array. If the target is not present then return the index where the target can be inserted. (4)

   (c) Given an $m \times n$ matrix of distinct numbers, return all lucky numbers in the matrix in any order. A lucky number is an element of the matrix such that it is the minimum element in its row and maximum in its column. (4)

   Example :
   Input: matrix = [[3,7,8],[9,11,13],[15,16,17]]
   Output : 15

3. **Attempt any two:**

(a) Read matrix dimension (rows = columns) from user. Dynamically allocate memory for two matrices and perform the following: **(6)**

I. Add these two matrices

II. Subtract these two matrices

III. Multiply these two matrices

(b) Given an $m \times n$ matrix, return **True** if the matrix is Toeplitz. Otherwise, return **False**. A matrix is Toeplitz if every diagonal from top-left to bottom-right has the same elements. **(6)**

1	2	3	4
5	1	2	3
9	5	1	2

Example :

Output: **True**

Explanation: In the above matrix, the diagonals are: "[9]", "[5, 5]", "[1, 1, 1]", "[2, 2, 2]", "[3, 3]", "[4]".

In each diagonal all elements are the same, so the answer is **True**.

(c) You have a long row of benches in a park where some of the benches are occupied, and some are unoccupied. However, no two people can sit on adjacent benches. Given an integer array benchRow containing 0's and 1's, where 0 means unoccupied and 1 means occupied, and an integer n, return true if n new people can sit on the benches without violating the no-adjacent-people rule and false otherwise. **(6)**

Example :

Input: benchRow = [1,0,0,0,1], n = 1 $\Longrightarrow$ Output: **True**

Input: benchRow = [1,0,0,0,1], n = 2 $\Longrightarrow$ Output: **False**

Why do programmers dislike nature?
It has too many bugs!