RIPHAH INTERNATIONAL UNIVERSITY, LAHORE CAMPUS.

RIPHAH SCHOOL OF COMPUTING & INNOVATION (RSCI)

COMPLEX PROBLEM SOLVING

ASSIGNMENT 01

Issue Date: 22-10-2022
Due Date: 28-10-2022
Semester: FALL 2022
Class: BSCS (7A)
Total Marks: 100

Objectives:

- Practice on problem solving.

Instructions:

- Assignment type is individual, so no sharing is allowed.
- You can use internet and books as helping resources but sharing content with peers is strictly prohibited.
- **Plagiarized assignments will get zero and may fail the course.**
- I am available for your help/guidance.
- **Start early!**

Submission Method:

- Write everything in word file. You can also solve on paper and attach pictures in word file.
- Submit only MS Word or PDF file at **this link**.

NAME :	**Muhammad Sameer Sohail**
SAP :	**15000**

Problem A2-1

Given the array of size **4**, containing a four-digit number. Each digit in a number is stored on a single index of array. You are required to add 1 in array

Example1:

Input: 2512 is a single number which is stored in an array

2	5	1	2

Output: 1 is added in 2512 (2512+1=2513)

2	5	1	3

Example 2:

Input: 8999 is a single number which is stored in an array

8	9	9	9

Output: 1 is added in 8999 (8999+1=9000)

9	0	0	0

ALGORITHM:-

- Start
- Take an array of four indexes and store a value.
- Now start parsing the given array from the end.
- If last element = 9 make it 0 and carry = 1.
- Check carry for the following iteration, and if it adds up to 10.
- repeat step 2
- make carry = 0 for next iteration
- if the number is added change the index value and move forward
- Do the same for all array
- end

Problem A2-2

The heights of different peaks at a mountain are given in the array. Each element of array shows the peak height. A climber has some superpowers and can directly jump from one peak to another peak. Your task is to find the maximum jump that a climber made during his journey.

For Example:

Input:

4	6	2	1	7	3	8	5

Output:
6

Explanation:
The maximum jump made is from '1' to '7' elements which is 6.

ALGORITHM:

- start
- take an array of 0 to 7 indexes
- start loop to compare all elements of the array
- declare a variable Jump
- compare the first two indexes and store the max value index number in the jump
- then compare the 3rd index with the jump
- if the 3rd index Is greater the jump, then replace the value
- do this for all the array
- print the highest index number.
- End

Problem A3-3

Given the array of size **n**, find the **Majority Element** from the array. Return -1 if there is no any majority element.

Majority element is an element that exist more than half of the size of array.

Example 1:

Input:

5	1	2	1	6	3	9	5

Output:
-1

Explanation:
There is no any element that exists more than half of the array. So answer is -1.

Algorithm:

- Start
- Take a variable to store the maximum count.
- Traverse the array from start to end.
- Run another loop to find similar values in the array.
- If arr[i] = max count
- Update the maximum count and save the index in a different variable if the count exceeds the maximum count.
- Print the element if the maximum count is greater then half of array
- Otherwise print -1.