BAHRIA UNIVERSITY, (Karachi Campus)

Department of Software Engineering

Assignment #01– Spring 2023

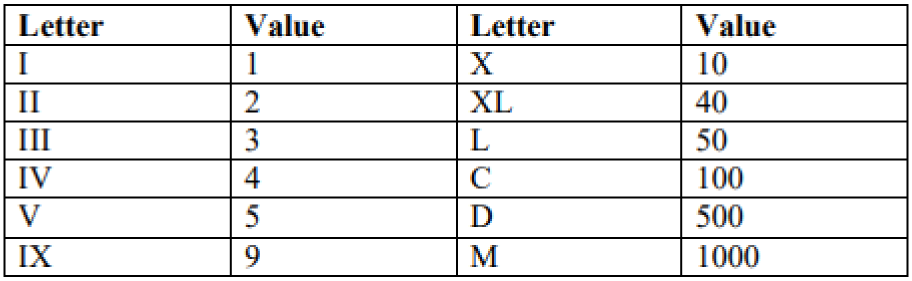
COURSE TITLE:	**D&AA**	COURSE CODE:	**CSC-321**
Class:	**BSE 4A&B**	Shift:	**Morning**
Course Instructor:	**ENGR. BUSHRA FAZAL KHAN**	Assignment Date:	**22-Mar-2023**
Max. Marks:	**5 Points**	Assignment Due:	**29-Mar-2023**

Question # 1: [CLO3:1.5]

Write an algorithm in pseudo-code for converting your name along with your favorite sports personality name (your name and favorite sports personality name) without spaces into roman letters. Firstly, the names should be converted into numbers through ASCII codes then covert that name into roman letters and print as an outcome.

For example, if your name is “Ahmed Khan” and your favorite personality name is “Roger Federer”, the ASCII code for ‘A’ is 65; ‘h’ is 104 and so on. The result will be (65+104+109+101+100+75+104+97+110) +(82+111+103+101+114+70+101+100+101+114 +101+114) = 2077

Recall the following conversion:



Letter	Value	Letter	Value
I	1	X	10
II	2	XL	40
III	3	L	50
IV	4	C	100
V	5	D	500
IX	9	M	1000

Question # 2: [CLO3:1.5]

Write an algorithm in pseudo-code to find square root of a number using Babylonian square-root method.

Suppose you are given any positive number S. To find the square root of S, do the following:

1. Make an initial guess. Guess any positive number x0.
2. Improve the guess. Apply the formula x1 = (x0 + S / x0) / 2. The number x1 is a better approximation to sqrt(S).

3. **3.** Iterate until convergence. Apply the formula xn+1 = (xn + S / xn) / 2 until the process converges. Convergence is achieved when the digits of xn+1 and xn agree to as many decimal places as you desire.

Let's use this algorithm to compute the square root of S = 20 to at least two decimal places.

1. An initial guess is x0 = 10.
2. Apply the formula: x1 = (10 + 20/10)/2 = 6. The number 6 is a better approximation to sqrt(20).
3. Apply the formula again to obtain x2 = (6 + 20/6)/2 = 4.66667. The next iterations are x3 = 4.47619 and x4 = 4.47214.

Because x3 and x4 agree to two decimal places, the algorithm ends after four iterations. An estimate for sqrt(20) is 4.47214.

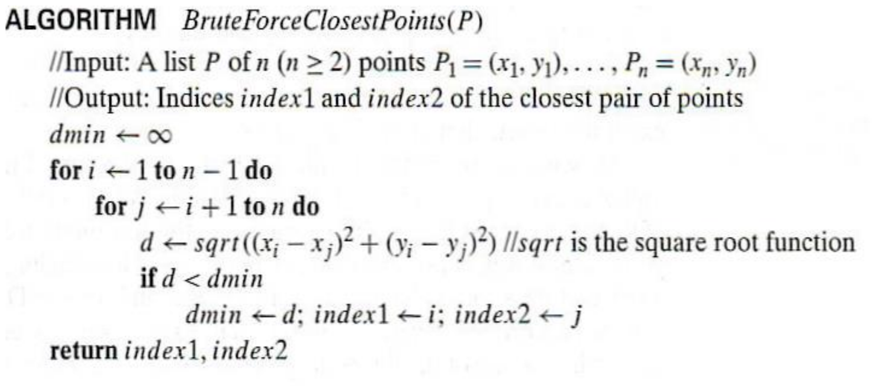
Question # 3: [CLO4:2]

Consider the following version of an important algorithm find time complexity for this algorithm

1.

```
ALGORITHM GE(A[0..n − 1, 0..n])
//Input: An n × (n + 1) matrix A[0..n − 1, 0..n] of real numbers
for i ←0 to n − 2 do
    for j ←i + 1 to n − 1 do
        for k←i to n do
            A[j, k]←A[j, k]− A[i, k] ∗ A[j, i]/ A[i, i]
```

2.



```
ALGORITHM BruteForceClosestPoints(P)
    //Input: A list P of n (n ≥ 2) points P_1 = (x_1, y_1), . . . , P_n = (x_n, y_n)
    //Output: Indices index1 and index2 of the closest pair of points
    dmin ← ∞
    for i ← 1 to n − 1 do
        for j ← i + 1 to n do
            d ← sqrt((x_i − x_j)^2 + (y_i − y_j)^2) //sqrt is the square root function
            if d < dmin
                dmin ← d; index1 ← i; index2 ← j
    return index1, index2
```