OBJECT ORIENTED PROGRAMMING

SECTION – C [WEDNESDAY OCTOBER 06, 2021: 2:00 PM – 5:00 PM]

ASSIGNMENTS – 06 (RP06) CODE: ASSIGN06

NOTES:

i) Create files with the following file naming conventions: If your roll number ends with **abc**, year of admission is 20**19** and assignment code is **Assign06** then, use the file name as follows: Assign062019abc.cpp (use appropriate extension .cpp suitably).
   For example, if the roll number ends with 127; year of admission is 2019 & the assignment code is Assign06, then the file name should be **Assign062019127.cpp**

ii) Strictly follow the file naming convention. Otherwise, it would attract a penalty up to 20%.

PROBLEM: **[Total Marks: 20]**

You should choose C++ (.cpp) to solve these problems using Constructors and 2D Arrays
Define a class and methods with the same name as suggested above for the file name.

a) [4 Marks] Define a class: `Matrix` consisting m rows and n columns

b) [4 Marks] Write a method to populate the following values: `void populateValues();`
   You may use getter and setter methods as necessary within the above method.

2	7	5	11	17	9	3	17	4	3
4	**5**	3	83	7	8	4	**3**	**7**	9
6	**6**	**11**	9	5	15	7	13	**23**	3
3	4	8	1	2	7	5	9	4	6
15	7	3	2	8	8	5	6	89	7
8	7	13	7	3	1	7	7	41	7
3	97	**7**	6	4	7	**11**	**6**	2	3
5	**2**	**5**	3	8	4	6	**5**	1	8
2	4	3	2	4	9	2	3	5	2

c) [5 Marks] There is a pattern among the highlighted numbers with white background: Let x be the minimum of three numbers. The pattern follows the property: x, x+1, x+(x+1) and these three elements are organized in 'L' shape. This 'L' shaped pattern may have been rotated either clockwise or anticlockwise once or several times. Now write a function to find and print all such patterns. For example, for the highlighted pattern: 5, 6, 11, you may print the output as follows: 5 – 6 – 11 (one per line).

`void findPatterns();`

d) [2 Marks] Write a method to find whether a given number is a prime number or not? This method should return 1, if num is a prime number and 0, otherwise.

`int isPrime(int num);`

e) [5 Marks] There is a pattern among the highlighted numbers with grey background: Let x, y, and z be three prime numbers. The pattern follows the property: x < y < z and these three elements are organized in 'L' shape. This 'L' shaped pattern may have been rotated either clockwise or anticlockwise once or several times. Now write a function to find all such patterns having prime numbers. For example, for the highlighted pattern with grey background: 2 – 5 – 7, you may print the output as follows: 2 – 5 – 7 (one per line)

`void findPatternsWithPrimes();`