OBJECT ORIENTED PROGRAMMING

[THURSDAY NOVEMBER 09, 2023: 08:45 AM – 12:00 NOON]

ASSIGNMENTS – 09 CODE: ASSIGN09

NOTES:

i) Create a main class with the following file naming conventions: *If the roll number ends with 127; year of admission is 2022 & the assignment code is Assign09, then the file name should be* **Assign092022127.java** (use .java in lowercase)

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

ASSIGNMENT PROBLEMS: package name: iiits.oop.m2023

Consider the following 12 x 12 matrix:

2	3	9	2	5	1	4	3	4	9	17	11
6	5	5	4	7	9	3	7	7	8	1	2
7	3	2	3	0	16	1	7	8	15	16	3
8	9	2	8	3	5	6	2	4	7	31	2
6	6	4	10	2	3	11	4	7	0	2	3
4	3	2	5	12	23	12	3	5	7	6	12
8	21	2	7	11	13	14	2	3	11	1	11
2	11	9	5	9	4	15	1	8	8	14	2
3	6	5	4	3	5	5	11	5	6	7	16
3	4	2	41	6	8	2	3	4	11	17	4
2	1	6	8	7	11	12	4	10	4	12	7
15	13	7	9	2	18	23	5	7	3	1	9

Use Object Oriented Concepts to solve the following problems:

a) Define a superclass: **`SquareGrid`** with suitable parameterised constructor(s) with **row size**. This will create a square matrix of order **row size x row size.** Define the following methods:
`getSize()`: This returns the size of the SquareGrid
`printValuesSG()`: Takes size as an argument and prints values of the SquareGrid

b) Define an extended class **`Matrix`** that is to be defined as a square matrix of size k (k = 3);
   i. **`getCountSM(int type)`**: This has to be a recursive method that returns the total number of square matrices satisfying the following conditions:
      a. If type = 1, the middle element of the submatrix is even.
      b. If type = 2, the middle element of the submatrix is prime.
      c. If type = 3, the sum of the elements of the submatrix is odd.
   ii. **`printMatrix()`**: This has to print the type and values of the matrix satisfying the type

Wrapper class with the file name as given in the instruction will only have the main method. So every solution to be written for the following problems should be invoked from this main method only. Apply object oriented concepts wherever possible.

c) Write a recursive method that finds the values in L - shaped structure consisting of 3 elements x, y, z in the following order: x < y < z
`void findLShapeCW()` - assume the number of arguments as required.
Remember that L can be in any orientation of the clockwise direction.

d) Write a recursive method that finds the values in L - shaped structure consisting of 3 elements x, y, z in the following order: x > y and y < z
`void findLShapeXZ()` - assume necessary arguments as required.

e) Write a recursive method that finds the values in L - shaped structure consisting of 3 elements x, y, z in the following order: x + y <= z
`void findLShapeSum()` - assume necessary arguments as required.

f) Write a recursive method that finds the values in L - shaped structure consisting of 3 elements x, y, z in the following order: x < y < z
`void findLShapeLCR()` - assume necessary arguments as required.
Remember that L can be in any orientation of the anti-clockwise direction.

g) Write a recursive method that finds the values in L - shaped structure consisting of 3 elements x, y, z where any of these two elements are prime numbers\
`void findLShapeTP()` - assume necessary arguments as required.

For problems c) - g) use appropriate data structure to store the position and the values in the L-shaped arrangement of elements and print the same using a method with Object as an argument.