ISLAMIC UNIVERSITY OF TECHNOLOGY (IUT)
ORGANISATION OF ISLAMIC COOPERATION (OIC)
DEPARTMENT OF ELECTRICAL AND ELECTRONIC ENGINEERING

Lab Quiz - **02 (Set-E)**   Summer Semester - 2025

Course Number: EEE 4416   Full Marks: 20

Course Title: Simulation Lab   Time: 35 minutes

Question – 01

Generate an (m x n) matrix containing random integers uniformly distributed within the range of -100 to 100, inclusive.

- Find the minimum of each row of the matrix.

- Find the summation of all the elements in that matrix

- Return a sorted row vector with duplicates removed.

Sample Test Case:

- Input: m =3, n = 3

- Output:

    i.   randomMatrix =   [85   81
                                -14   96
                                -63   -12]

    ii.   [81; -14; -63]

    iii.   173

    iv.   Sorted_out = [-63   -14   -12   81   85   96]

Question – 02

An **Evil number** is a non-negative integer that has an **even number of 1s** in its binary representation.

Write a function called **'evil_num'** that takes an integer as input. If the input is indeed an evil number, the function should return 'evil'. If not, the function should return 'odious'.

- You can use 'dec2bin' function for decimal to binary conversion.

Test case – 01

- Input: 9
- Output: 'evil'

Test case – 02

- Input: 27
- Output: 'evil'

Test case – 03

- Input: 97
- Output: 'odious'

Test case – 04

- Input: 9778
- Output: 'evil'

Test case – 05

- Input: 9779
- Output: 'odious'

Test case – 06

- Input: 4
- Output: 'odious'

Question – 03

A **Pythagorean prime** is a prime number of the form $4n + 1$. They can be written as the sum of squares of two integers - $p = a^2 + b^2$. These two equations are logically equivalent.

For example,

- o 5 is a Pythagorean prime since –
  - 5 = 4*1 + 1
  - $5 = 1^2 + 2^2$
- o 29 is a Pythagorean prime since –
  - 29 = 4*7 + 1
  - $29 = 2^2 + 5^2$
- o 21 is not a Pythagorean prime since –
  - 21 = 4*5 + 1
  - 21 is not a prime number.

The first few Pythagorean primes are - 5, 13, 17, 29, 37, 41, 53, 61, 73, 89, 97, 101, 109, 113, … … .

Write a function called '**Pythagorean_prime**' that takes an integer k as input and returns the k-th Pythagorean prime.

Test case – 01
- Input: 5
- Output: 37

Test case – 02
- Input: 12
- Output: 101

Test case – 03
- Input: 25
- Output: 257

Test case – 04
- Input: 625
- Output: 10313

Test case – 05
- Input: 15625
- Output: 367229

Question – 4

Write a function named **'parenthesis_check'** that checks if the input string has balanced parentheses.

Test Case 1:

Input: '(a+b)*(c-d)'

Output: 1

Test Case 2:

Input: '(a+b)*(c-d'

Output: 0

Test case – 01

- Input: (a+b)*(c-d)'
- Output: 1

Test case – 02

- Input: '(a+b)*(c-d'
- Output: 0

Test case – 03

- Input: '(((how you doing)?)'
- Output: 0

Test case – 04

- Input: '((()))'
- Output: 1

Test case – 05*

- Input: ')))))(((('
- Output: 0