National University of Computer & Emerging Sciences

Lab # 3

For

Programming Fundamentals - Lab

Instructor	Mughees Ismail
Semester	Fall 2023

FAST School of Computing

Instructions:

1. Attempt all your questions on a paper, once done scan it with your cell phone and upload the PDF to the portal at Google Classroom.
2. Plagiarism is strictly prohibited.
3. Late submissions are not allowed.
4. This is a pair-programming task, only one person is to submit the solutions.
5. Write your roll numbers, lab#, section and date on the top right corner of the page.
6. You are encouraged to discuss the problem and potential solutions with your partner.
7. Write the inputs and outputs of the program before beginning the pseudocode.

Protip: Getting the requirements right for a software project is absolutely vital. It's like setting the blueprint for a building before construction begins. Without accurate requirements, efforts can be wasted building the wrong features, leading to financial loss and time wastage. Properly defined requirements ensure that the development team understands what needs to be built, reduces rework, and ultimately leads to a successful and efficient software outcome.

Scenarios:

For all the scenarios given below you are to think of solutions and write them down in the form of a pseudocode. Please follow the syntax taught in the class.

Question#1

Write a pseudocode to display the first 3 natural numbers.

Question#2

Write a pseudocode to display the first 300 natural numbers.

Question#3

Write a pseudocode to display the first 5 even numbers.

Question#4

Write a pseudocode to display the first 500 odd numbers.

Question#5

Write a pseudocode to display the first ‘n’ even numbers, where ‘n’ is a positive integer given by the user.

Question#6

Write a pseudocode to display the ‘n’ even numbers after the number ‘start’, where ‘n’ and ‘start’ are positive integers given by the user.

For example: if the user enters n=5, start=21

Output: 22, 24, 26, 28, 30 (the next 5 even numbers **after** the number 21)

Question#6

Write a pseudocode to check if a give number is a prime number or not.

Question#7

Write a program that prompts the user for a starting and ending number, then prints all the numbers in that range.

Question#8

Create a program that calculates the sum of all the numbers from 1 to a given positive integer n.

Question#9

Write a program that takes an integer n and prints the multiplication table up to 10 for that number.

Question#10

Implement a program that computes the factorial of a given positive integer n.

Question#11

Write a program that calculates the result of raising a number base to an exponent exp.

Question#12

Create a program that calculates the sum of the squares of all the numbers from 1 to n.

Note: Use of logical operators (AND, OR) is not allowed In the following questions.

Question#13

Write a program that takes coordinates (x, y) as input and determines which quadrant they lie in.

Question#14

Implement a program that takes three numbers as input and prints them in ascending order.

Question#15

Create a program that takes the lengths of three sides of a triangle and determines whether it's equilateral, isosceles, or scalene.

Question#16

Implement a program that simulates a traffic light. It should prompt the user for the current color and then display the next color.

Question#17

Write a program that counts and prints the number of digits in a positive integer.

Question#18

Write a program that takes an integers n and m and prints the multiplication table of ‘n’ up to ‘m’ for that number.

Note: for this question you cannot use multiplication * in any statement.