Assignment 1: If-Else with Logical Operators

Write a program that checks whether a given year is a leap year. A year is a leap year if:

- It is divisible by 4,
- But not divisible by 100,
- Unless it is also divisible by 400.
  The program should handle invalid inputs gracefully.

Assignment 2: Nested If-Else with Range Checking

Write a program that calculates income tax based on the following slab:

- Income <= ₹2,50,000: No tax
- ₹2,50,001 - ₹5,00,000: 5%
- ₹5,00,001 - ₹10,00,000: 20%
- Above ₹10,00,000: 30%
  The program should accept an annual income as input and output the tax amount.

Assignment 3: Switch Statement with Functions

Create a simple calculator using a switch statement. The program should ask the user to input two numbers and a choice of operation (+, -, *, /, %). Implement each operation in a separate function and call the appropriate function based on the user's input.

Assignment 4: For Loop with Advanced Patterns

Write a program that generates the following pyramid pattern for a given number of rows n:

markdown

Copy code

```
    1
   1 2
  1 2 3
 1 2 3 4
1 2 3 4 5
```

Use nested loops and appropriate spacing for alignment.

Assignment 5: While Loop with Prime Numbers

Write a program that accepts a number n and prints all prime numbers less than or equal to n. Use a while loop and a function to check if a number is prime.

Assignment 6: Do-While with Number Reversal

Write a program that repeatedly asks the user to input a positive integer and then reverses the digits of the number. For example, input 1234 should output 4321. Stop the program when the user enters 0.

Assignment 7: Break and Continue with Nested Loops

Write a program to find and print all pairs (x, y) such that:

- x^2 + y^2 = n
  where n is an input number, and both x and y range from 0 to sqrt(n). Use break and continue statements to optimize the solution.

Assignment 8: Pattern Generation with Conditional Statements

Write a program to print a checkerboard pattern of size n x n:

# # # #

# # # #

# # # #

# # # #

Use nested loops and an if-else condition to determine whether to print # or a space.

Assignment 9: Menu-Driven Program with Loops

Create a menu-driven program that allows the user to perform the following operations repeatedly:

1. Find the factorial of a number.
2. Find the GCD of two numbers.
3. Find the LCM of two numbers.

4. Exit the program.
   Implement the menu using a loop and appropriate control structures.

Assignment 10: Nested Control Statements with Arrays

Write a program to find all subsets of an array whose sum equals a given target value.

- Input: Array of integers and a target sum.
- Output: All subsets (in any order) that sum up to the target.
  For example:
  Input: [2, 3, 5], Target = 5
  Output: [2, 3] and [5]