1. **Body Mass Index (BMI) Calculator and Category:** Write a program that takes two float inputs, weight (in kilograms) and height (in feet and inches), and calculates the BMI (Body Mass Index) of a person. The program should then output the BMI value and the BMI category according to the following classifications:
    - Underweight: BMI < 18.5
    - Normal weight: 18.5 <= BMI < 24.9
    - Overweight: 24.9 <= BMI < 29.9
    - Obesity (Class 1): 29.9 <= BMI < 34.9
    - Obesity (Class 2): 34.9 <= BMI < 39.9
    - Extreme obesity (Class 3): BMI >= 39.9

    BMI Formula: weight (kg) / [height (m)]2

    1 meter = 39.37 inches.

2. **Calculate the distance between two points in a 2D plane:** Write a program that takes four float inputs (x1, y1, x2, y2) representing the coordinates of two points (P1 and P2) in a 2D plane. The program should calculate and output the Euclidean distance between these two points.

    Formula: **distance = sqrt((x2 - x1)^2 + (y2 - y1)^2)**

    Note: You will need to use the **sqrt()** function from the **math.h** library.

3. **Roman Numeral Converter:** Write a program that takes an integer input (between 1 and 3999) and converts it to its Roman numeral representation. The program should output the Roman numeral as a string.

    Note: You will need to use a series of conditional statements to break down the input number into components that correspond to Roman numeral symbols

    I = 1 , V = 5, X = 10 L = 50 , C = 100, D = 500, M = 1000

4. **Coordinate System Conversion - Cartesian to Polar:**

    Write a program that takes two float inputs (x, y) representing the Cartesian coordinates of a point in a 2D plane. The program should convert these coordinates to polar coordinates (r, θ) and output the radius (r) and angle (θ) in degrees.

    Formulas:

    a. Radius: **r = sqrt(x^2 + y^2)**
    b. Angle: **θ = atan2(y, x) * (180 / PI)**

    Note: You will need to use the **sqrt()**, **atan2()**, and other functions from the **math.h** library. Also, be sure to handle different quadrants and edge cases properly.