Mensuration Calculator

Submitted by: M.Namrata
Singha
Enrollment ID:
ADTU/1/2024-28/BCSS/048
BTech.CSE 2nd Semester
ASSAM DOWN TOWN
UNIVERSITY Certificate
Course in C Programming

1.Introduction

This project report documents the development of a C program that calculates the area and perimeter or circumference of basic geometric shapes including Circle, Rectangle, and Triangle. The project aims to reinforce understanding of user-defined functions, input validation, and control structures in the C programming language.

2. Objectives and Tools Used

The main objectives of this project are: - To
implement functions for shape-based calculations.
- To handle user input with validation. - To improve
code readability and documentation.

Tools Used: - Language: C - Compiler: GCC (via
Code::Blocks or any standard C IDE) - Platform:
Windows/Linux

3. Program Code Overview

The program is modular, with separate functions for calculating the area and perimeter or circumference for each shape. The main menu allows the user to choose the desired shape and input its dimensions. The code also includes input validation to handle invalid entries.

Example Function
Declaration:

```
float area_circle(float radius); float
perimeter_circle(float radius); float
area_rectangle(float length, float width);
float area_triangle(float base, float
height);
```

4. Sample Code Snippet

```
float area_circle(float
radius) { return 3.14 *
radius * radius; }
float perimeter_circle(float
radius) {
    return 2 * 3.14 *
}   radius;


void main() {  int choice; float radius;
printf("Choose shape: 1.Circle 2.Rectangle
3.Triangle\n");  scanf("%d", &choice);



    switch(choice)
    {   case 1:
            printf("Enter radius: "); scanf("%f",
            &radius); printf("Area: %.2f\n",
            area_circle(radius));
```

```
            printf("Circumference: %.2f\n",
            perimeter_circle(radius)); break;
    }
}
```

5. Conclusion

The Mensuration Calculator project provided practical experience in using C for real-world applications. It involved modular coding, user interaction, and control flow management. This project successfully demonstrates how basic mathematical functions can be implemented in a user-friendly C program.

Appendix: Complete Source Code

```
#include <stdio.h>
#define PI 3.14

float area_circle(float radius) {
    return PI * radius * radius;
}

float perimeter_circle(float radius) {
    return 2 * PI * radius;
}

float area_rectangle(float length, float width) {
    return length * width;
}

float perimeter_rectangle(float length, float width) {
    return 2 * (length + width);
}

float area_triangle(float base, float height) {
    return 0.5 * base * height;
}

void show_menu() {
    printf("\n--- Mensuration Calculator ---\n");
    printf("1. Circle\n2. Rectangle\n3. Triangle\n4. Exit\n");
    printf("Enter your choice: ");
}

int main() {
    int choice;
    float radius, length, width, base, height;

    do {
        show_menu();
        scanf("%d", &choice);
        switch (choice) {
            case 1:
                printf("Enter radius of the circle: ");
                scanf("%f", &radius);
                printf("Area: %.2f\n", area_circle(radius));
                printf("Circumference: %.2f\n", perimeter_circle(radius));
                break;
            case 2:
                printf("Enter length and width of the rectangle: ");
                scanf("%f %f", &length, &width);
                printf("Area: %.2f\n", area_rectangle(length, width));
```

```
                printf("Perimeter: %.2f\n", perimeter_rectangle(length, width));
                break;
            case 3:
                printf("Enter base and height of the triangle: ");
                scanf("%f %f", &base, &height);
                printf("Area: %.2f\n", area_triangle(base, height));
                break;
            case 4:
                printf("Exiting the program.\n");
                break;
            default:
                printf("Invalid choice. Please try again.\n");
        }
    } while (choice != 4);

    return 0;
}
```