Project Title: Student Marks & Grade Management System

Submitted By:

Chetan Prajapat
BCA – 2nd Semester
SAGE University, Indore
(Powered by Sunstone)

Introduction

C programming is a powerful and widely-used language that forms the base of many modern programming languages. It is known for its speed, efficiency, and control over system resources. C is often used in system programming, game development, and embedded systems. It provides a strong foundation in logic building and problem solving, making it ideal for beginners in the programming world.

Objective of the Project

The main objective of this project is to create a simple and user-friendly program in C that allows a student to enter their marks in five different subjects. The program then calculates the total marks, average marks, and assigns a grade based on the average. The project is built using functions to ensure modularity and easy understanding of each operation.

Project Title Introduction

Student Marks & Grade Management System

This project is a simple and practical implementation of a C program that helps manage student performance. It takes input of marks for 5 different subjects, validates them, and then calculates the total marks, average marks, and finally assigns a grade (A, B, C, D or F) based on the average.

The project uses a function-based approach to make the code clean, organized, and easy to understand. It is a real-life example of how programming can be used to automate repetitive academic tasks like result processing.

What I've Built in This Project

1. Marks Input System

I created a function called inputMarks() which asks the user to enter marks for 5 subjects. I used a do-while loop to make sure the input is between 0 and 100. If someone enters a wrong value (like -5 or 105), the program asks again. This keeps the inputs valid.

2. Total Marks Calculation

For this part, I made a function called calculateTotal(). It uses a for loop to add all subject marks together and returns the total. The logic is simple and easy to understand.

3. Calculating Average

I used a function named calculateAverage() that takes the total marks and divides it by 5 (number of subjects). It returns the average in float format, so it can show decimal values (like 76.40).

4. Grade Assignment

To assign the grade, I wrote a function called calculateGrade(). It uses an if-else ladder to check the average and assign the grade:

- A for 90 and above

- B for 75 to 89
- C for 60 to 74
- D for 40 to 59
- F for below 40

5. Final Output Display

At the end of the program, it prints the total marks, average, and grade in a clear format. The output is easy to read and user-friendly, so the result looks clean and professional.

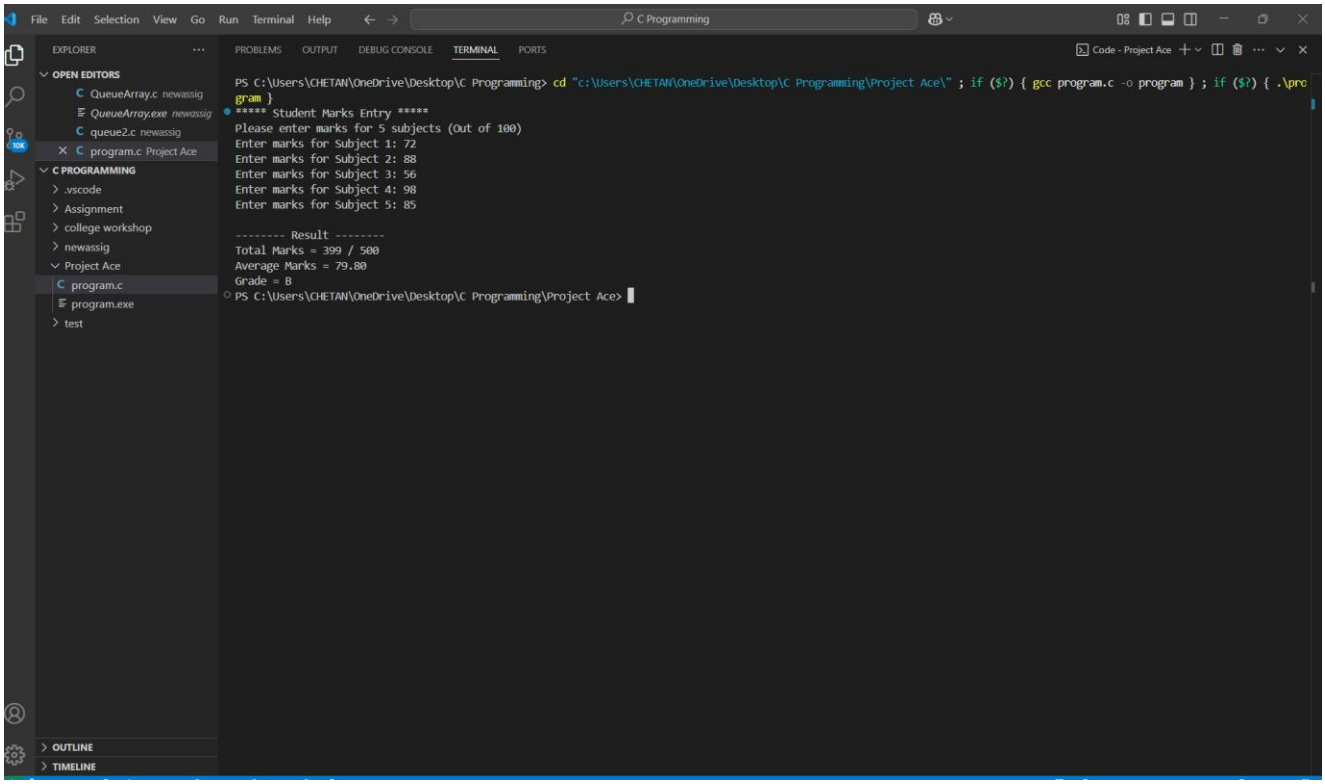
My Program

```
#include <stdio.h>

// Function declarations
void inputMarks(int m[], int n);
int findTotal(int m[], int n);
float findAverage(int total, int n);
char getGrade(float avg);

int main() {
   int marks[5];  // Array to store marks of 5 subjects
   int total;
   float average;
   char grade;

   printf("***** Student Marks Entry *****\n");
   printf("Please enter marks for 5 subjects (Out of 100)\n");

   // Function to take input
```

```
    inputMarks(marks, 5);

    // Calculating total, average and grade
    total = findTotal(marks, 5);
    average = findAverage(total, 5);
    grade = getGrade(average);

    // Display result
    printf("\n-------- Result --------\n");
    printf("Total Marks = %d / 500\n", total);
    printf("Average Marks = %.2f\n", average);
    printf("Grade = %c\n", grade);

    return 0;
}

// Function to take input with validation
void inputMarks(int m[], int n) {
    for (int i = 0; i < n; i++) {
        do {
            printf("Enter marks for Subject %d: ", i + 1);
            scanf("%d", &m[i]);

            if (m[i] < 0 || m[i] > 100) {
                printf("!! Invalid marks. Please enter between 0 to 100
only.\n");
            }
        } while (m[i] < 0 || m[i] > 100);
    }
}

// Function to calculate total marks
int findTotal(int m[], int n) {
```

```
    int total = 0;
    for (int i = 0; i < n; i++) {
        total = total + m[i];
    }
    return total;
}

// Function to calculate average
float findAverage(int total, int n) {
    return (float)total / n;
}

// Function to calculate grade
char getGrade(float avg) {
    if (avg >= 90) {
        return 'A';
    } else if (avg >= 75) {
        return 'B';
    } else if (avg >= 60) {
        return 'C';
    } else if (avg >= 40) {
        return 'D';
    } else {
        return 'F';
    }
}
```

Output




```
PS C:\Users\CHETAN\OneDrive\Desktop\C Programming\Project Ace> cd "c:\Users\CHETAN\OneDrive\Desktop\C Programming\Project Ace\" ; if ($?) { .\program }
***** Student Marks Entry *****
Please enter marks for 5 subjects (Out of 100)
Enter marks for Subject 1: 44
Enter marks for Subject 2: 0
Enter marks for Subject 3: 65
Enter marks for Subject 4: 34
Enter marks for Subject 5: 0

-------- Result --------
Total Marks = 143 / 500
Average Marks = 28.60
Grade = F
PS C:\Users\CHETAN\OneDrive\Desktop\C Programming\Project Ace>
```

Github Repu

https://github.com/grchetan/ student-marks-grade-system

Gmail: chetanprajapat340@gmail.com