STUDENT MARKS MANAGMENT SYSTEM IN C

Project report.

NAME- Siddharth Solanki

Course code- CSEG1032

COURSE TITLE- programming in C

1st semester

SAP ID- 590022620

ABSTRACT-

this project implements a simple student marks management system using the c programming language.
it demonstrates practical use of functions, loops, arrays, string handling, and conditional logic.
the program allows the user to enter student details such as name, roll number, and marks in multiple subjects.
it then calculates the total marks and percentage, stores the data in variables, and displays a formatted student record on the console.

the system uses a menu-driven approach, making it easy for users to navigate different operations such as entering details, calculating percentage, clearing the screen, and viewing the final report.
this project shows how basic c concepts can be combined to build a functional academic record system suitable for beginners.

Problem Definition-

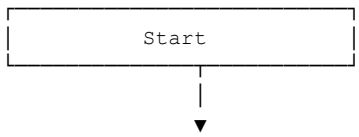
the aim of this project is to build a simple student marks management system in c that can store and process basic academic records.
the program allows the user to enter student details and marks, calculates the total and percentage, and displays the complete student report.

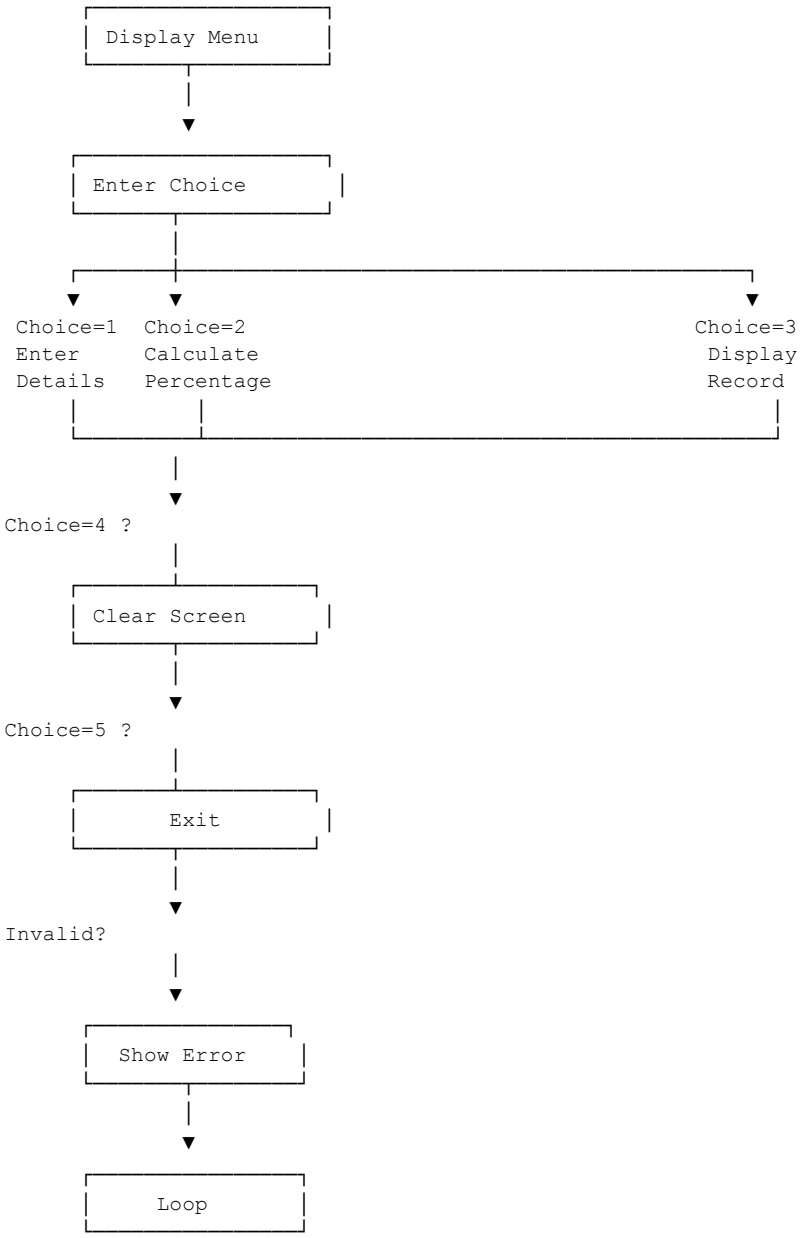
it uses a menu-driven interface with options to add details, compute percentage, view the record, clear the screen, and exit.
this system helps automate basic student result processing using fundamental c programming concepts.

Algorithm:-

1. Start
2. Initialize variables for name, roll number, marks array, subject count, and percentage
3. Display the main menu with options
4. Read the user's choice
5. If choice = 1, ask for student name, roll number, and number of subjects
6. Input marks for each subject and validate them
7. Mark that the data has been entered
8. If choice = 2, calculate total marks
9. Compute percentage using: (total / (subjects × 100)) × 100
10. If choice = 3, display the stored student details and marks
11. If choice = 4, clear the screen
12. If choice = 5, exit the program
13. If input is invalid, show error message
14. Repeat the menu until the user exits
15. end

Flowchart-





Implementation

The program is implemented in C using GCC. Arrays are used to store marks, loops handle repeated input, a separate function determines the grade, and file handling stores the report card.
The
implementation highlights modular and structured programming.

THE CODE-

```
#include <stdio.h>
```

```c
#include <string.h>

void enterStudentDetails();
void calculatePercentage();
void displayStudentRecord();
void clearScreen();

char studentName[50];
char rollNumber[20];
int marks[10];
int totalSubjects = 0;
float percentage = 0;
int dataEntered = 0;

int main() {
    int choice;

    while (1) {
        printf("\n========================================\n");
        printf("     STUDENT MARKS MANAGEMENT SYSTEM\n");
        printf("========================================\n");
        printf("1. Enter Student Details\n");
        printf("2. Calculate Percentage\n");
        printf("3. Display Student Record\n");
        printf("4. Clear Screen\n");
        printf("5. Exit\n");
        printf("----------------------------------------\n");
        printf("Enter your choice: ");
        scanf("%d", &choice);

        switch (choice) {
            case 1:
                enterStudentDetails();
                break;

            case 2:
                calculatePercentage();
                break;

            case 3:
                displayStudentRecord();
                break;

            case 4:
                clearScreen();
                break;

            case 5:
                printf("\nExiting program. Thank you!\n");
                return 0;

            default:
```

```
                printf("\nInvalid choice. Please try again.\n");
        }
    }

    return 0;
}

void enterStudentDetails() {
    int i;

    printf("\nEnter Student Name: ");
    scanf(" %[^\n]", studentName);

    printf("Enter Roll Number: ");
    scanf("%s", rollNumber);

    printf("Enter number of subjects: ");
    scanf("%d", &totalSubjects);

    if (totalSubjects <= 0 || totalSubjects > 10) {
        printf("\nInvalid subject count. Try again.\n");
        return;
    }

    for (i = 0; i < totalSubjects; i++) {
        printf("Enter marks for Subject %d (out of 100): ", i + 1);
        scanf("%d", &marks[i]);

        if (marks[i] < 0 || marks[i] > 100) {
            printf("Invalid marks! Please enter between 0 and 100.\n");
            i--;
            continue;
        }
    }

    printf("\nDetails stored successfully.\n");
    dataEntered = 1;
}

void calculatePercentage() {
    if (!dataEntered) {
        printf("\nNo data available. Enter student details first.\n");
        return;
    }

    int sum = 0, i;

    for (i = 0; i < totalSubjects; i++) {
        sum += marks[i];
    }

    percentage = (sum / (float)(totalSubjects * 100)) * 100.0;
```

```
    printf("\nPercentage calculated successfully.\n");
}

void displayStudentRecord() {
    if (!dataEntered) {
        printf("\nNo record available. Please enter student details.\n");
        return;
    }

    printf("\n=========================================\n");
    printf("              STUDENT RECORD\n");
    printf("=========================================\n");
    printf("Name           : %s\n", studentName);
    printf("Roll Number    : %s\n", rollNumber);
    printf("-----------------------------------------\n");

    int i;
    for (i = 0; i < totalSubjects; i++) {
        printf("Subject %d Marks : %d\n", i + 1, marks[i]);
    }

    printf("-----------------------------------------\n");
    printf("Percentage     : %.2f%%\n", percentage);
    printf("=========================================\n");
}

void clearScreen() {
    printf("\033[2J\033[1;1H");
    printf("Screen Cleared.\n");
```

Testing and results

```
siddharth@Siddharths-MacBook-Air-2 cprogram % gcc project.c -o project
./project

==========================================
     STUDENT MARKS MANAGEMENT SYSTEM
==========================================
1. Enter Student Details
2. Calculate Percentage
3. Display Student Record
4. Clear Screen
5. Exit
------------------------------------------
Enter your choice: 1

Enter Student Name: siddharth solanki
Enter Roll Number: 590022620
Enter number of subjects: 4
Enter marks for Subject 1 (out of 100): 70
Enter marks for Subject 2 (out of 100): 68
Enter marks for Subject 3 (out of 100): 81
Enter marks for Subject 4 (out of 100): 69

Details stored successfully.
```

```
Details stored successfully.

==========================================
     STUDENT MARKS MANAGEMENT SYSTEM
==========================================
1. Enter Student Details
2. Calculate Percentage
3. Display Student Record
4. Clear Screen
5. Exit
------------------------------------------
Enter your choice: 2

Percentage calculated successfully.

==========================================
     STUDENT MARKS MANAGEMENT SYSTEM
==========================================
1. Enter Student Details
2. Calculate Percentage
3. Display Student Record
4. Clear Screen
5. Exit
------------------------------------------
Enter your choice: 3

==========================================
```

Conclusion:-

This project demonstrates how core C programming concepts—such as functions, loops, arrays, and string handling—can be combined to build a simple Student Marks Management System.
The program allows users to enter student details, input marks for multiple subjects, calculate the total and percentage, and display a clear report.
Its menu-driven structure makes the system easy to use and understand.
Overall, the project successfully meets its goal of automating basic academic result processing using beginner-level C programming.

Future scope:-

1. Add file handling to store and retrieve student records permanently.
2. Extend the system to handle multiple students instead of one at a time.

3. Include automatic grade calculation based on percentage.

9. REFERENCES

Kernighan, B. W., & Ritchie, D. M. The C Programming Language.
Lecture Notes
Standard C Library Documentation: cppreference.com