**CIPHERBYTE TECHNOLOGIES**

**TASK 1**

STUDENT GRADE MANAGEMENT

Develop a program to manage student grades. Allow the user to input student information and multiple subject scores. Calculate and display average scores, identify the highest/lowest scores, and provide a summary of students' performance. The program will allow users, such as teachers or administrators, to input student information, including names and subject scores. It will then perform various calculations to provide valuable insights into students' performance.

**CODE:**

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#define MAX\_STUDENTS 100

#define MAX\_SUBJECTS 10

#define NAME\_LENGTH 50

#define SUBJECT\_NAME\_LENGTH 50

typedef struct {

char subject[SUBJECT\_NAME\_LENGTH];

float score;

} SubjectScore;

typedef struct {

char name[NAME\_LENGTH];

SubjectScore scores[MAX\_SUBJECTS];

int numScores;

} Student;

void addStudent(Student students[], int \*numStudents) {

if (\*numStudents >= MAX\_STUDENTS) {

printf("Maximum number of students reached.\n");

return;

}

printf("Enter the student's name: ");

scanf("%s", students[\*numStudents].name);

students[\*numStudents].numScores = 0;

(\*numStudents)++;

printf("Student added successfully.\n");

}

void addSubjectScore(Student students[], int numStudents) {

char name[NAME\_LENGTH];

printf("Enter the student's name: ");

scanf("%s", name);

for (int i = 0; i < numStudents; i++) {

if (strcmp(students[i].name, name) == 0) {

if (students[i].numScores >= MAX\_SUBJECTS) {

printf("Maximum number of subjects reached for this student.\n");

return;

}

printf("Enter the subject name: ");

scanf("%s", students[i].scores[students[i].numScores].subject);

printf("Enter the score: ");

scanf("%f", &students[i].scores[students[i].numScores].score);

students[i].numScores++;

printf("Score added successfully.\n");

return;

}

}

printf("Student not found.\n");

}

void displayStudentSummary(Student students[], int numStudents) {

char name[NAME\_LENGTH];

printf("Enter the student's name: ");

scanf("%s", name);

for (int i = 0; i < numStudents; i++) {

if (strcmp(students[i].name, name) == 0) {

printf("Student: %s\n", students[i].name);

printf("Scores:\n");

float total = 0;

float highest = -1;

float lowest = 101;

for (int j = 0; j < students[i].numScores; j++) {

printf(" %s: %.2f\n", students[i].scores[j].subject, students[i].scores[j].score);

total += students[i].scores[j].score;

if (students[i].scores[j].score > highest) {

highest = students[i].scores[j].score;

}

if (students[i].scores[j].score < lowest) {

lowest = students[i].scores[j].score;

}

}

printf("Average Score: %.2f\n", total / students[i].numScores);

printf("Highest Score: %.2f\n", highest);

printf("Lowest Score: %.2f\n", lowest);

printf("\n");

return;

}

}

printf("Student not found.\n");

}

int main() {

Student students[MAX\_STUDENTS];

int numStudents = 0;

int choice;

while (1) {

printf("1. Add a new student\n");

printf("2. Add subject score for a student\n");

printf("3. Display student summary\n");

printf("4. Exit\n");

printf("Enter your choice: ");

scanf("%d", &choice);

switch (choice) {

case 1:

addStudent(students, &numStudents);

break;

case 2:

addSubjectScore(students, numStudents);

break;

case 3:

displayStudentSummary(students, numStudents);

break;

case 4:

printf("Exiting the program.\n");

exit(0);

default:

printf("Invalid choice. Please try again.\n");

}

}

return 0;

}

**OUTPUT:**





