DEPARTMENT OF BASIC SCIENCE AND HUMANITIES
INSTITUTE OF ENGINEERING AND MANAGEMENT, KOLKATA

"<u>STUDENT MANAGEMENT SYSTEM</u>"

Submitted by:-

Name of the Student: Samadrito Modak
Enrolment Number: 12022002011033
Registration Number: 221040110562
Section: J
Class Roll Number: 13
Stream: Electrical Engineering (EE)
Subject: Programming for Problem Solving
Subject Code: ESC-103 (Pr)

Under the supervision of:-
Prof. Swarnendu Ghosh

Academic Year: 2022-26
(PROJECT REPORT SUBMITTED IN FULFILLMENT OF THE REQUIREMENTS FOR THE SECOND SEMESTER)

<u>CERTIFICATE OF RECOMMENDATION</u>

We hereby recommend that the project prepared under our supervision by **Dipto Majumder**, entitled **"<u>Student Management System</u>"** be accepted in fulfillment of the requirements for the degree of fulfillment of the second semester.

Head of the Department
IEM, Kolkata

Project Supervisor
Basic Science and Humanities

1. Introduction:

This project is assigned to me for developing a Student Management System with the help of basic C programming language.
The basic aim of the project is to create a student management system .This project is based on the concepts of arrays, and in this we can conveniently add, manage and modify the data record of students of a school or a university, like adding a new Student's record, modifying existing records, searching if a student exists or not and more. In this project, each task is distributed efficiently among different functions to improve the readability of the code and to help the reader better understand the concepts.

2. Variable Description:

The different variables used in this project are listed under:-
firstname- Array to store or take input of the first name of a Student.
Lastname - Array to store or take input of the last name of a Student.
rollNumber- To store roll number of students.
CGPA- To store the cgpa of a student.

3.Function Description:

The different functions (structures) used in this project are listed under:-
1.addStudents()- This method will let the user add a new student's record, by providing its firstName, lastName, rollNumber, CGPA and the courses in which they are enrolled and in the end we are incrementing the global variable **i** which we defined to keep track of number of students.
2.findByRollNumber() - This method will let the user to find if a student exists or not by verifying its firstName, in this we are asking the user to enter the first name of the studnet, and then are checking if any currently existing student has the matching firstName or not .
3.findRegisteredStudent() -This method will let the user to find if a student exits or not using the course ids, in this we are asking the user to enter the course id, and then are checking if any currently existing student is enrolled into that specfic course.
4.totalCount() - This method will display the total number of students into the system .
5.deleteStudent() - This method will let the user to delete a student's record, in this we are asking for the roll number of a student, and are iterating over a for loop onto the complete array database for that student, once found, we are simply updating the value at that position in the memory by its succssive position and hence removing the record .
6.updateStudent() - This method will let the user to update the records of a student using its roll number, in this we are asking the user to enter the student's roll number and then are checking if a student with that roll number exists or not, if found we are then asking what specific detail that the user wants to update, and then are updating that.

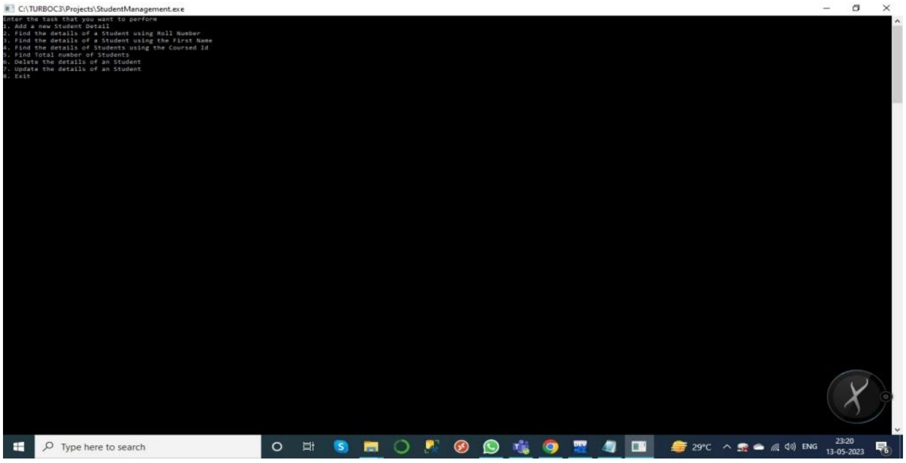
4.Program Source Code:

Student Management System.c

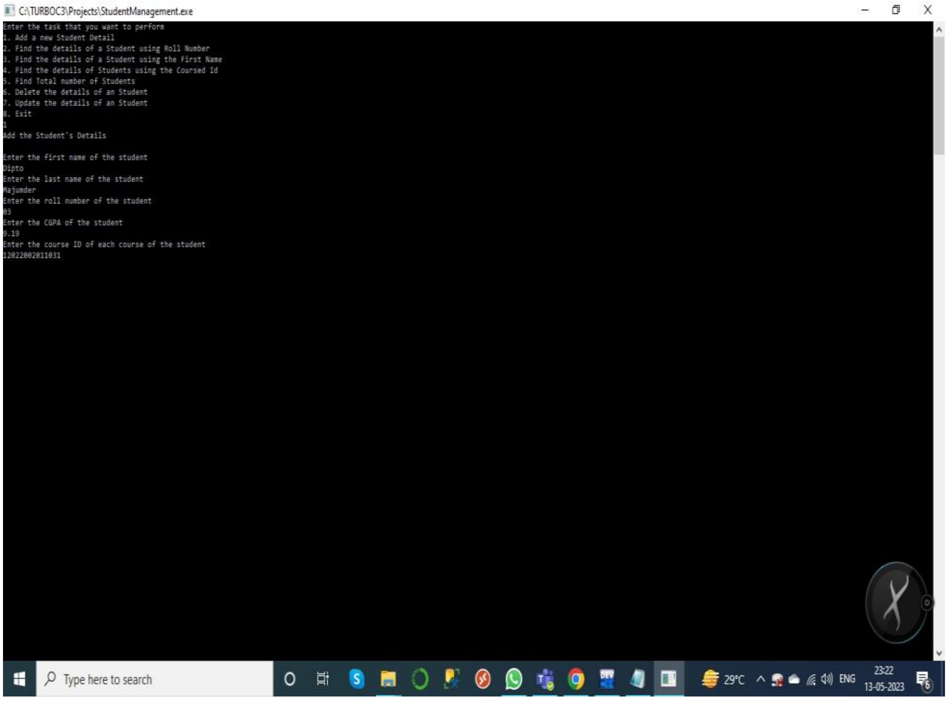
```
#include <stdio.h>
#include <math.h>
#include <string.h>
#include <stdlib.h>
void addStudent();
// To keep track of the number of students
int i = 0;

// A structure to store the student details
struct studentInfo
{
   char firstName[20];
   char lastName[20];
   int rollNumber;
   float CGPA;
   int courseId[10];
} st[500];

// Function to add student details
void addStudent()
{
   printf("Add the Student's Details\n\n");
   printf("Enter the first name of the student\n");
   scanf("%s", st[i].firstName);
   printf("Enter the last name of the student\n");
   scanf("%s", st[i].lastName);
   printf("Enter the roll number of the student\n");
   scanf("%d", &st[i].rollNumber);
   printf("Enter the CGPA of the student\n");
   scanf("%f", &st[i].CGPA);
   printf("Enter the course ID of each course of the student\n");
   for (int j = 0; j < 5; j++)
   {
      scanf("%d", &st[i].courseId[j]);
   }
   i = i + 1;
}

// Function to find the students using roll number
void findByRollNumber()
{
   int temp;
   printf("Enter the roll number of the student\n");
   scanf("%d", &temp);
   for (int j = 1; j <= i; j++)
   {
      if (temp == st[i].rollNumber)
      {
         printf("The student's details are\n");
         printf("The first name is %s\n", st[i].firstName);
         printf("The last name is %s\n", st[i].lastName);
         printf("The CGPA is %f\n", st[i].CGPA);
         for (int j = 0; j < 5; j++)
         {
            printf("The enrolled course Ids are %d\n", st[i].courseId[j]);
         }
      }
   }
}

// Function to find the student by the first name
void findByName()
{
   char temp[20];
   printf("Enter the First Name of the Student\n");
   scanf("%s", temp);
   int c = 0;

   for (int j = 1; j <= i; j++)
   {
      if (!strcmp(st[j].firstName, temp))
      {
         printf("The student's details are\n");
         printf("The first name is %s\n", st[i].firstName);
         printf("The last name is %s\n", st[i].lastName);
         printf("The CGPA is %f\n", st[i].CGPA);
         for (int j = 0; j < 5; j++)
         {
            printf("The enrolled course Ids are %d\n", st[i].courseId[j]);
         }
      }
   }
}

// Function to find students enrolled in a specific course
void findRegisteredStudent()
{
   int temp;
   printf("Enter the course ID\n");
   scanf("%d", &temp);
   int c = 0;

   for (int j = 1; j <= i; j++)
   {
      for (int d = 0; d < 5; d++)
      {
         if (temp == st[j].courseId[d])
         {
            printf("The student's details are\n");
            printf("The first name is %s\n", st[i].firstName);
            printf("The last name is %s\n", st[i].lastName);
            printf("The CGPA is %f\n", st[i].CGPA);
            for (int j = 0; j < 5; j++)
            {
               printf("The enrolled course Ids are %d\n", st[i].courseId[j]);
            }
         }
      }
   }
}

// Function to print the total number of students
void totalCount()
{
   printf("The total number of students currently in the system is %d\n", i);
}
// Function to delete a Student
void deleteStudent()
{
   int temp;
   printf("Enter the roll number of the student\n");
   scanf("%d", &temp);
   for (int j = 1; j <= i; j++)
   {
      if (temp == st[j].rollNumber)
      {
         for (int k = j; k < 499; k++)
         {
            st[k] = st[k + 1];
         }
         i--;
      }
   }
   printf("The entered student's record deleted successfully");
}

// Function to update a student's data
int main()
{
   int taskToPerform;
   while (1)
   {
      printf("Enter the task that you want to perform\n");
      printf("1. Add a new Student Detail\n");
      printf("2. Find the details of a Student using Roll Number\n");
      printf("3. Find the details of a Student using the First Name\n");
      printf("4. Find the details of Students using the Coursed Id\n");
      printf("5. Find Total number of Students\n");
      printf("6. Delete the details of an Student\n");
      printf("7. Update the details of an Student\n");
      printf("8. Exit\n");
      scanf("%d", &taskToPerform);

      switch (taskToPerform)
      {
      case 1:
         addStudent();
         break;
      case 2:
         findByRollNumber();
         break;
      case 3:
         findByName();
         break;
      case 4:
         findRegisteredStudent();
         break;
      case 5:
         totalCount();
         break;
      case 6:
         deleteStudent();
         break;
      case 7:
         updateStudent();
         break;
      case 8:
         exit(0);
         break;
      default:
         exit(0);
      }
   }

   return 0;
}
```

```
Outputs:
```

Sample outputs to demonstrate the functionalities in programs arelisted below.



```
Enter the task that you want to perform
1. Add a new Student Detail
2. Find the details of a Student using Roll Number
3. Find the details of a Student using the First Name
4. Find the details of Students using the Coursed Id
5. Find Total number of Students
6. Delete the details of an Student
7. Update the details of an Student
8. Exit
```



```
Enter the task that you want to perform
1. Add a new Student Detail
2. Find the details of a Student using Roll Number
3. Find the details of a Student using the First Name
4. Find the details of Students using the Coursed Id
5. Find Total number of Students
6. Delete the details of an Student
7. Update the details of an Student
8. Exit
1
Add the Student's Details

Enter the first name of the student
Dipto
Enter the last name of the student
Majumder
Enter the roll number of the student
03
Enter the CGPA of the student
9.19
Enter the course ID of each course of the student
12022002011031
```

The Task that you want to perform
1. Add the Student Details
2. Find the Student Details by Roll Number
3. Find the Student Details by First Name
4. Find the Student Details by Course Id
5. Find the Total number of Students
6. Delete the Students Details by Roll Number
7. Update the Students Details by Roll Number
8. TO Exit Enter your choice to find the task
1
Add the Students Details ———————————-
Enter the first name of student

```
Dipto
Enter the last name of student
Majumder
 Enter the Roll Number
3
Enter the CGPA you obtained
 9.19
 Enter the course ID of each course 1 2 3 4 5
The Task that you want to perform
1. Add the Student Details
2. Find the Student Details by Roll Number
3. Find the Student Details by First Name
4. Find the Student Details by Course Id
5. Find the Total number of Students
6. Delete the Students Details by Roll Number
7. Update the Students Details by Roll Number
8. TO Exit Enter your choice to find the task
2
Enter the Roll Number of the student
3
The Students Details are
The First name is Dipto
The Last name is Majumder
The CGPA is 9.19
Enter the course ID of each course The course ID are 1 The course ID are 2 The course ID are 3
The course ID are 4 The course ID are 5
The Task that you want to perform
1. Add the Student Details
2. Find the Student Details by Roll Number
3. Find the Student Details by First Name
4. Find the Student Details by Course Id
5. Find the Total number of Students
6. Delete the Students Details by Roll Number
7. Update the Students Details by Roll Number
8. TO Exit Enter your choice to find the task
3
Enter the First Name of the student
Dipto
The Students Details are
The First name is Dipto
The Last name is Majumder
The Roll Number is 3
The CGPA is 9.19
Enter the course ID of each course The course ID are 1 The course ID are 2 The course ID are 3
The course ID are 4 The course ID are 5
The Task that you want to perform
1. Add the Student Details
2. Find the Student Details by Roll Number
3. Find the Student Details by First Name
4. Find the Student Details by Course Id
5. Find the Total number of Students
6. Delete the Students Details by Roll Number
7. Update the Students Details by Roll Number
 8. TO Exit Enter your choice to find the task
4
Enter the course ID
1
The Students Details are
The First name is Dipto
The Last name is Majumder
The Roll Number is 3
The CGPA is 9.19
The Task that you want to perform
1. Add the Student Details
 2. Find the Student Details by Roll Number
3. Find the Student Details by First Name
4. Find the Student Details by Course Id
5. Find the Total number of Students
6. Delete the Students Details by Roll Number
7. Update the Students Details by Roll Number
8. TO Exit Enter your choice to find the task
5
The total number of Student is 1 you can have a max of 100 students you can have 99 more
students
The Task that you want to perform
1. Add the Student Details
2. Find the Student Details by Roll Number
3. Find the Student Details by First Name
4. Find the Student Details by Course Id
5. Find the Total number of Students
6. Delete the Students Details by Roll Number
7. Update the Students Details by Roll Number
8. TO Exit Enter your choice to find the task
6
Enter the Roll Number which you want to delete
3
The Roll Number is removed Successfully
The Task that you want to perform
1. Add the Student Details
2. Find the Student Details by Roll Number
3. Find the Student Details by First Name
4. Find the Student Details by Course Id
5. Find the Total number of Students
6. Delete the Students Details by Roll Number
7. Update the Students Details by Roll Number
8. TO Exit Enter your choice to find the task
2
Enter the Roll Number of the student
1
The course ID are 0 The course ID are 0 The course ID are 0 The course ID are 0 The course ID
are 0
The Task that you want to perform
1. Add the Student Details
2. Find the Student Details by Roll Number
3. Find the Student Details by First Name
4. Find the Student Details by Course Id
5. Find the Total number of Students
6. Delete the Students Details by Roll Number
7. Update the Students Details by Roll Number
 8. TO Exit Enter your choice to find the task
 8
```

THANK YOU!!