**STUDENT MANAGEMENT SYSTEM**

# Submitted by

**Name of the Students:** DEBDATTA BASAK

**Enrolment Number:** 12022002017078

**Section:** F

**Class Roll Number:** 83

**Stream:** CSE IOT

**Subject:** Programming for Problem Solving with C

**Subject Code:** ESC103(Pr.)

**Department:** Basic Science and Humanities

Under the supervision of

<Name of the Teachers>

**Academic Year: 2022-26**

PROJECT REPORT SUBMITTED IN PARTIAL FULFILLMENT OF THE

## REQUIREMENTS FOR THE FIRST SEMESTER



**DEPARTMENT OF BASIC SCIENCE AND HUMANITITES**

**INSTITUTE OF ENGINEERING AND MANAGEMENT, KOLKATA**



# CERTIFICATE OF RECOMMENDATION

We hereby recommend that the project prepared under our supervision by **Name of the Student,** entitled <TITLE OF THE PROJECT REPORT> be accepted in partial fulfillment of the requirements for the degree of partial fulfillment of the first semester.

Prof. Prabir kumar Prof Swarnendu Ghsoh

Head of the Department Project Supervisor

Basic Sciences and Humanities

IEM,kolkata

# Introduction

The project of 2nd semester includes the use of c programming language for creation of a complex piece of code that has different modules/sections that divides the complex into smaller and easier parts.

# 

## Objective

The objective of the program is to design a program that forms student management system.

**1.2 Organization of the Project**

1) void addStudent() - To add student details

2)void fifByRollNumber – To enter the roll number of the student

To enter the name, cgpa and details of students

3) void tiotalstudent()- to find the total number of student

4) void deleteStudent()- To delete student

5) void updateStudent()

Database Descriptions

Describe the different databases that are used in the project

# Programs

#include <stdio.h>

#include <math.h>

#include <string.h>

#include <stdlib.h>

/ 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 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

void updateStudent()

{

   int temp;

   printf("Enter the roll number of the student\n");

   scanf("%d", &temp);

   for (int j = 0; j < i; j++)

   {

      if (temp == st[j].rollNumber)

      {

         printf("1. First Name\n"

                "2. Last Name\n"

                "3. Roll Number\n"

                "4. CGPA\n"

                "5. Courses\n");

         int c;

         scanf("%d", &c);

         switch(c) {

            case 1:

               printf("Enter the updated first name : ");

               scanf("%s", st[j].firstName);

               break;

            case 2:

               printf("Enter the updated last name : ");

               scanf("%s", st[j].lastName);

               break;

            case 3:

               printf("Enter the updated roll number : ");

               scanf("%d", st[j].rollNumber);

               break;

            case 4:

               printf("Enter the updated CGPA : ");

               scanf("%f", st[j].CGPA);

               break;

            case 5:

               printf("Enter the updated list of courses: ");

               scanf("%d%d%d%d%d", st[j].courseId[0],st[j].courseId[1], st[j].courseId[2], st[j].courseId[3], st[j].courseId[5]);

               break;

         }

         printf("Records updated successfully");

      }

   }

}

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;

}