STUDENT MANAGMENT SYSTEM

Academic Year: 2021-22 ODD-SEMESTER

Department with Specialization: B-Tech Computer Science and Engineering with Specialization in Artificial Intelligence and Machine Learning.

Semester : 1

Course Code : 18CSS101J

Course Title : Programming for Problem Solving

Submitted by

ARNAV SHARMA (RA2111026010075)

MAYANK MUKHERJEEE (RA2111026010081)

PRIYANSH BHANDARI (RA2111026010087)

Under the Guidance of

Dr. KAYALVIZHI JAYAVEL

(Associate Professor, S.G)

DEPARTMENT OF COMPUTING

COLLEGE OF ENGINEERING AND TECHNOLOGY

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

KATTANKULATHUR- 603 203

JANUARY 2022

AIM

To create a Student Management System which helps in managing a student database through accessing with features such as to add record , delete record , or modify an record and it also has protection enabled encryption which makes the data of students secure.

ABSTRACT

This student management system made in c language has been developed with an encrypted login system which makes it leak proof with no risk of data abuse or loss, and it takes very little time to perform security tasks making it the most efficient system. This student management system being a mini project designed and build in c language is straightforward and clean, making it easy for users to learn, use, and navigate.

ALGORITHM

- **STEP 1**: START.

- **STEP 2**: WRITE ALL THE NECESSARY HEADER FILES.

- **STEP 3**: DECLARE FILE POINTERS:
    1. DEL
    2. TP
    3. FP

- **STEP 4**: DECLARE ALL THE FUNCTIONS USED:
    1. ADD()
    2. MODIFY()
    3. DISPLAY()
    4. INDIVIDUAL()
    5. PASSWORD()
    6. PRINT CHAR()
    7. TITLE()

- **STEP 5**: DEFINE FUNCTION GOTOXY.

- **STEP 6**: DECLARE STRUCTURES
    1. PASS WITH MEMBER PA
    2. STUD.

- **STEP 7**: DECLARE THE VARIABLES TO BE USED.

- **STEP 8**: GET THE USERNAME AND PASSWORD FROM THE USER. IF FIRST TIME LOGIN JUST PRESS ENTER. IF THE GIVEN INFORMATION IS VALID GO TO MAIN SCREEN.

- **STEP 9**: USE SWITCH CASE TO WORK ON EACH CASE.

    1. **CASE 1**: IF THE USER CHOICE IS 1, GO TO ADD() FUNCTION AND ASK FOR THE DETAILS OF STUDENTS TO BE ADDED.
    2. **CASE 2**: IF THE USER CHOICE IS 2, GO TO MODIFY() FUNCTION AND ASK FOR ROLL NUMBER TO BE MODIFIED.
    3. **CASE 3**: IF THE USER CHOICE IS 3, GO TO DISPLAY() FUNCTION AND SHOW THE RECORD FOR ALL THE STUDENTS.
    4. **CASE 4**: IF THE USER CHOICE IS 4, GO TO INDIVIDUAL() FUNCTION AND ASK FOR ROLL NUMBER TO BE DISPLAYED.
    5. **CASE 5**: IF THE USER CHOICE IS 5, GO TO FILE DEL AND ASK THE ROLL NUMBER OF STUDENT TO BE REMOVED.
    6. **CASE 6**: IF THE USER CHOICE IS 6, GO TO PASSWORD() FUNCTION AND ASK FOR A NEW PASSWORD.
    7. **CASE 7**: IF THE USER CHOICE IS 7, LOGOUT.

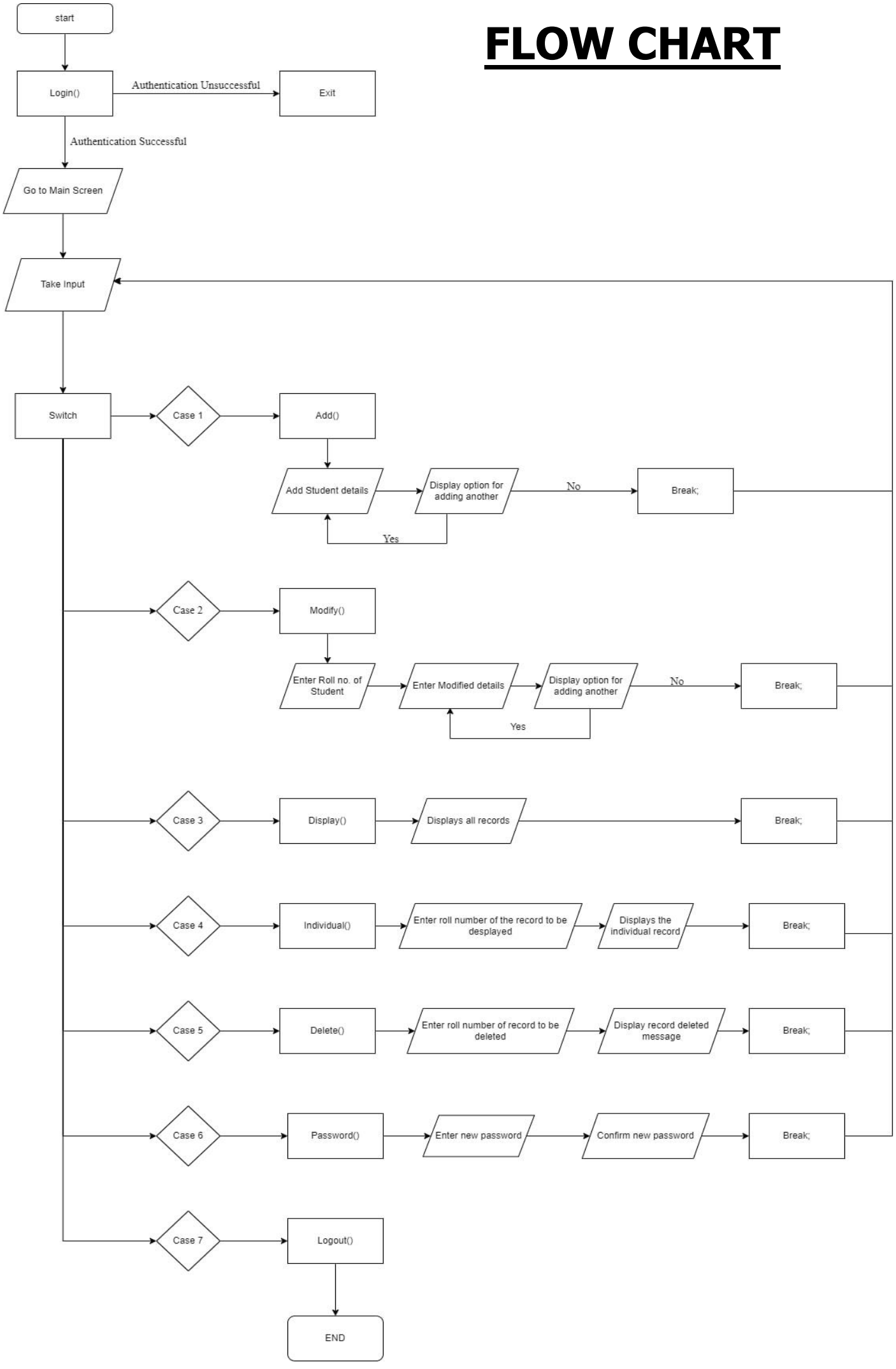
- **STEP 10**: DEFINE ALL THE FUNCTIONS.
    1. PASSWORD()
    2. PRINTCHAR()
    3. TITLE()
    4. ADD()
    5. MODIFY()

6. DISPLAY()
7. INDIVIDUAL()

❖ **STEP 11**: DEFINE FILE POINTER DEL.

❖ **STEP 12**: EXIT THE PROGRAM.

FLOW CHART


start
Login()
Authentication Unsuccessful
Exit
Authentication Successful
Go to Main Screen
Take Input
Switch
Case 1
Add()
Add Student details
Display option for adding another
No
Break;
Yes
Case 2
Modify()
Enter Roll no. of Student
Enter Modified details
Display option for adding another
No
Break;
Yes
Case 3
Display()
Displays all records
Break;
Case 4
Individual()
Enter roll number of the record to be desplayed
Displays the individual record
Break;
Case 5
Delete()
Enter roll number of record to be deleted
Display record deleted message
Break;
Case 6
Password()
Enter new password
Confirm new password
Break;
Case 7
Logout()
END

SOURCE CODE

```
#include<stdio.h>
#include<string.h>
#include<stdlib.h>
#include<math.h>
#include<windows.h>
#define Student struct Stud

void add(FILE * fp);
void modify(FILE * fp);
void display(FILE * fp);
void Individual(FILE *fp);
void password();
FILE * del(FILE * fp);
void printChar(char ch,int n);
void title();
FILE *tp;

void gotoxy(int x,int y)
{
        COORD CRD;
   CRD.X = x;
   CRD.Y = y;
   SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE),CRD);
}

struct pass
{
        char pass[25];
}pa;

struct Stud
{
   char name[100];
   char dept[50];
   int roll;
   float sgpa[12];
   float cgpa;
};


int main()
{
   int ch,id,k,i;
   char c,pas[50];
   SetConsoleTitle("Student Management System | DIU");
   FILE * fp;
   Student s;
   int option;
   char another;

   if((fp=fopen("db.txt","rb+"))==NULL)
```

```
{
   if((fp=fopen("db.txt","wb+"))==NULL)
   {
      printf("Can't create or open Database.");
      return 0;
   }
}
system("color 9f");
        gotoxy(42,8);
        printf("LOGIN(If 1st login press ENTER)");
        gotoxy(42,10);
        printf("_____");
        gotoxy(42,11);
        printf("|\tEnter password :           |");
        gotoxy(42,12);
        printf("|_____|");
        //printf("\n\t\t\t\t\t");
        gotoxy(65,11);
        while( k<10)
        {
           pas[k]=getch();
           char s=pas[k];
           if(s==13)
                    break;
           else printf("*");
           k++;
        }
        pas[k]='\0';
        tp=fopen("F:/Password.txt","r+");
fgets(pa.pass,25,tp);
if(strcmp(pas,pa.pass)==0)
        {
                    system("cls");
                    gotoxy(10,3);
                    printf("<<<< Loading Please Wait >>>>");
                    for(i=0; i<5; i++)
   {
      printf("\t(*_*)");
      Sleep(500);
   }
   printf(" \n\n\n\n\n\t\t\t\t\t    * * * * * * * * *");
                    printf("\n\n\t\t\t\t\t    *               *");
                    printf("\n\n\t\t\t\t\t    *      Welcome     *");
                    printf("\n\n\t\t\t\t\t    *               *");
                    printf("\n\n\t\t\t\t\t    * * * * * * * * *");
                    printf("\n\n\n\n\n\t\t\t\t\tPress any key to continue..... ");
                    getch();

title();
printf("\n\n\t\t\t\t STUDENT MANAGMENT SYSTEM ");
printf("\n\n\t\t\t\t\tTEAM MAP");
```

```
printf("\n\n\t\t\t  SRM INSTITUTE OF SCIENCE AND TECHNOLOGY , KTR\n\t\t\t");
printChar('=',48);
printf("\n\n\n\t\t\t       press any key to Enter");
getch();

while(1)
{
   title();
   printf("\n\t");
   printChar('*',64);

   printf("\n\n\t\t\t\t1. Add Student");
   printf("\n\n\t\t\t\t2. Modify Student");
   printf("\n\n\t\t\t\t3. Show All Student");
   printf("\n\n\t\t\t\t4. Individual View");
   printf("\n\n\t\t\t\t5. Remove Student");
   printf("\n\n\t\t\t\t6. Change Password");
   printf("\n\n\t\t\t\t7. Logout\n\t");
   printChar('*',64);
   printf("\n\n\t\t\t\tEnter Your Option :--> ");
   scanf("%d",&option);

   switch(option)
   {
      case 1:
         add(fp);
         break;
      case 2:
         modify(fp);
         break;
      case 3:
         display(fp);
         break;
      case 4:
         Individual(fp);
         break;
      case 5:
         fp=del(fp);
         break;
      case 6:
            system("cls");
                                       system("color 5f");
                                  password();
         break;
      case 7:
         return 1;
         break;
      default:
         printf("\n\t\tNo Action Detected");
         printf("\n\t\tPress Any Key\n\n\n");
         getch();
```

```
            system("pause");
        }
    }
    }
    else
    {
        printf("Wrong Password . Get Out");
        getch();
    }
    return 1;

}


void password()
{
        char c;
        printf("\nEnter new password :");
        fflush(stdin);
        gets(pa.pass);
        printf("\nSave password (y/n) :");
        fflush(stdin);
        scanf("%c",&c);
        if(c=='y'||c=='Y')
        {
                tp=fopen("F:/Password.txt","w+");
            fwrite(&pa,sizeof(pa),1,tp);
            fclose(tp);
                printf("\n\tPassword Saved\n");
        }
        else
        {
                printf("Password not saved :\n");
                printf("Press any key to continue >>>");
                getch();
        }
}


void printChar(char ch,int n)
{
    while(n--)
    {
        putchar(ch);
    }
}

void title()
{
    system("cls");
    system("COLOR 03");
```

```
    printf("\n\n\t");
    printChar('=',19);
    printf(" Student Management System ");
    printChar('=',19);
    printf("\n");
}


//Insert at end

 void add(FILE * fp)
{
    title();

    char another='y';
    Student s;
    int i;
    float cgpa;

    fseek(fp,0,SEEK_END);
    while(another=='y'||another=='Y')
    {

        printf("\n\n\t\tEnter Full Name of Student: ");
        fflush(stdin);
        fgets(s.name,100,stdin);
        s.name[strlen(s.name)-1]='\0';

        printf("\n\n\t\tEnter Depertment Name: ");
        fflush(stdin);
        fgets(s.dept,50,stdin);
        s.dept[strlen(s.dept)-1]='\0';

        printf("\n\n\t\tEnter Roll number: ");
        scanf("%d",&s.roll);

        printf("\n\n\tEnter SGPA for 12 semesters\n");
        for(i=0,cgpa=0; i<12; i++)
        {
            scanf("%f",&s.sgpa[i]);
            cgpa+=s.sgpa[i];

        }

        cgpa/=12.0;
        s.cgpa=cgpa;

        fwrite(&s,sizeof(s),1,fp);

        printf("\n\n\t\tAdd another student?(Y/N)?");
        fflush(stdin);
```

```
        another=getchar();
    }
}


FILE * del(FILE * fp)
{
    title();

    Student s;
    int flag=0,tempRoll,siz=sizeof(s);
    FILE *ft;

    if((ft=fopen("temp.txt","wb+"))==NULL)
    {
        printf("\n\n\t\t\t\\t!!! ERROR !!!\n\t\t");
        system("pause");
        return fp;
    }

    printf("\n\n\tEnter Roll number of Student to Delete the Record");
    printf("\n\n\t\t\tRoll No. : ");
    scanf("%d",&tempRoll);

    rewind(fp);


    while((fread(&s,siz,1,fp))==1)
    {
        if(s.roll==tempRoll)
        {
            flag=1;
            printf("\n\tRecord Deleted for");
            printf("\n\n\t\t%s\n\n\t\t%s\n\n\t\t%d\n\t",s.name,s.dept,s.roll);
            continue;
        }

        fwrite(&s,siz,1,ft);
    }


    fclose(fp);
    fclose(ft);

    remove("db.txt");
    rename("temp.txt","db.txt");

    if((fp=fopen("db.txt","rb+"))==NULL)
    {
        printf("ERROR");
        return  NULL;
```

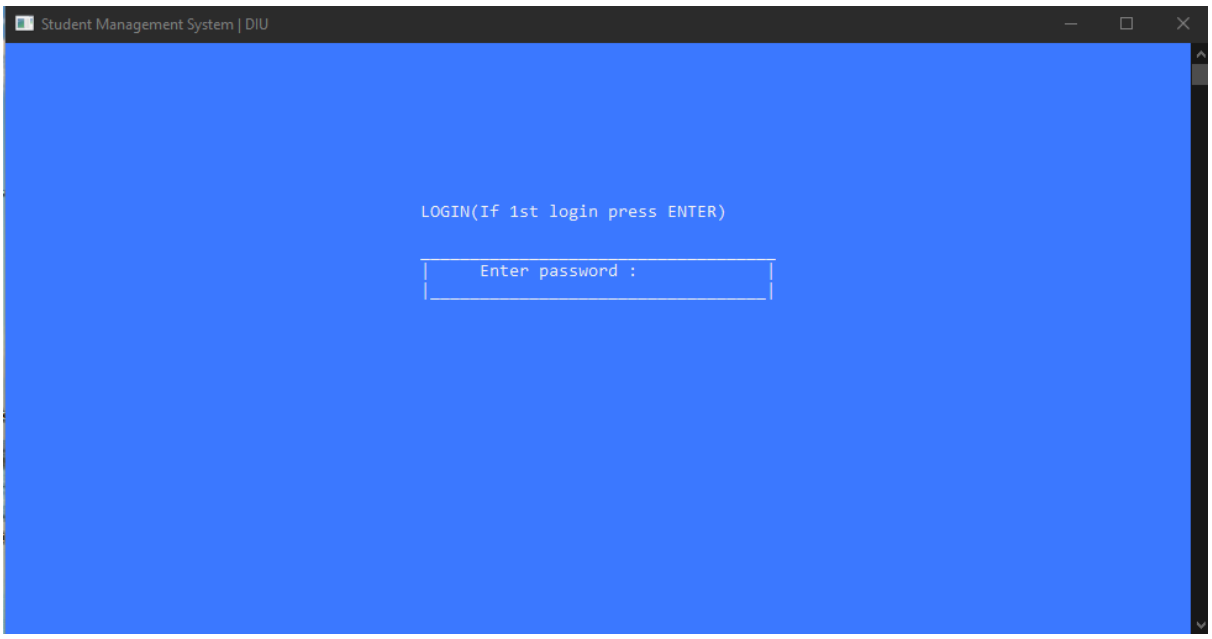
```
    }

    if(flag==0) printf("\n\n\t\tNO STUDENT FOUND WITH THE INFORMATION\n\t");

    printChar('-',65);
    printf("\n\t");
    system("pause");
    return fp;
}


void modify(FILE * fp)
{
    title();

    Student s;
    int i,flag=0,tempRoll,siz=sizeof(s);
    float cgpa;

    printf("\n\n\tEnter Roll Number of Student to MODIFY the Record : ");
    scanf("%d",&tempRoll);

    rewind(fp);

    while((fread(&s,siz,1,fp))==1)
    {
        if(s.roll==tempRoll)
        {
            flag=1;
            break;
        }
    }

    if(flag==1)
    {
        fseek(fp,-siz,SEEK_CUR);
        printf("\n\n\t\t\t\tRecord Found\n\t\t\t");
        printChar('=',38);
        printf("\n\n\t\t\tStudent Name: %s",s.name);
        printf("\n\n\t\t\tStudent Roll: %d\n\t\t\t",s.roll);
        printChar('=',38);
        printf("\n\n\t\t\tEnter New Data for the student");

        printf("\n\n\t\t\tEnter Full Name of Student: ");
        fflush(stdin);
        fgets(s.name,100,stdin);
        s.name[strlen(s.name)-1]='\0';

        printf("\n\n\t\t\tEnter Department: ");
        fflush(stdin);
        fgets(s.dept,50,stdin);
```

```
        s.dept[strlen(s.dept)-1]='\0';

        printf("\n\n\t\t\tEnter Roll number: ");
        scanf("%d",&s.roll);


        printf("\n\n\t\tEnter SGPA for 12 semesters\n");
        for(i=0,cgpa=0; i<12; i++)
        {
           scanf("%f",&s.sgpa[i]);
           cgpa+=s.sgpa[i];

        }
        cgpa=cgpa/8.0;


        fwrite(&s,sizeof(s),1,fp);
    }

    else printf("\n\n\t!!!! ERROR !!!! RECORD NOT FOUND");

    printf("\n\n\t");
    system("pause");

}

void display(FILE * fp)
{
    title();
    Student s;
    int i,siz=sizeof(s);

    rewind(fp);

    while((fread(&s,siz,1,fp))==1)
    {
        printf("\n\t\tNAME : %s",s.name);
        printf("\n\n\t\tDepertment : %s",s.dept);
        printf("\n\n\t\tROLL : %d",s.roll);
        printf("\n\n\tSGPA: ");

        for(i=0; i<12; i++)
            printf("| %.2f |",s.sgpa[i]);
        printf("\n\n\t\tCGPA : %.2f\n\t",s.cgpa);
        printChar('-',65);
    }
    printf("\n\n\n\t");
    printChar('*',65);
    printf("\n\n\t");
    system("pause");
}
```

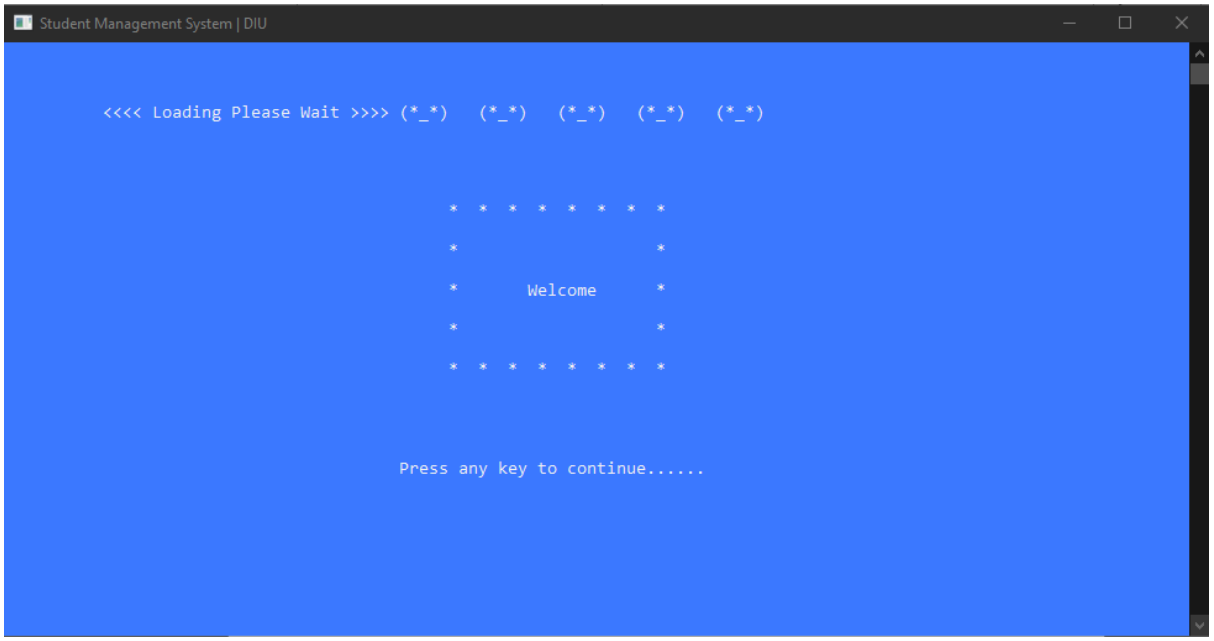
```
void Individual(FILE *fp)
{
   title();

   int tempRoll,flag,siz,i;
   Student s;
   char another='y';

   siz=sizeof(s);

   while(another=='y'||another=='Y')
   {
      printf("\n\n\tEnter Roll Number: ");
      scanf("%d",&tempRoll);

      rewind(fp);

      while((fread(&s,siz,1,fp))==1)
      {
         if(s.roll==tempRoll)
         {
            flag=1;
            break;
         }
      }

      if(flag==1)
      {
         printf("\n\t\tNAME : %s",s.name);
         printf("\n\n\t\tDepartment : %s",s.dept);
         printf("\n\n\t\tROLL : %d",s.roll);
         printf("\n\n\tSGPA: ");

         for(i=0; i<12; i++)
            printf("| %.2f |",s.sgpa[i]);
         printf("\n\n\t\tCGPA : %.2f\n\t",s.cgpa);
         printChar('-',65);

      }
      else printf("\n\n\t\t!!!! ERROR RECORD NOT FOUND !!!!");


      printf("\n\n\t\tShow another student information? (Y/N)?");
      fflush(stdin);
      another=getchar();
   }
}
```

OUTPUT

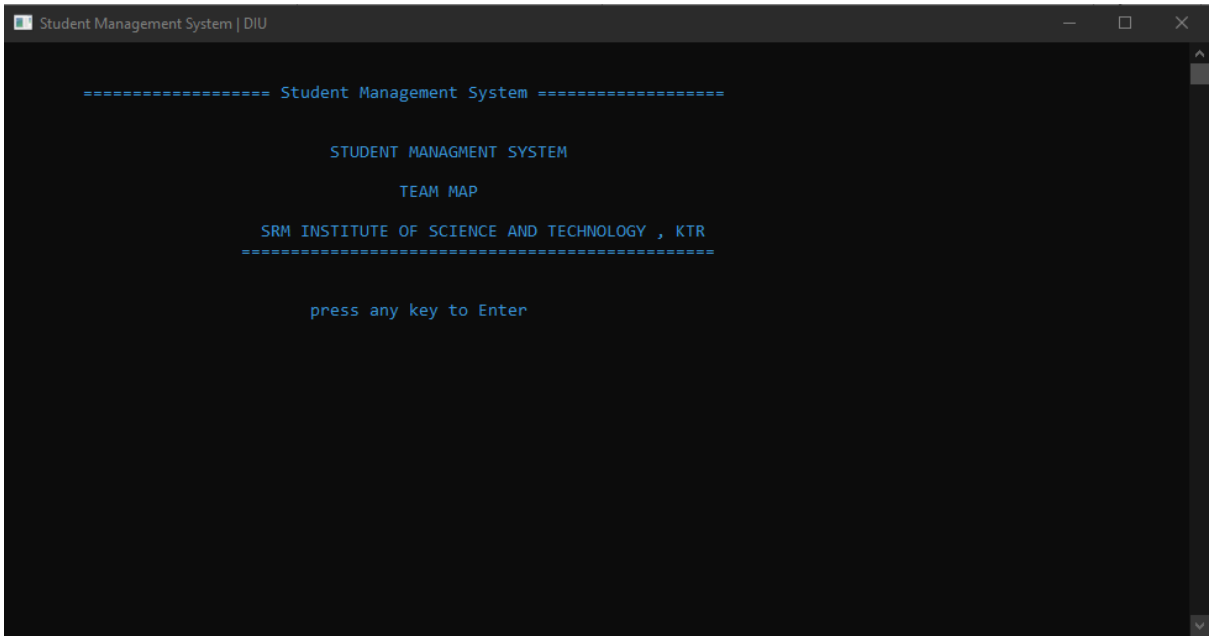
LOGIN SCREEN



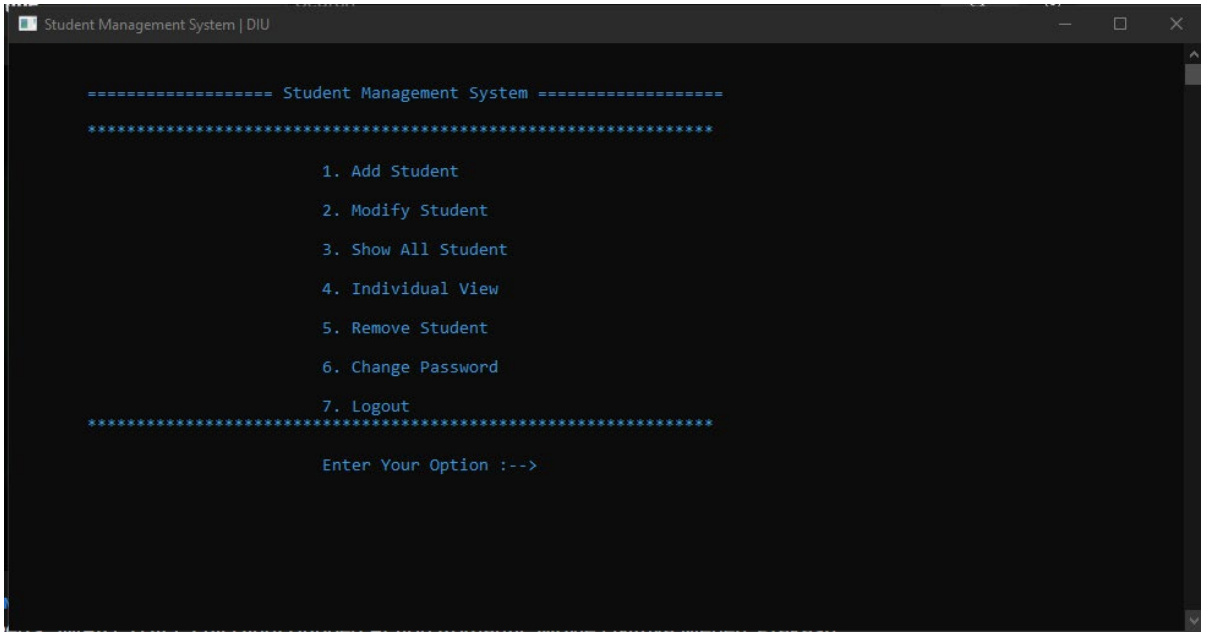
WELCOME SCREEN



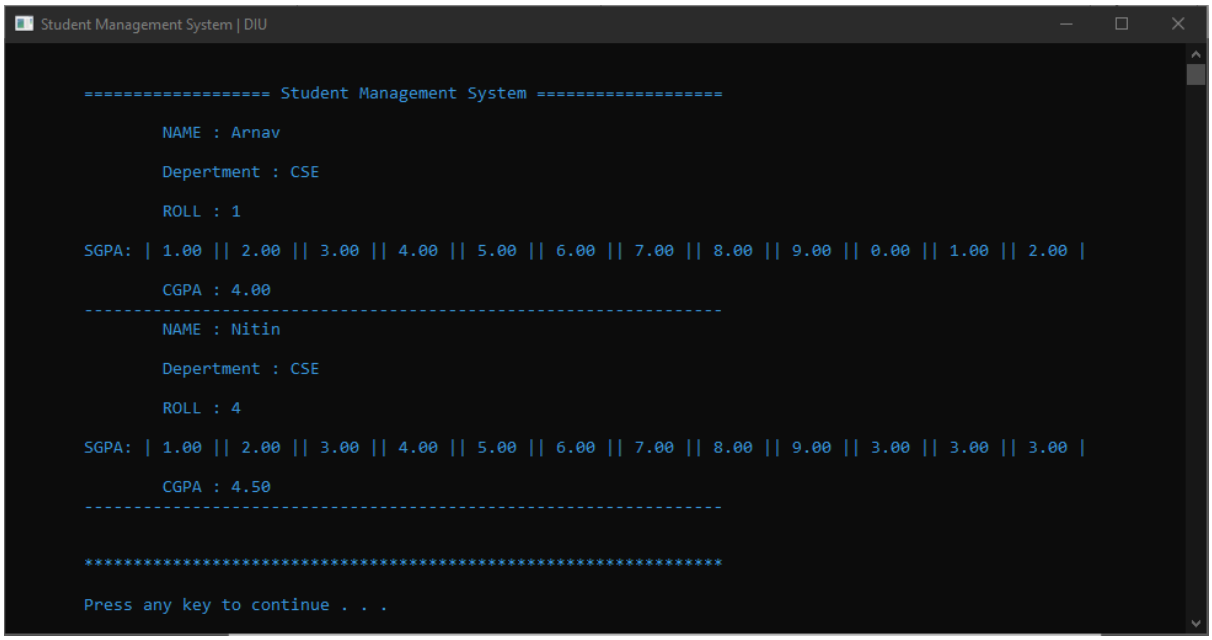
INTIAL DISPLAY



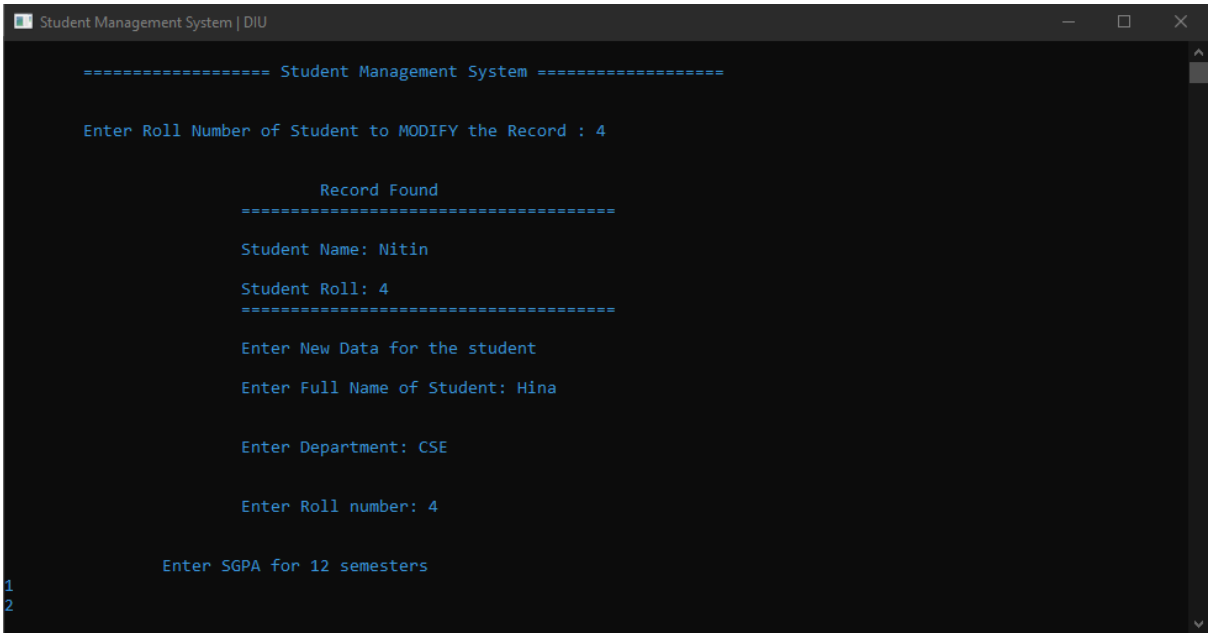
MENU SCREEN



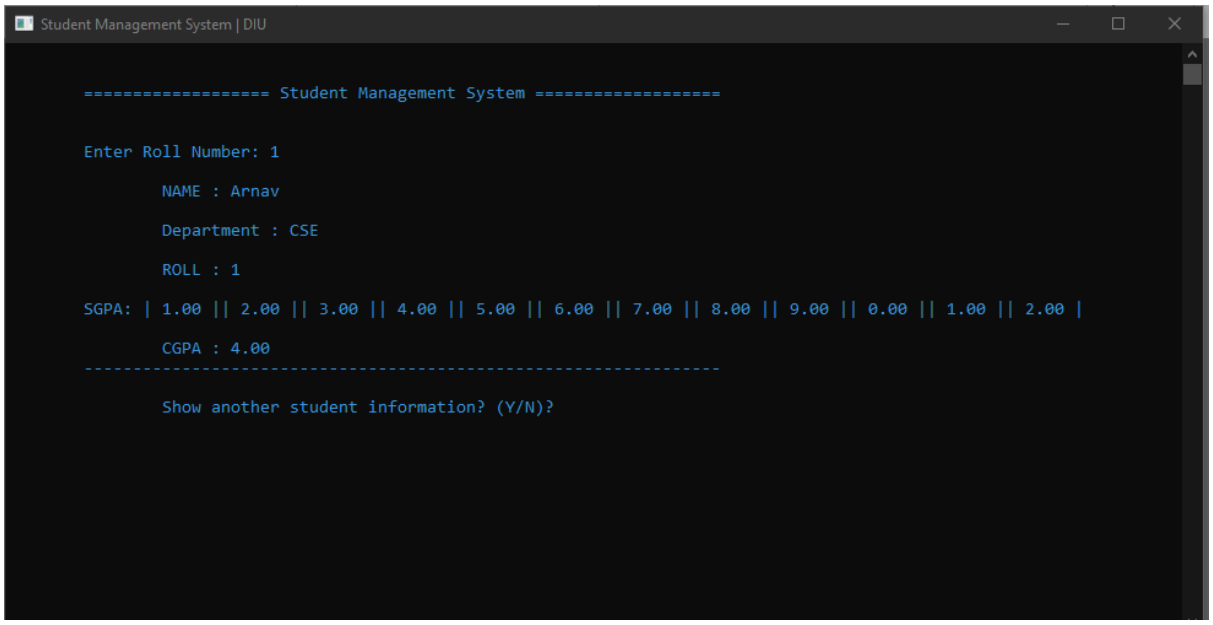
DISPLAYING RECORDS OF ALL STUDENTS



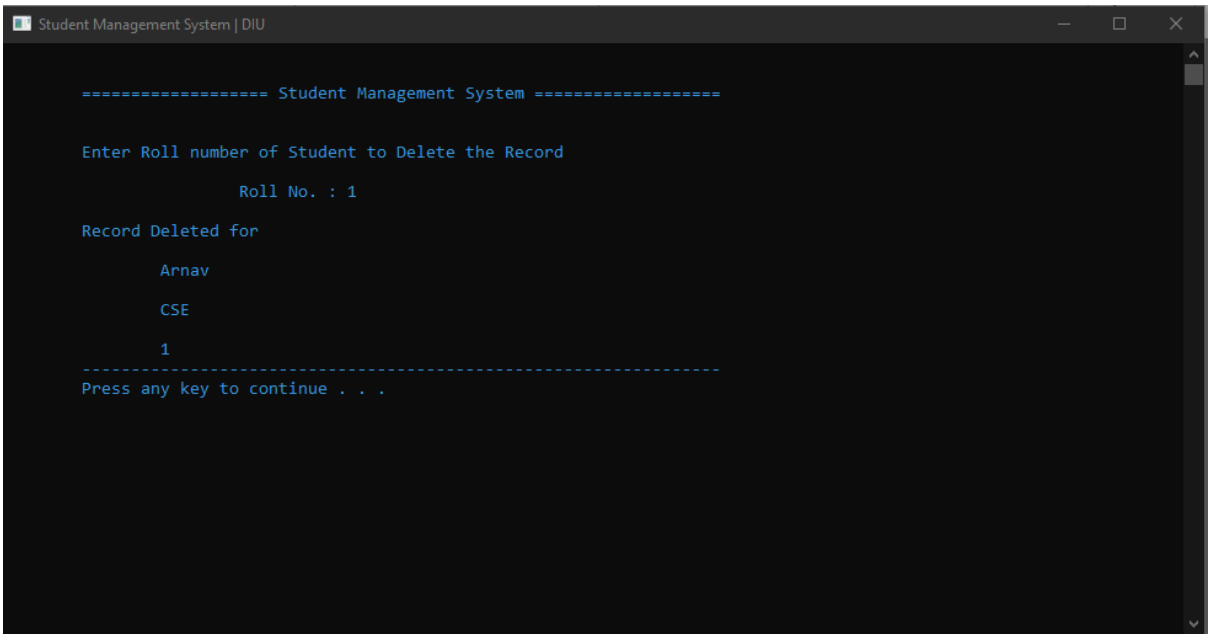
MODIFYING RECORD OF A STUDENT



INDIVIDUAL VIEW OF A STUDENT PROFILE

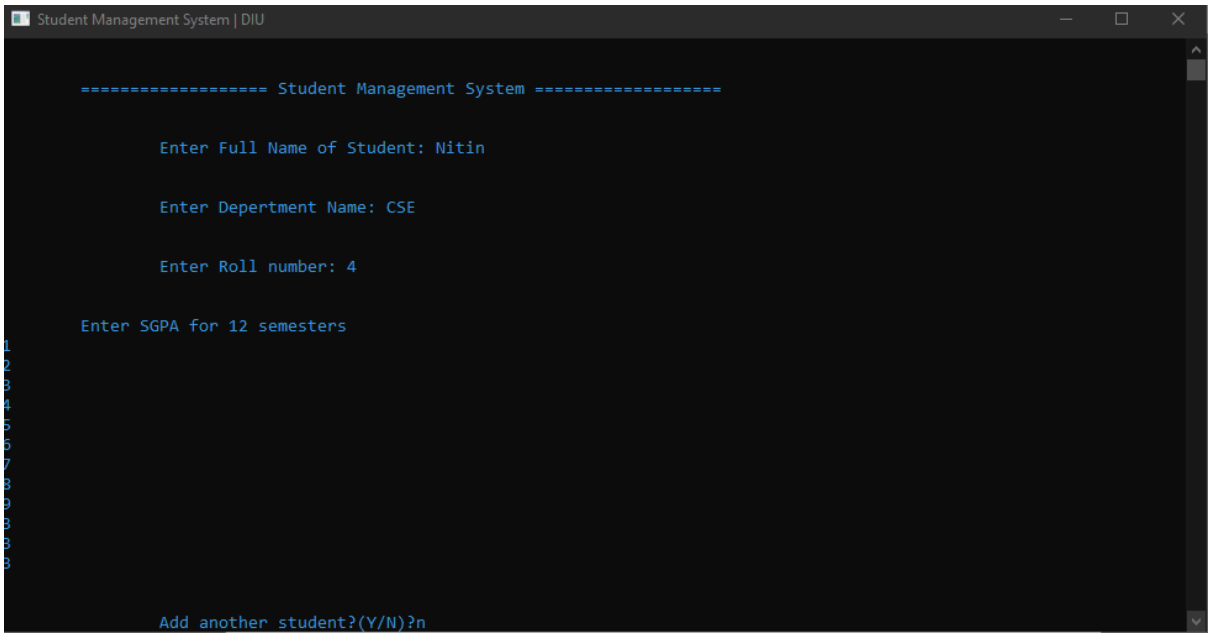


DELETING RECORD OF A STUDENT



CHANGING PASSWORD OF THE SYSTEM

ADDING A STUDENT RECORD



RESULT

Our project Student Management System provides an easy way for booking the bus tickets. Our project has succeeded in managing the data and providing the best output.

CONCLUSION

Utilizing the concepts of programming embedded in C language, the three developers have tried their best to create a simple and optimized program that does the work of a Student Management System in real life, with a user-friendly terminal for the executable file of the source code. It has also exposed the developers to the intricate technicalities when working with older generation high level languages, in this case C, which is a 3rd generation High Level Language as opposed to modern 4th generation High Level languages like Python, Ruby etc., which has made the three developers appreciate the older generation languages which pioneered the programming scenarios among the general masses while also laying the foundation for the latest generation languages.