



College Database Management System

Thesis submitted for partial fulfillment of the requirement

For the award of the degree of

Bachelor in Technology

in

Information Technology

Submitted by:

Madhav Chandok (1902160130023)

Under the guidance of:

Mr. Vishwas Chandra (Asst. Prof.)

IIMT

1994

IIMT College of Engineering, Greater Noida

Department of Information Technology

(Affiliated to Abdul Kalam Technical University)

Project Period: Oct, 2020 to Dec, 2020

Table of Content

| | | |
|------------------------------------|-------|-----|
| Declaration | ~~~~~ | i |
| Acknowledgement | ~~~~~ | ii |
| Certificate | ~~~~~ | iii |
| Abstract | ~~~~~ | iv |
| | | |
| 1. Introduction | | |
| 1.1 Background | ~~~~~ | 1 |
| 1.2 Problem Statement | ~~~~~ | 2 |
| 1.3 Problem Discussion | ~~~~~ | 2 |
| 1.4 Overview | ~~~~~ | 2 |
| | | |
| 2. Problem Solution | | |
| 2.1 Method | ~~~~~ | 3 |
| 2.2 Programming Environment | ~~~~~ | 3 |
| 2.3 Function | ~~~~~ | 4 |
| 2.4 Flow Chart of Functions | ~~~~~ | 5 |
| | | |
| 3. Conclusion | ~~~~~ | 9 |
| 4. Reference | ~~~~~ | 10 |
| | | |
| Appendix I Source Code | ~~~~~ | 11 |
| Appendix II Screenshot | ~~~~~ | 61 |

Declaration

I hereby declare that the project work entitled “College Database Management System” submitted to the IIMT College of Engineering, is a record of an original work done by me under the guidance of Mr. Vishwas Chandra, Deputy Head of Department of Information technology, IIMT College of Engineering, Greater Noida, and this project work is submitted in the partial fulfillment of the requirements for the award of the degree of Bachelor of Technology in Information Technology. The results embodied on this thesis have not been submitted to any other University or Institute for the award degree of diploma.

Name: Madhav Chandok

Roll No: 1902160130023

Acknowledgement

I would like to express my special thanks of gratitude to my mentor Mr. Vishwas Chandra who gave me the golden opportunity to do this wonderful project on the topic College Database Management System, which also helped me in doing a lot of research and I come to know about so many things.

I am very thankful to them.

Secondly I would also like to thank my friends who helped me a lot in finishing this project within the limited.

It helped me increase my knowledge and skills,

THANKS AGAIN TO ALL WHO SUPPORTED.

Thesis Certificate

This is to certify that the thesis title “College Database Management System” submitted to IIMT College of Engineering, Greater Noida by Madhav Chandok, for the award of the degree of Bachelor of Technology is a bonafide record of the research work by him under my supervision. The contents of this thesis have not been submitted to any other Institute or University for the award of any degree or diploma.

Research Guide: Mr. Vishwas Chandra

Designation: Assistant Professor

B.Tech Information Technology

IIMT College of Engineering, Greater Noida

1994

Abstract

The objective of College Information Management System is to allow the administrator of any organization to edit and find out the personal details of a student and allows the student to keep up to date his profile. It will also facilitate keeping all the records of student and faculty members such name, father's name, phone number. Address, and so on. So all the information about a student will be available in a few seconds. College Information Management System is intended to help any origination to main and manage personal data.

It has a simple user interface and is intuitive that the users spend less time in learning the system. It has efficient security features provide data privacy. In this application you can add, remove, and update username and password.

This is a simple program written in C program using the file handling concept.

Chapter 1: Introduction

1.1 Background

Database Technology is referred as a collection of logically related data, and DBMS as a software system allowing the users to define, create, maintain and control access to the database.

Database management systems are set up on specific data handling concepts, as the practice of administrating a database evolves. The earliest databases only handled individual single pieces of specially formatted data. Today's more evolved systems can handle different kinds of less formatted data and tie them together in more elaborate ways. [1]

The process of constructing such kind of systems is not so simple. It involves a mutual development of application program and database. The application program is actually the bridge between the users and the database, where the data is stored. Thus, the well-developed application program and database are very important for the reliability, flexibility and functionality of the system.

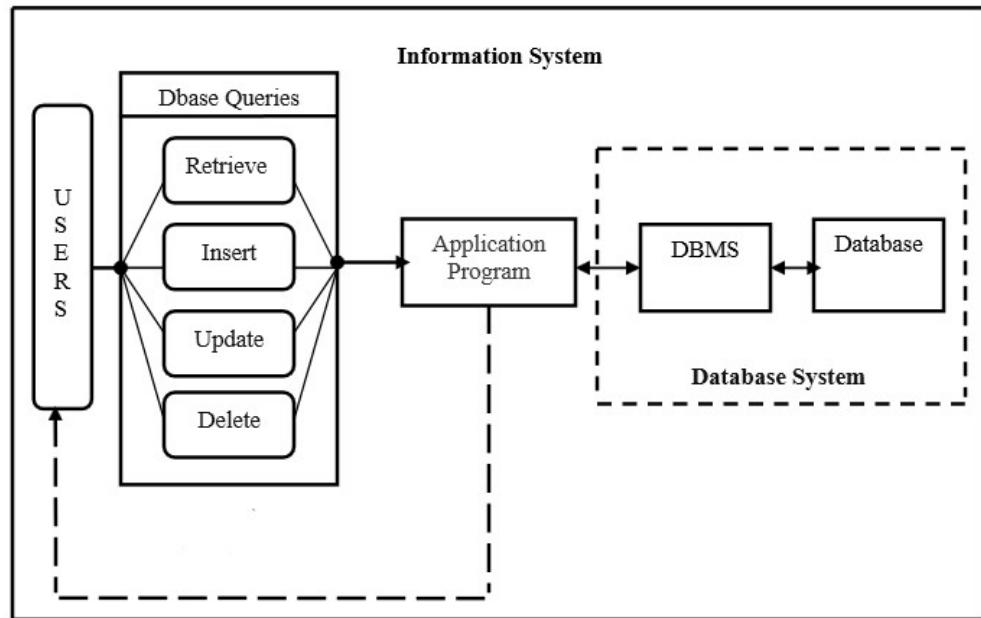


Fig 1.1: Principle of Database Information System [2]

1.2 Problem Statement

This synopsis documentation goes through the whole process of both application program and database development. It also comprises the development tools have been utilized for these purposes.

1.3 Problem Discussion:

This system should consist of an application program, on one hand, and a database on the other. The program should perform the basic operations upon the database as retrieving, inserting, updating and deleting data.

It is a kind of strategy to start the development from designing and constructing the database, as this structure will determine the further structure of the application program.

The Interface of the program should be user-friendly, and the program should be as easy for use as it is possible. Both controls and forms should logically and functionally be related within the program and fully respond to the structure of the database.

Another problem is establishing the connections with the database, every time, when a query is needed to be performed upon it. Exception-handling should also be taken into an account during the system's development due to eventual exceptions that may occur.

1.4 Overview:

The next chapter and its subsections will turn the attention to the method for resolving the problem, the programming environments used for developing the system and the implementation of the operations performed upon the database.

Chapter 2: Problem Solution

This chapter involves some subsections that concern the basic scheme of resolving the given task and comprise both the methods and tools of its development as well.

2.1 Method

At the very start of the project, I proceeded to a decision to carry out the development of my task into the following steps:

1. Exploring the technique to develop application.
2. Designing the database.
3. Database design.
4. Program's Structure Analyzing.
5. GUI (Graphical User Interface) constructing.
6. Testing.

2.2 Programming Environment

The given task concerns a small organization. I decided to use the file handling concept as a Database Management System and C language as a programming language for developing my project.

C is a procedural programming language. It was initially developed by Dennis Ritchie as a system programming language to write operating system. The main features of C language include low-level access to memory, simple set of keywords, and clean style, these features make C language suitable for system programming like operating system or compiler development. [3]

So far the operations using C program are done on a prompt / terminal which is not stored anywhere. But in the software industry, most of the programs are written to store the information fetched from the program. One such way is to store the

fetched information in a file. Different operations that can be performed on a file are: [4]

1. Creation of a new file (**fopen** with attributes as “a” or “a+” or “w” or “w++”)
2. Opening an existing file (**fopen**)
3. Reading from file (**fscanf** or **fgets**)
4. Writing to a file (**fprintf** or **fputs**)
5. Moving to a specific location in a file (**fseek**, **rewind**)
6. Closing a file (**fclose**)

2.3 Functions

There are various function involved while using College Database Management System.

- ❖ Password Protected: The whole program is password protected and fully secured. Only authorized person can access the information system. Username and password can be modified and remove from the database. Also you can add new username and password.
- ❖ Add Record: Using this function you can add the details of students and faculty members.
- ❖ Remove Record: Using this function you can remove the details of students and faculty members.
- ❖ Modify Record: Using this function you can also modify the details of students and faculty members.
- ❖ Search Record: Using this function you can search the record of the student and faculty members.

- ❖ Print All Records: Using this function you can find the details of all the students and faculty members available in database.

2.4 Flow Chart of Functions:

a. Authentication: Login into the application

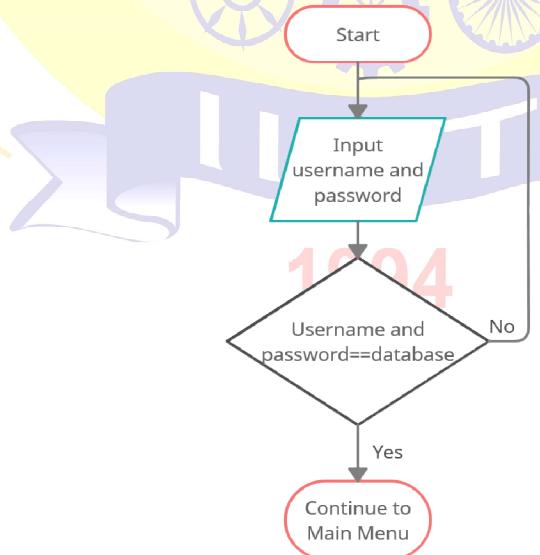
- ❖ **Goal:** Gain access into the application.
- ❖ **Brief:** This use case is when the administrator wants to access the database to access/update details of student and faculty members.

Basic Flow:

- + The application asks the administrator for the username and password.
- + The administrator enters the information.
- + The application validates and then verifies the details and sends it to admin page.
- + The administrator now given access to use application for performing task(s).

Alternative Flow:

The administrator enters invalid username and password, then not allowed to do any operations.



b. Add student/faculty member information.

- ❖ **Goal:** Registration of student/ faculty member in the database.
- ❖ **Brief:** This use case is when the administrator registers the information of student/faculty member.

Basic Flow:

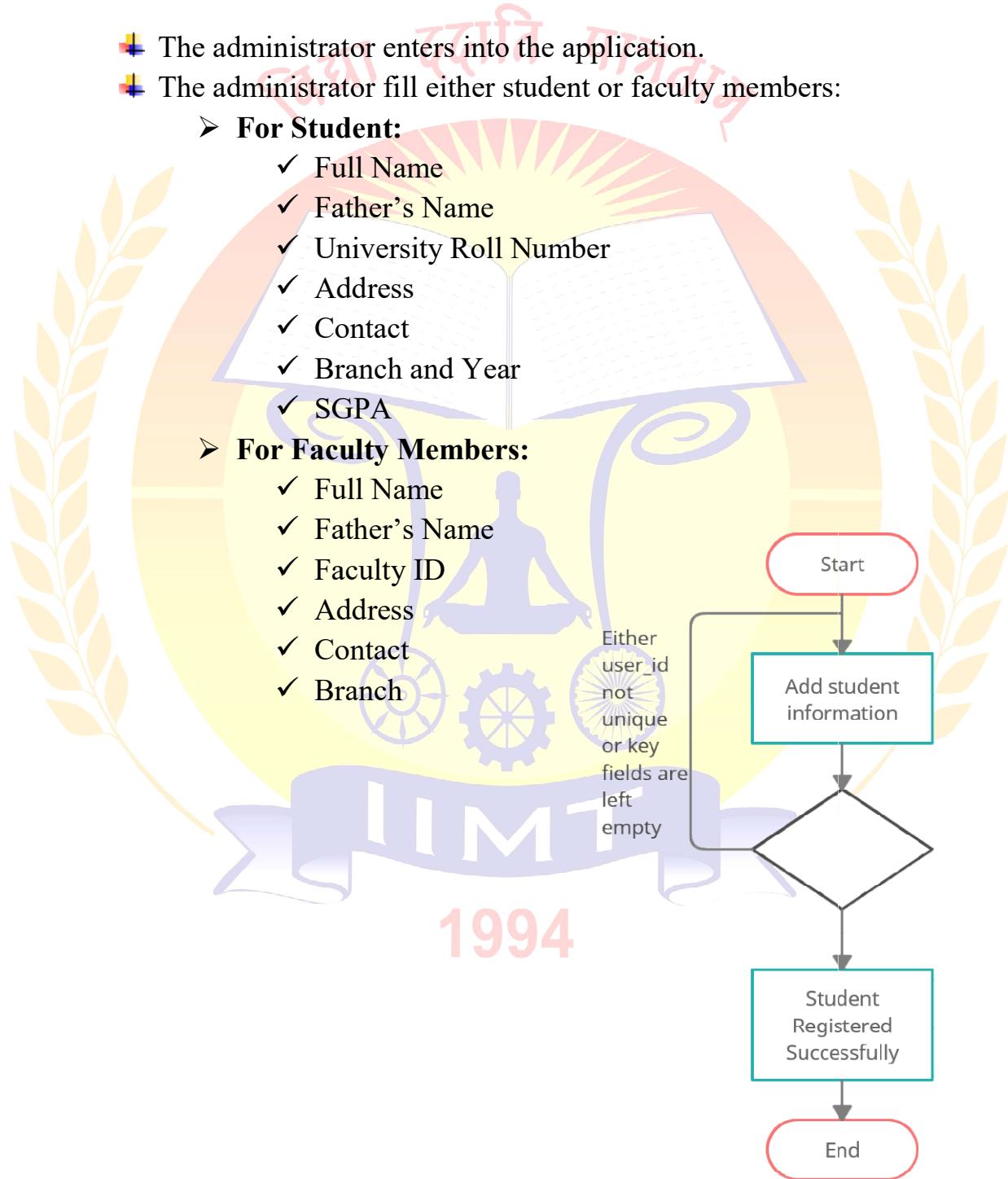
- The administrator enters into the application.
- The administrator fill either student or faculty members:

➤ **For Student:**

- ✓ Full Name
- ✓ Father's Name
- ✓ University Roll Number
- ✓ Address
- ✓ Contact
- ✓ Branch and Year
- ✓ SGPA

➤ **For Faculty Members:**

- ✓ Full Name
- ✓ Father's Name
- ✓ Faculty ID
- ✓ Address
- ✓ Contact
- ✓ Branch



c. Modification of Student/ Faculty Member Information.

- ❖ **Goal:** Modification of details of Student/Faculty Members
- ❖ **Brief:** This use case is when the administrator wants to edit the details of student/faculty members that already exist in the database.

Basic Flow:

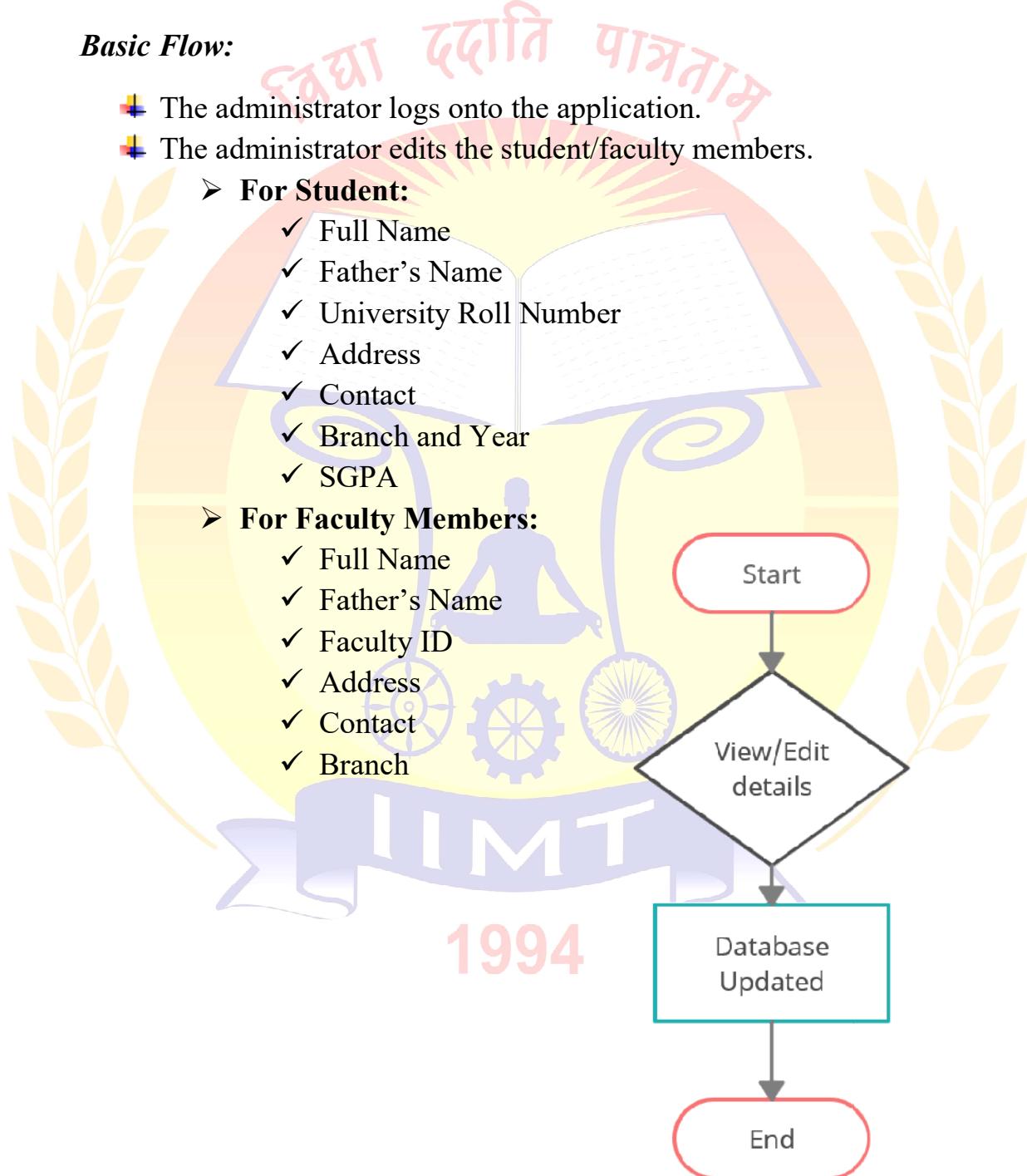
- ✚ The administrator logs onto the application.
- ✚ The administrator edits the student/faculty members.

➤ **For Student:**

- ✓ Full Name
- ✓ Father's Name
- ✓ University Roll Number
- ✓ Address
- ✓ Contact
- ✓ Branch and Year
- ✓ SGPA

➤ **For Faculty Members:**

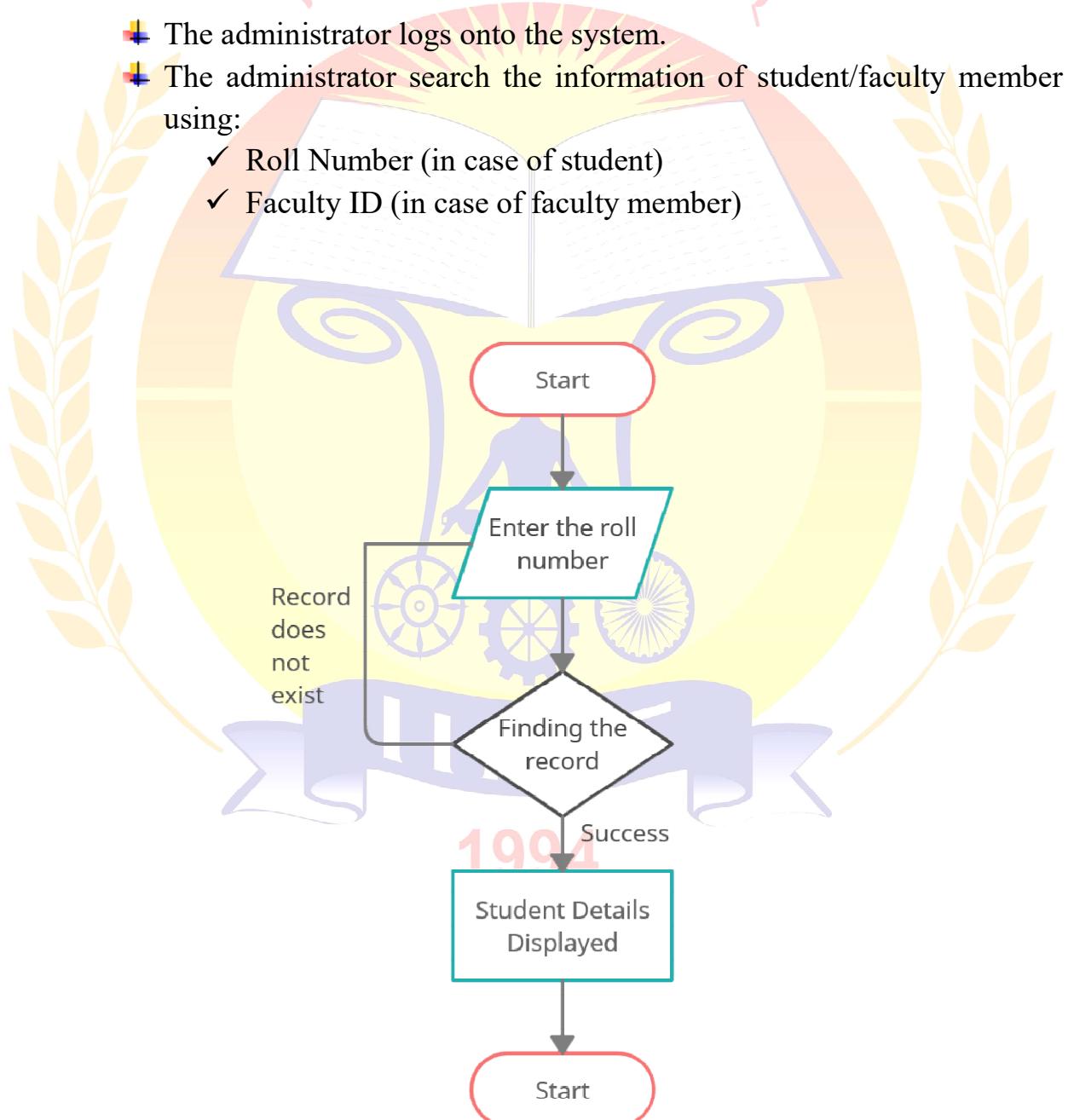
- ✓ Full Name
- ✓ Father's Name
- ✓ Faculty ID
- ✓ Address
- ✓ Contact
- ✓ Branch



d. Searching Record of Student/Faculty Members in the database.

- ❖ **Goal:** Search the information of student/faculty members in the database.
- ❖ **Brief:** This use case is when the administrator wants to view the details of student/faculty members that already exist in the database.

Basic Flow:



Chapter 3: Conclusion

In this report, an information system development has been presented. It was emphasized on the basic steps, consequently taken during the project development course as a particular attention was turned to the basic operative function performed upon the data into the database.

This report's content comprises the whole task solution, starting from the programming environment have been selected, going through the design, the application's analysis and construction and at last the testing.

As a future work, some additional stuff could be implemented and integrated into the application code making it much more reliable and flexible.

4. Reference

- [1]. <https://www.techopedia.com/definition/24361/database-management-systems-dbms>
- [2]. <http://lnu.diva-portal.org/smash/get/diva2:204828/FULLTEXT01.pdf>
- [3]. <https://www.geeksforgeeks.org/c-programming-language/>
- [4]. <https://www.geeksforgeeks.org/basics-file-handling-c/>



5. Source Code

```
/*=====COLLEGE RECORD MANAGEMENT SYSTEM=====
```

```
MINI PROJECT CREATED BY MADHAV CHANDOK
```

```
UNIVERSITY ROLL NO: 1902160130023
```

```
BRANCH: INFORMATION TECHNOLOGY 2ND YEAR 3RD SEMESTER
```

```
COLLEGE: IIMT COLLEGE OF ENGINEERING, GREATER NOIDA-201310
```

```
UNIVERSITY: ABDUL KALAM TECHNICAL UNIVERSITY (AKTU), LUCKNOW-226031
```

```
*/
```

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

```
#include<conio.h>
```

```
#include<stdlib.h>
```

```
#include<string.h>
```

```
#include<math.h>
```

```
//~~~~~DECLARATION OF STRUCTURE USED~~~~~
```

```
struct authentication
```

```
{
```

```
    char username[20];
```

```
    char password[10];
```

```
};
```

```
struct stud
```

```
{
```

```
    char name[50];
```

```
    char f_name[50];
```

```
    char roll[50];
```

```
char address[500];
char contact[20];
char branch_year[50];
float sgpa[9];
float cgpa;
};

struct faculty
{
    char name[50];
    char f_name[50];
    char id[50];
    char address[500];
    char contact[20];
    char branch[50];
};

//~~~~~DECLARATION OF FUNCTION USED~~~~~
void printHead();
void printChar(char ch,int n);
void printLast();
void pass(FILE *pas);
int login(FILE *pas);
void add(FILE *fp1, FILE *fp2);
FILE *delstud(FILE *fp);
FILE *delfacul(FILE *fp);
void modifystud(FILE *fp);
void modifyfacul(FILE *fp);
void displayListstud(FILE *fp);
```

```
void displayListfacul(FILE *fp);
void searchRecordstud(FILE *fp);
void searchRecordfacul(FILE *fp);
void update_user_pass(FILE *pas);
FILE *deletepass(FILE *pas);
void loading_screen();
void information();
void thanks();
```

//~~~~~FUNCTION THAT WILL PRINT COLLEGE NAME~~~~~

```
void printHead()
{
    printf("\e[01;97m");
    printChar('\xdb',120);
    printf("\n");
    printChar('=',120);
    printf("\n\t\t\t\t\t\t [COLLEGE DATABASE MANAGEMENT SYSTEM]");
    printf("\n\n\t\t\t\t\t\t IIMT COLLEGE OF ENGINEERING, GREATER NOIDA");
    printf("\n");
    printChar('=',120);
    printf("\n");
    printChar('\xdb',120);
    printf("\033[0m");
}
```

```
void printChar(char ch,int n)
{
    while(n--)
    {
```

```

printf("%c",ch);
}

}

void printLast()
{
    printf("\e[1;97m");
    printf("\n\n");
    printChar('=',120);
    printf("\n");
    printChar('\xdb',120);
    printf("\n");
    printChar('=',120);
    printf("\033[0m");
}
//~~~~~FUNCTION THAT WILL ADD NEW USERNAME AND PASSWORD TO DATABASE~~~~~

void pass(FILE *pas)
{
    system("cls");
    printHead();
    struct authentication log;
    fseek(pas,0,SEEK_END);
    printf("\033[01;36m");
    printf("\n\n\n\t\t\t~~~~~ USERNAME AND PASSWORD ENTRY MENU
~~~~~");
    printf("\033[01;33m");

```

```
printf("\n\n\n\t\t Enter Username: ");

scanf("%s",log.username);

printf("\n\t\t Enter Password: ");

scanf("%s",log.password);

printf("\033[1;32m");

printf("\n\n\n\t\t SUCCESS! YOUR USERNAME AND PASSWORD SUCCESSFULLY REGISTERED");

printf("\033[0m");

printf("\n\n\t\t Press any key to continue....");

printLast();

getch();

fwrite(&log,sizeof(log),1,pas);

fflush(stdin);

fclose(pas);

}

//~~~~~MULTI USER LOGIN SYSTEM FUNCTION~~~~~

int login(FILE *pas)

{

    struct authentication log;

    int x=0;

    char user[20];

    char pass[10];

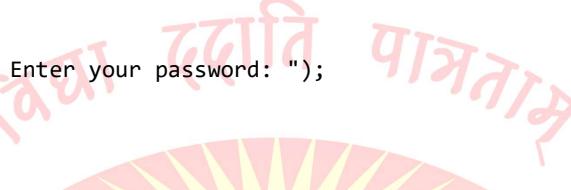
    while(x!=3)

    {

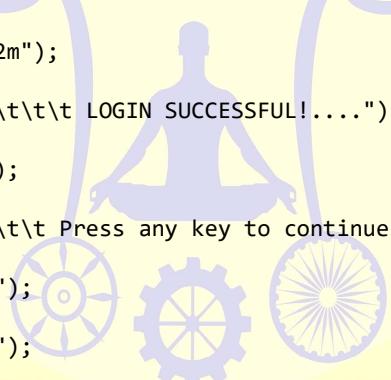
        system("cls");

        printHead();

        printf("\e[01;36m");
```



```
printf("\n\n\t\t\t ~~~~~");
printf("\n\t\t\t\t Login Page");
printf("\n\t\t\t ~~~~~");
printf("\033[01;33m");
printf("\n\n\t\t\t Enter your username: ");
scanf("%s",&user);
printf("\n\t\t\t Enter your password: ");
scanf("%s",&pass);
printf("\033[0m");
rewind(pas);
while(fread(&log,sizeof(log),1,pas)==1)
{
    if(strcmp(user,log.username)==0 && strcmp(pass,log.password)==0)
    {
        printf("\033[1;32m");
        printf("\n\n\t\t\t LOGIN SUCCESSFUL!....");
        printf("\033[0m");
        printf("\n\t\t\t Press any key to continue....");
        printf("\e[1;97m");
        printf("\n\n");
        printChar('=',120);
        printf("\n");
        printChar('\xdb',120);
        printf("\n");
        printChar('=',120);
        printf("\033[0m");
        getch();
        fclose(pas);
        return 1;
    }
}
```



```

        }

    if(strcmp(user,log.username)!=0 || strcmp(pass,log.password)!=0)

    {

        printf("\033[1;31m");

        printf("\n\n\n\t\t\t\t LOGIN UNSUCCESSFUL!....TRY AGAIN....");

        printf("\033[0m");

        x++;

    }

    printf("\n\n\t\t\t\t Press any key to continue....");

    printf("\e[1;97m");

    printf("\n\n\n\n");

    printChar('=',120);

    printf("\n");

    printChar('\xdb',120);

    printf("\n");

    printChar('=',120);

    printf("\033[0m");

    getch();

    if(x==3)

        fclose(pas);

    }

}

//~~~~~FUNCTION THAT WILL ADD STUDENT AND FACULTY MEMBER INFORMATION TO
DATABASE~~~~~
```

```

void add(FILE *fp1, FILE *fp2)

{

    int flag=1;

    while(flag==1)
```



```
fflush(stdin);

fgets(s.name,50,stdin);

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

printf("\n\t\t\t Enter Father's Name: ");

fflush(stdin);

fgets(s.f_name,50,stdin);

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

printf("\n\t\t\t Enter University Roll Number: ");

fflush(stdin);

fgets(s.roll,50,stdin);

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

printf("\n\t\t\t Enter Address: ");

fflush(stdin);

fgets(s.address,500,stdin);

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

printf("\n\t\t\t Enter Contact: ");

fflush(stdin);

fgets(s.contact,20,stdin);

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

printf("\n\t\t\t Enter Branch and Year: ");

fflush(stdin);

fgets(s.branch_year,50,stdin);

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

printf("\n\t\t\t Enter SGPA of 8 semesters\n\n");
```

```
float cgpa=0.0;

float sgpano=0.0;

for(int i=0;i<8;i++)

{

    printf("\t\t\t\tSGPA %d semester: ",i+1);

    scanf("%f",&s.sgpa[i]);

    cgpa+=s.sgpa[i];

    if(s.sgpa[i]!=0.0)

        sgpano++;

}

cgpa/=sgpano;

s.cgpa=cgpa;

printf("\033[0m");

fwrite(&s,sizeof(s),1,fp1);

printf("\033[1;32m");

printf("\n\n\n\t\t\t\t DO YOU WANT TO ENTER ANOTHER STUDENT INFORMATION....(Y/N)\n");

printf("\033[0m");

fflush(stdin);

printLast();

a=getch();

}

flag=0;

break;

case 2:while(a=='Y' || a=='y')

{



    system("cls");

    printHead();
```


printf("\n\t\t\t Enter Branch: ");
fflush(stdin);
fgets(f.branch,50,stdin);
f.branch[strlen(f.branch)-1]='\0';

fwrite(&f,sizeof(f),1,fp2);

printf("\033[1;32m");

printf("\n\n\t\t\t DO YOU WANT TO ENTER ANOTHER FACULTY MEMBER INFORMATION....(Y/N)\n");

fflush(stdin);
printLast();
a=getch();
}

flag=0;
break;

case 3: flag=0;
break;

default:flag=1;
printf("\033[1;31m");
printf("\n\n\t\t\t ERROR! YOU HAVE ENTERED WRONG OPTION!");
printf("\n\t\t\t TRY AGAIN!");
printf("\033[0m");
printLast();
getch();
break;

}

}

//~~~~~FUNCTION THAT WILL REMOVE STUDENT INFORMATION FROM DATABASE~~~~~

```
FILE *delstud(FILE *fp)
{
    struct stud s;
    char tempRoll[30];
    int flag=0;
    FILE *ft;
    system("cls");
    printHead();

    ft=fopen("temp.txt","w+");

    printf("\033[01;36m");
    printf("\n\n\n\t\t\t~~~~~ STUDENT RECORD DELETION PAGE
~~~~~");
    printf("\033[01;33m");
    printf("\n\n\n\t\t\t ENTER UNIVERSITY ROLL NUMBER OF STUDENT");
    printf("\n\n\t\t\t UNIVERSITY ROLL NO: ");
    scanf("%s",&tempRoll);

    rewind(fp);
    while((fread(&s,sizeof(s),1,fp))==1)
    {
        if(strcmp(s.roll,tempRoll)==0)

        {
            system("cls");
            printHead();
            printf("\033[1;32m");
            printf("\n\n\n\t\t\t SUCCESS! RECORD FOUND!");
        }
    }
}
```

```
flag=1;

sleep(2);

system("cls");

printHead();

printf("\033[01;36m");

printf("\n\n\t\t\t RECORD DELETED FOR");

printf("\033[1;31m");

printf("\n\t\t\t FULL NAME: %s",s.name);

printf("\n\t\t\t FATHER'S NAME: %s",s.f_name);

printf("\n\t\t\t UNIVERSITY ROLL NUMBER: %s",s.roll);

printf("\n\t\t\t ADDRESS: %s",s.address);

printf("\n\t\t\t CONTACT: %s",s.contact);

printf("\n\t\t\t BRANCH AND YEAR: %s",s.branch_year);

printf("\033[0m");

printf("\n\n\t\t Press any key to continue.... ");

getch();

printf("\033[1;32m");

printf("\n\t\t\t RECORD SUCCESSFULLY REMOVED FROM DATABASE!!!!");

printf("\033[0m");

continue;

}

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

}

fclose(fp);

fclose(ft);

remove("StudentInfo.txt");

rename("temp.txt","StudentInfo.txt");
```

```
if(flag==0)
{
    system("cls");
    printHead();
    printf("\033[1;31m");
    printf("\n\n\n\t\t\t ERROR! RECORD NOT FOUND");
    printf("\033[0m");
}
printf("\n\n\n\t\t\t Press any key to continue....\n");
getch();
return fp;
}

//~~~~~FUNCTION THAT WILL REMOVE FACULTY MEMBER INFORMATION FROM DATABASE~~~~~

FILE *delfacul(FILE *fp)
{
    struct faculty f;
    char tempRoll[30];
    int flag=0;
    FILE *ft;
    system("cls");
    printHead();

    ft=fopen("temp.txt", "w+");
    printf("\033[01;36m");
```



```
printf("\n\n\n\t\t\t~~~~~ FACULTY MEMBER RECORD DELETION PAGE  
~~~~~");  
  
printf("\033[01;33m");  
  
printf("\n\n\n\t\t\t ENTER FACULTY ID OF FACULTY MEMBER");  
  
printf("\n\n\t\t\t FACULTY ID: ");  
  
scanf("%s",&tempRoll);  
  
rewind(fp);  
  
while((fread(&f,sizeof(f),1,fp))==1)  
{  
    if(strcmp(f.id,tempRoll)==0)  
    {  
        system("cls");  
        printHead();  
        printf("\033[1;32m");  
        printf("\n\n\n\t\t\t SUCCESS! RECORD FOUND!");  
        flag=1;  
        sleep(2);  
  
        system("cls");  
        printHead();  
        printf("\033[01;36m");  
        printf("\n\n\n\t\t\t RECORD DELETED FOR");  
        printf("\033[1;31m");  
        printf("\n\n\t\t\t FULL NAME: %s",f.name);  
        printf("\n\n\t\t\t FATHER'S NAME: %s",f.f_name);  
        printf("\n\n\t\t\t FACULTY ID: %s",f.id);  
        printf("\n\n\t\t\t ADDRESS: %s",f.address);  
        printf("\n\n\t\t\t CONTACT: %s",f.contact);  
        printf("\n\n\t\t\t BRANCH: %s",f.branch);  
        printf("\033[0m");
```

```
    printf("\n\n\n\t\t\t Press any key to continue.... ");

    getch();

    printf("\033[1;32m");

    printf("\n\n\n\t\t\t RECORD SUCCESSFULLY REMOVED FROM DATABASE!!!!");

    printf("\033[0m");

    continue;

}

fwrite(&f,sizeof(f),1,ft);

}

fclose(fp);

fclose(ft);

remove("facultyInfo.txt");

rename("temp.txt","facultyInfo.txt");

if(flag==0)

{

    system("cls");

    printHead();

    printf("\033[1;31m");

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

    printf("\033[0m");

}

printf("\n\n\n\t\t\t Press any key to continue...\n");

printLast();

getch();

return fp;

}
```



```
flag=0;

}

if(flag==1)
{
    system("cls");
    printHead();
    printf("\033[1;32m");
    printf("\n\n\n\t\t\t SUCCESS! RECORD FOUND!");
    flag=1;
    sleep(2);

    system("cls");
    printHead();
    printf("\033[01;36m");
    printf("\n\n\n\t\t\t RECORD MODIFICATION FOR");
    printf("\033[1;32m");
    printf("\n\n\t\t\t FULL NAME: %s",s.name);
    printf("\n\n\t\t\t FATHER'S NAME: %s",s.f_name);
    printf("\n\n\t\t\t UNIVERSITY ROLL NUMBER: %s",s.roll);
    printf("\n\n\t\t\t ADDRESS: %s",s.address);
    printf("\n\n\t\t\t CONTACT: %s",s.contact);
    printf("\n\n\t\t\t BRANCH AND YEAR: %s",s.branch_year);
    printf("\n\n\t\t\t SGPA: ");
    for(int l=0;l<8;l++)
    {
        printf("|%.2f|",s.sgpa[l]);
    }
    printf("\n\n\t\t\t CGPA: %.2f",s.cgpa);
```

printf("\033[0m");
printf("\n\n\n\t\t\tPress any key to continue....");
getch();

system("cls");
printHead();

printf("\033[01;36m");
printf("\n\n\n\t\t\t~~~~~ ENTER NEW INFORMATION OF STUDENT
~~~~~");  
printf("\033[01;33m");  
printf("\n\n\n\t\t\t Enter Full Name: ");  
fflush(stdin);  
fgets(s.name,50,stdin);  
s.name[strlen(s.name)-1]='\0';

printf("\n\t\t\t Enter Father's Name: ");  
fflush(stdin);  
fgets(s.f\_name,50,stdin);  
s.f\_name[strlen(s.f\_name)-1]='\0';

printf("\n\t\t\t Enter University Roll Number: ");  
fflush(stdin);  
fgets(s.roll,50,stdin);  
s.roll[strlen(s.roll)-1]='\0';

printf("\n\t\t\t Enter Address: ");  
fflush(stdin);  
fgets(s.address,500,stdin);  
s.address[strlen(s.address)-1]='\0';

```
printf("\n\t\t\t Enter Contact: ");

fflush(stdin);

fgets(s.contact,20,stdin);

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

printf("\n\t\t\t Enter Branch and Year: ");

fflush(stdin);

fgets(s.branch_year,50,stdin);

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

printf("\n\t\t\t Enter SGPA of 8 semesters\n\n");

float cgpa=0.0;

float sgpano=0.0;

for(int i=0;i<8;i++)

{

    printf("\t\t\t\tSGPA %d semester: ",i+1);

    scanf("%f",&s.sgpa[i]);

    cgpa+=s.sgpa[i];

    if(s.sgpa[i]!=0.0)

        sgpano++;

}

cgpa/=sgpano;

s.cgpa=cgpa;

printf("\033[0m");



fseek(fp,-760,SEEK_CUR);

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

}
```

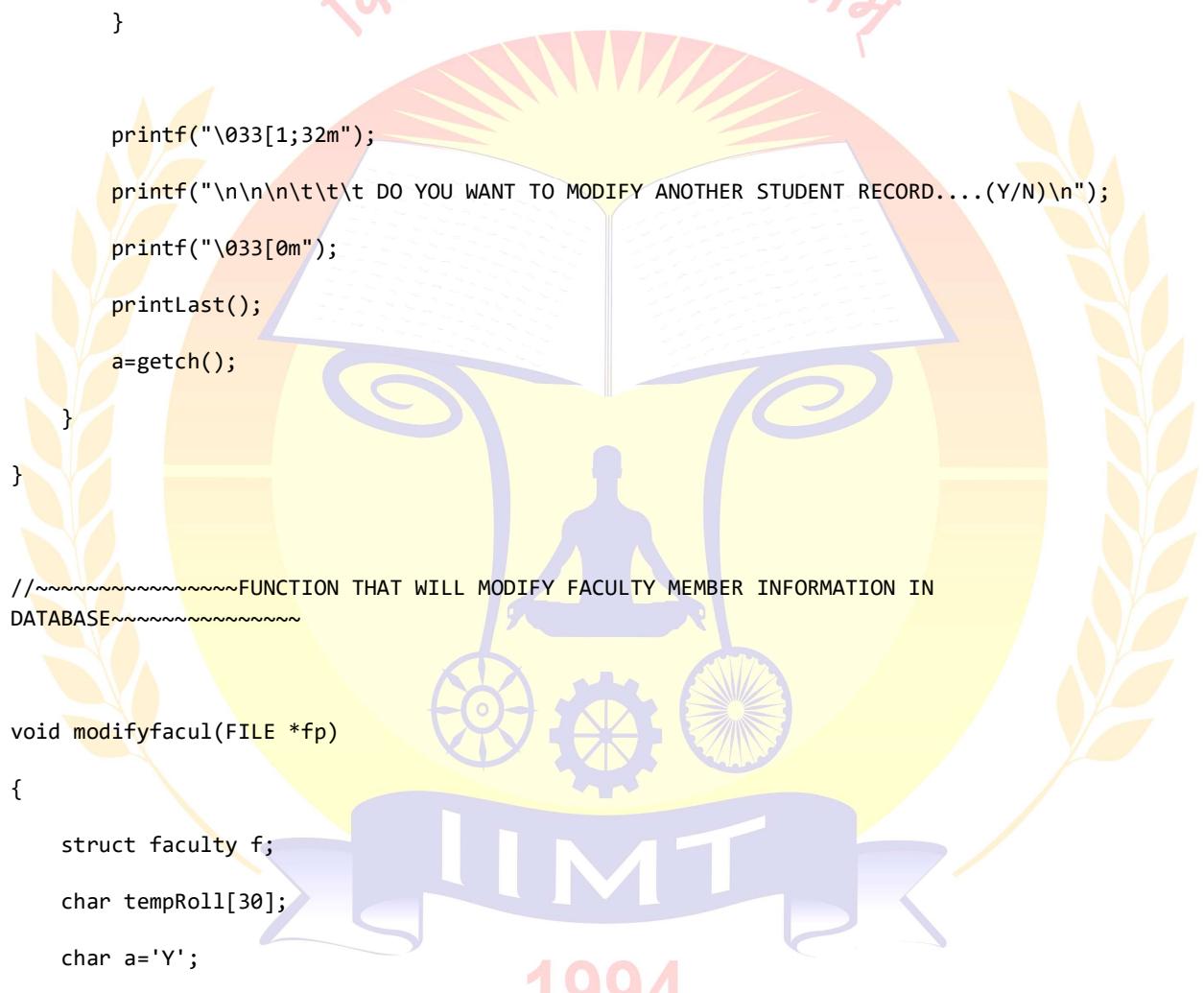
```
else
{
    system("cls");
    printHead();
    printf("\033[1;31m");
    printf("\n\n\n\t\t\t ERROR! RECORD NOT FOUND");
    printf("\033[0m");
}

printf("\033[1;32m");
printf("\n\n\n\t\t\t DO YOU WANT TO MODIFY ANOTHER STUDENT RECORD....(Y/N)\n");
printf("\033[0m");
printLast();
a=getch();
}

}

//~~~~~FUNCTION THAT WILL MODIFY FACULTY MEMBER INFORMATION IN
DATABASE~~~~~
void modifyfacul(FILE *fp)
{
    struct faculty f;
    char tempRoll[30];
    char a='Y';
    int flag=0;

    while(a=='Y' || a=='y')
    {
        system("cls");
        printHead();
```





printHead();  
printf("\033[01;36m");  
printf("\n\n\n\t\t\t RECORD MODIFICATION FOR");  
printf("\033[1;32m");  
printf("\n\n\t\t\t FULL NAME: %s",f.name);  
printf("\n\n\t\t\t FATHER'S NAME: %s",f.f\_name);  
printf("\n\n\t\t\t FACULTY ID: %s",f.id);  
printf("\n\n\t\t\t ADDRESS: %s",f.address);  
printf("\n\n\t\t\t CONTACT: %s",f.contact);  
printf("\n\n\t\t\t BRANCH: %s",f.branch);  
printf("\033[0m");  
printf("\n\n\n\t\tPress any key to continue....");  
getch();  
  
system("cls");  
printHead();  
  
printf("\033[01;36m");  
printf("\n\n\n\t\tENTER NEW INFORMATION OF FACULTY  
MEMBER ~~~~~");  
printf("\033[01;33m");  
printf("\n\n\t\tEnter Full Name: ");  
fflush(stdin);  
fgets(f.name,50,stdin);  
f.name[strlen(f.name)-1]='\0';  
  
printf("\n\t\tEnter Father's Name: ");  
fflush(stdin);  
fgets(f.f\_name,50,stdin);  
f.f\_name[strlen(f.f\_name)-1]='\0';

```
printf("\n\t\t\t Enter Faculty ID: ");

fflush(stdin);

fgets(f.id,50,stdin);

f.id[strlen(f.id)-1]='\0';

printf("\n\t\t\t Enter Address: ");

fflush(stdin);

fgets(f.address,500,stdin);

f.address[strlen(f.address)-1]='\0';

printf("\n\t\t\t Enter Contact: ");

fflush(stdin);

fgets(f.contact,20,stdin);

f.contact[strlen(f.contact)-1]='\0';

printf("\n\t\t\t Enter Branch: ");

fflush(stdin);

fgets(f.branch,50,stdin);

f.branch[strlen(f.branch)-1]='\0';

fseek(fp,-720,SEEK_CUR);

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

}

else

{

system("cls");

printHead();

printf("\033[1;31m");

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

printf("\033[0m");

```



```
printf("\n\n\t\t\t ADDRESS: %s",s.address);

printf("\n\n\t\t\t CONTACT: %s",s.contact);

printf("\n\n\t\t\t BRANCH AND YEAR: %s",s.branch_year);

printf("\n\n\t\t\t SGPA: ");

for(int i=0;i<8;i++)

{

    printf("|%.2f|",s.sgpa[i]);

}

printf("\n\n\t\t\t CGPA: %.2f",s.cgpa);

printf("\033[0m");

printLast();

sleep(1);

}

printf("\n\n\n\t\t\t Press any key to continue....\n");

printLast();

getch();

}

//~~~~~FUNCTION THAT WILL DISPLAY ALL FACULTY MEMBER INFORMATION PRESENT IN DATABASE~~~~~
```

```
printf("\033[0m");

rewind(fp);

while(fread(&f,sizeof(f),1,fp)==1)
{
    printf("\033[01;33m");
    printf("\n\n\n\t\t\t FULL NAME: %s",f.name);
    printf("\n\n\t\t\t FATHER'S NAME: %s",f.f_name);
    printf("\n\n\t\t\t FACULTY ID: %s",f.id);
    printf("\n\n\t\t\t ADDRESS: %s",f.address);
    printf("\n\n\t\t\t CONTACT: %s",f.contact);
    printf("\n\n\t\t\t BRANCH: %s",f.branch);
    printf("\033[0m");
    printLast();
    sleep(1);
}
printf("\n\n\n\t\t Press any key to continue....\n");
printLast();
getch();
}

//~~~~~FUNCTION THAT WILL SEARCH STUDENT INFORMATION IN DATABASE~~~~~
void searchRecordstud(FILE *fp)
{
    int flag;
    struct stud s;
    char tempRoll[30];
    char a='Y';
}
```





```

    printf("\033[0m");

    printLast();

    a=getch();

}

}

//~~~~~FUNCTION THAT WILL SEARCH FACULTY MEMBER INFORMATION IN
//DATABASE~~~~~
```

```

void searchRecordfacul(FILE *fp)
{
    int flag;
    struct faculty f;
    char tempRoll[30];
    char a='Y';

    while(a=='Y' || a=='y')
    {
        system("cls");
        printHead();
        printf("\033[01;36m");
        printf("\n\n\n\t\t\t~~~~~ FACULTY MEMBER RECORD SEARCHING PAGE
~~~~~");
 printf("\033[01;33m");
 printf("\n\n\n\t\t\t ENTER FACULTY ID OF FACULTY MEMBER");
 printf("\n\t\t\t FACULTY ID: ");
 scanf("%s", &tempRoll);

 rewind(fp);
 while((fread(&f,sizeof(f),1,fp))==1)
```

```
{
 if(strcmp(f.id,tempRoll)==0)
 {
 flag=1;
 break;
 }
}
```

```
if(flag==1)
{
 system("cls");
 printHead();

 printf("\033[1;32m");
 printf("\n\n\n\t\t\t SUCCESS! RECORD FOUND!");
 flag=1;
 sleep(2);

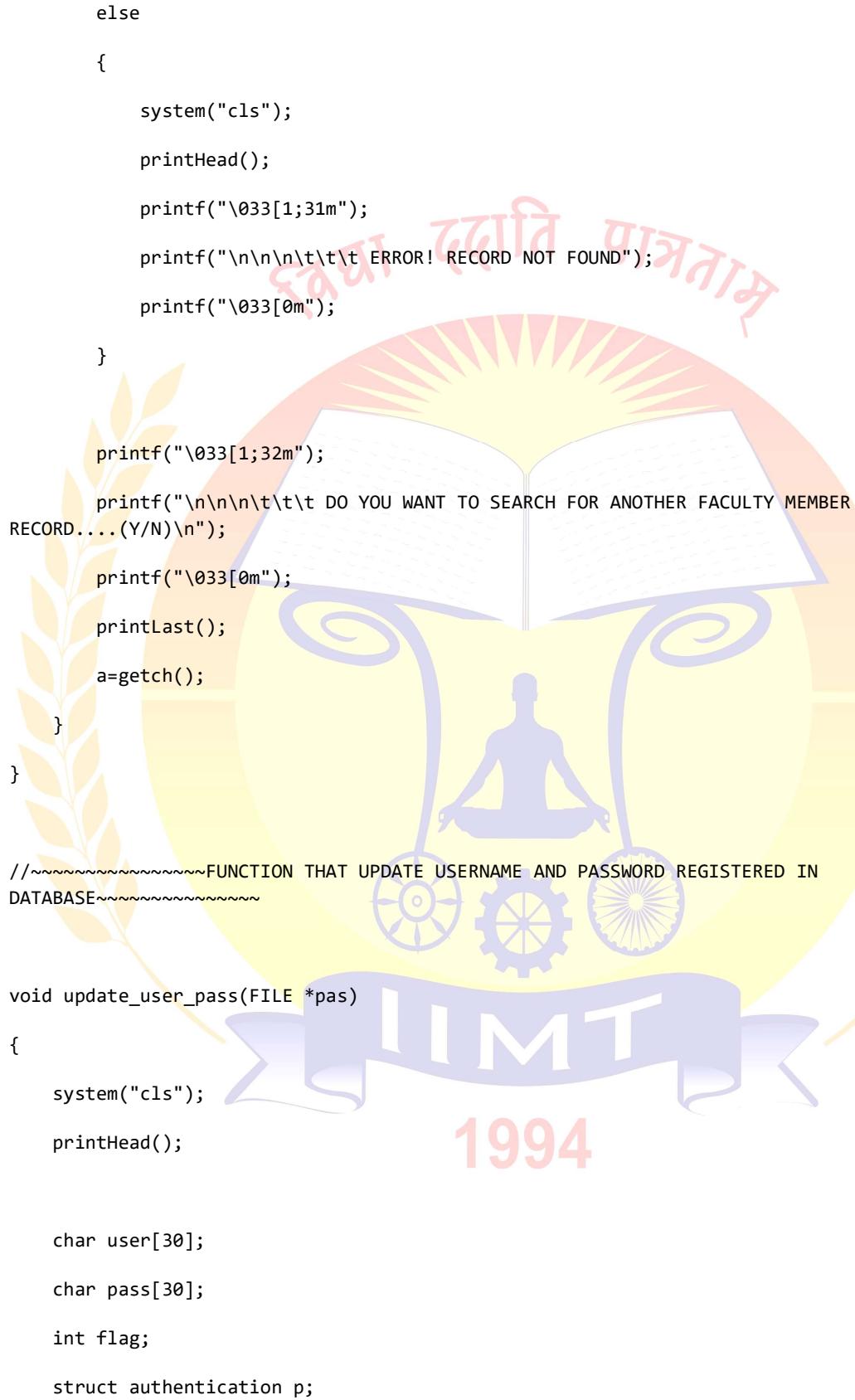
 system("cls");
 printHead();
 printf("\033[01;36m");
 printf("\n\n\n\t\t\t ~~~~~~ DETAILS OF FACULTY MEMBER
~~~~~");  
    printf("\033[1;33m");  
    printf("\n\n\t\t\t FULL NAME: %s",f.name);  
    printf("\n\n\t\t\t FATHER'S NAME: %s",f.f_name);  
    printf("\n\n\t\t\t FACULTY ID: %s",f.id);  
    printf("\n\n\t\t\t ADDRESS: %s",f.address);  
    printf("\n\n\t\t\t CONTACT: %s",f.contact);  
    printf("\n\n\t\t\t BRANCH: %s",f.branch);  
}  
}
```

```
else
{
    system("cls");
    printHead();
    printf("\033[1;31m");
    printf("\n\n\n\t\t\t ERROR! RECORD NOT FOUND");
    printf("\033[0m");
}

printf("\033[1;32m");
printf("\n\n\n\t\t\t DO YOU WANT TO SEARCH FOR ANOTHER FACULTY MEMBER
RECORD....(Y/N)\n");
printf("\033[0m");
printLast();
a=getch();
}

}

//~~~~~FUNCTION THAT UPDATE USERNAME AND PASSWORD REGISTERED IN
DATABASE~~~~~
void update_user_pass(FILE *pas)
{
    system("cls");
    printHead();
    char user[30];
    char pass[30];
    int flag;
    struct authentication p;
```



printf("\033[01;36m");  
 printf("\n\n\n\t\t\t~~~~~ USERNAME AND PASSWORD MODIFICATION PAGE  
~~~~~");

printf("\033[01;33m");
 printf("\n\n\n\t\t\t ENTER YOUR EXISTING USERNAME : ");
 scanf("%s",&user);
 printf("\n\n\t\t\t ENTER YOUR EXISTING PASSWORD: ");
 scanf("%s",&pass);

rewind(pas);
 while((fread(&p,sizeof(p),1,pas))==1)

{
 if(strcmp(user,p.username)==0 && strcmp(pass,p.password)==0)
 {
 flag=1;
 break;
 }
 }

if(flag==1)

{
 system("cls");
 printHead();
 fseek(pas,-sizeof(p),SEEK_CUR);
 }
 }

printf("\033[1;32m");
 printf("\n\n\t\t\t SUCCESS! USERNAME AND PASSWORD FOUND!");
 flag=1;

```
sleep(2);

system("cls");

printHead();

printf("\033[01;36m");
printf("\n\n\n\t\t\t~~~~~ ENTER NEW USERNAME AND PASSWORD
~~~~~");
printf("\033[01;33m");
printf("\n\n\n\t\t\t ENTER A NEW USERNAME: ");
scanf("%s",p.username);
printf("\n\n\t\t\t ENTER A NEW PASSWORD:");
scanf(" %s",p.password);

fwrite(&p,sizeof(p),1,pas);
printf("\033[1;32m");
printf("\n\n\t\t\t SUCCESS! NEW USERNAME AND PASSWORD REGISTERED");
printf("\033[0m");
}

else
{
system("cls");
printHead();
printf("\033[1;31m");
printf("\n\n\t\t\t ERROR! NO SUCH USERNAME AND PASSWORD REGISTERED");
printf("\033[0m");
}

printf("\n\n\t\t\t Press any key to continue....");
getch();
fclose(pas);

}
```

//~~~~~FUNCTION THAT DELETE USERNAME AND PASSWORD REGISTERED IN DATABASE~~~~~

```
FILE *deletepass(FILE *pas)
{
    struct authentication p;
    int siz(sizeof(p));
    char user[30],pass[30];
    int flag;
    FILE *ft;
    char a='Y';

    while(a=='Y' || a=='y')
    {
        system("cls");
        printHead();
        ft=fopen("temp.txt","w+");
        printf("\033[01;36m");
        printf("\n\n\n\t\t\t~~~~~USERNAME AND PASSWORD DELETION PAGE~~~~~");
        printf("\033[01;33m");
        printf("\n\n\n\t\t\t ENTER YOUR EXISTING USERNAME : ");
        scanf("%s",&user);
        printf("\n\n\t\t\t ENTER YOUR EXISTING PASSWORD: ");
        scanf("%s",&pass);

        rewind(pas);
        while((fread(&p,siz,1,pas))==1)
        {
```

```
if(strcmp(user,p.username)==0 && strcmp(pass,p.password)==0)
{
    flag=1;
    printf("\033[1;32m");
    printf("\n\n\n\t\t\t SUCCESS! USERNAME AND PASSWORD FOUND!");
    flag=1;
    sleep(2);
    system("cls");
    printHead();
    printf("\033[01;36m");
    printf("\n\n\n\t\t\t USERNAME AND PASSWORD FOR DELETION");
    printf("\033[1;31m");
    printf("\n\n\t\t\t USERNAME: %s",p.username);
    printf("\n\n\t\t\t PASSWORD: %s",p.password);
    printf("\033[0m");
    printf("\n\n\n\t\t\t Press any key to continue.... ");
    getch();
    printf("\033[1;32m");
    printf("\n\n\n\t\t\t USERNAME AND PASSWORD SUCCESSFULLY REMOVED FROM
DATABASE!!!!");
    printf("\033[0m");
    continue;
}
fwrite(&p,siz,1,ft);

fclose(pas);
fclose(ft);
```

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

```

if((pas=fopen("password.txt","rb+"))==NULL)
{
  return NULL;
}

if(flag==1)
{
}
else
{
  system("cls");
  printHead();
  printf("\033[1;31m");
  printf("\n\n\n\t\t\t ERROR! NO SUCH USERNAME AND PASSWORD REGISTERED");
  printf("\033[0m");
}

printf("\n\n\t\t\t DO YOU WANT TO REMOVE ANOTHER USERNAME AND PASSWORD....(Y/N)");
a=getch();
}

return pas;
}

```

//~~~~~LOADING SCREEN FUNCTION~~~~~

```

void loading_screen()
{

```



```
    printf("\n\n\n\t\t\t Press any key to continue....");

    getch();

}
```

```
void thanks()
```

```
{
```

```
    system("cls");

    printHead();
```

```
    printf("\033[01;36m");

    printf("\n\n\n\t\t\t~~~~~ < THANKS FOR USING APPLICATION >
~~~~~");
```

```
    printf("\033[1;32m");

    printf("\n\n\t\t\t~~~~~ < COLLEGE DATABASE MANAGEMENT SYSTEM >
~~~~~");
```

```
    printf("\033[01;33m");

    printf("\n\n\t\t\t~~~~~ < CREATED BY >
~~~~~");
```

```
    printf("\n\t\t\t~~~~~ < MADHAV CHANDOK >
~~~~~");
```

```
    printf("\033[0m");

    sleep(2);
```

```
}
```

```
int main(void)
```

```
{
```

```
    information();
```

```
    int acc;
```

```
    int flag1,flag2,flag3,flag4;
```

```
    FILE *fp;
```

```
    if((fp=fopen("password.txt","r+"))==NULL)
```

```
{
```

```
    if((fp=fopen("password.txt","w+"))==NULL)
```

```
{  
    printf("\n\n\t\t\t CAN'T OPEN FILE!....");  
}  
}  
  
}
```

```
acc=login(fp);  
fclose(fp);  
if(acc==1)  
{  
    FILE *fp1, *fp2, *pas;  
    int option;  
  
    loading_screen();  
  
    while(1)  
    {  
        if((fp1=fopen("StudentInfo.txt","r+"))==NULL)  
        {  
            if((fp1=fopen("StudentInfo.txt","w+"))==NULL)  
            {  
                printf("\n\n\t\t\t CAN'T OPEN FILE!....");  
            }  
        }  
        if((fp2=fopen("facultyInfo.txt","r+"))==NULL)  
        {  
            if((fp2=fopen("facultyInfo.txt","w+"))==NULL)  
            {  
                printf("\n\n\t\t\t CAN'T OPEN FILE!....");  
            }  
        }  
    }  
}
```

```
}

if((pas=fopen("password.txt","r+"))==NULL)
{
    if((pas=fopen("password.txt","w+"))==NULL)
    {
        printf("\n\n\t\t\t\t CAN'T OPEN FILE!....");
    }
}
system("cls");
printHead();
printf("\e[01;36m");
printf("\n\n\t\t\t\t~~~~~ WELCOME TO THE MAIN MENU ~~~~~~");
printf("\033[01;33m");
printf("\n\n\t\t\t\t PRESS[1]-----> ADD RECORD IN DATABASE");
printf("\n\n\t\t\t\t PRESS[2]-----> DELETE RECORD IN DATABASE");
printf("\n\n\t\t\t\t PRESS[3]-----> UPDATE RECORD IN DATABASE");
printf("\n\n\t\t\t\t PRESS[4]-----> DISPLAY RECORD IN DATABASE");
printf("\n\n\t\t\t\t PRESS[5]-----> SEARCH RECORD IN DATABASE");
printf("\n\n\t\t\t\t PRESS[6]-----> UPDATE USERNAME AND PASSWORD");
printf("\n\n\t\t\t\t PRESS[7]-----> ADD NEW USERNAME AND PASSWORD");
printf("\n\n\t\t\t\t PRESS[8]-----> REMOVE USERNAME AND PASSWORD");
printf("\n\n\t\t\t\t PRESS[0]-----> EXIT FROM DATABASE SYSTEM");
printf("\033[1;32m");
printf("\n\n\t\t\t\t ENTER YOUR OPTION: ");
scanf("%d",&option);
printf("\033[0m");

switch(option)
{
```

```
case 0: thanks();  
return 0;  
break;  
  
case 1: add(fp1,fp2);  
break;  
  
case 2: flag1=1;  
while(flag1==1)  
{  
    system("cls");  
    printHead();  
    int x;  
  
    printf("\033[01;36m");  
    printf("\n\n\n\t\t\t\t~~~~~ RECORD DELETION MENU  
~~~~~");  
  
    printf("\033[01;33m");  
    printf("\n\n\n\t\t\t\t PRESS[1]----->DELETE STUDENT  
RECORD");  
  
    printf("\n\n\t\t\t\t PRESS[2]----->DELETE FACULTY MEMBER  
RECORD");  
  
    printf("\n\t\t\t\t PRESS[3]----->RETURN TO MAIN MENU");  
    printf("\033[1;32m");  
    printf("\n\n\t\t\t\t ENTER YOUR OPTION: ");  
    scanf("%d",&x);  
    printf("\033[0m");  
    switch(x)  
    {  
        case 1:fp1=delstud(fp1);  
        flag1=0;  
        break;  
        case 2:fp2=delfacul(fp2);  
        flag1=0;
```

```

        break;

    case 3:flag1=0;
        break;

default:printf("\033[01;31m");

printf("\n\n\n\t\t\t ERROR! YOU HAVE ENTERED WRONG
OPTION!");

printf("\n\n\n\t\t\t TRY AGAIN!");

printLast();

printf("\033[0m");

sleep(1);

flag1=1;
break;

}

}

break;

case 3: flag2=1;
while(flag2==1)
{
system("cls");
printHead();
int y;
printf("\033[01;36m");
printf("\n\n\n\t\t\t~~~~~ RECORD MODIFICATION MENU
~~~~~");
printf("\033[01;33m");
printf("\n\n\n\t\t\t PRESS[1]----->MODIFY STUDENT
RECORD");

printf("\n\n\t\t\t PRESS[2]----->MODIFY FACULTY MEMBER
RECORD");

printf("\n\t\t\t PRESS[3]----->RETURN TO MAIN MENU");

```



```
printf("\033[1;32m");

printf("\n\n\n\n\t\t\t ENTER YOUR OPTION: ");

scanf("%d",&y);

printf("\033[0m");

switch(y)

{

    case 1: modifystud(fp1);

        flag2=0;

        break;

    case 2: modifyfacul(fp2);

        flag2=0;

        break;

    case 3: flag2=0;

        break;

    default:printf("\033[01;31m");

        printf("\n\n\n\t\t\t ERROR! YOU HAVE ENTERED WRONG

OPTION!");

        printf("\n\n\t\t\t TRY AGAIN!");

        printLast();

        printf("\033[0m");

        sleep(1);

        flag2=1;

        break;

    case 4: flag3=1;

        while(flag3==1)

        {

            system("cls");

            printHead();
```



```
flag4=0;
break;
case 3: flag4=0;
break;
default:printf("\033[01;31m");
printf("\n\n\n\t\t\t\t ERROR! YOU HAVE ENTERED WRONG
OPTION!");
printf("\n\n\t\t\t\t TRY AGAIN!");
printLast();
printf("\033[0m");
sleep(1);
flag4=1;
break;
}
}
break;
case 6:update_user_pass(pas);
break;
case 7:pass(pas);
break;
case 8:pas=deletepass(pas);
break;
default:system("cls");
printHead();
printf("\033[1;31m");
printf("\n\n\n\t\t\t\t ERROR! YOU HAVE ENTERED WRONG OPTION!");
printf("\n\n\t\t\t\t TRY AGAIN!");
printf("\033[0m");
printLast();
sleep(2);
break;
```

```
        }

        fclose(fp1);

        fclose(fp2);

        fclose(pas);

    }

}

else

{

    system("cls");

    printHead();

    printf("\033[1;32m");

    printf("\n\n\n\t\t\t YOU HAVE ENTERED WRONG PASSWORD 3 TIMES CONSECUTIVELY!");

    sleep(3);

    printf("\033[0m");

    for(int i=5;i>0;i--)

    {

        system("cls");

        printHead();

        printf("\033[1;32m");

        printf("\n\n\n\t\t\t EXITING THE PROGRAM IN %d SECONDS.....",i);

        printf("\033[0m");

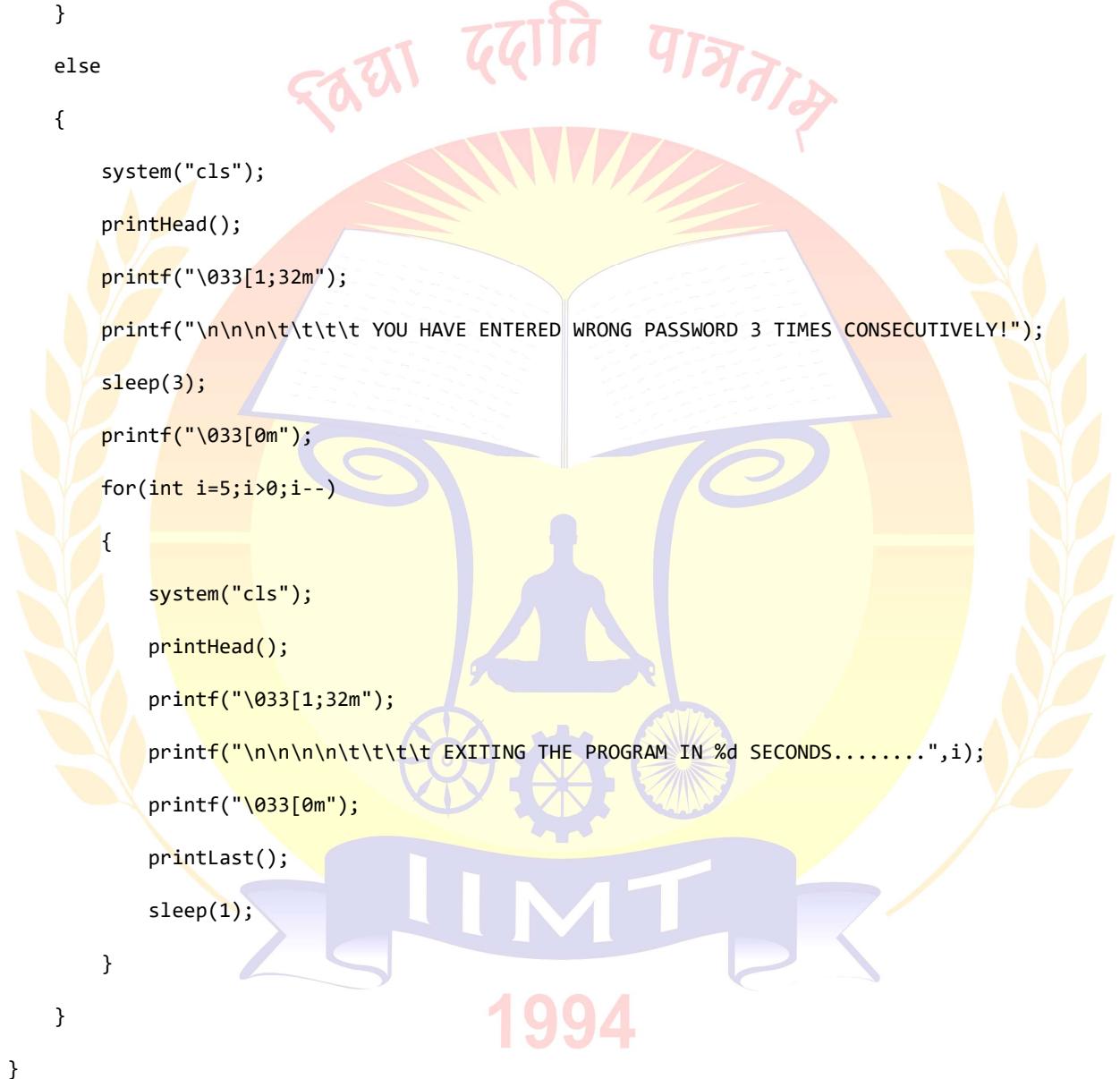
        printLast();

        sleep(1);

    }

}

}
```



6. Screenshots

```
[COLLEGE DATABASE MANAGEMENT SYSTEM]
IIMT COLLEGE OF ENGINEERING, GREATER NOIDA

~~~~~ MINI PROJECT INFORMATION ~~~~~

MINI PROJECT ON COLLEGE DATABASE MANAGEMENT SYSTEM

CREATED BY MADHAV CHANDOK

ROLL NO: 1902160130023
BRANCH AND YEAR: INFORMATION TECHNOLOGY(IT) 2ND YEAR 3RD SEMESTER
SESSION: 2020-21
COLLEGE: IIMT COLLEGE OF ENGINEERING, GREATER NOIDA - 201310
UNIVERSITY: ABDUL KALAM TECHNICAL UNIVERSITY (AKTU), LUCKNOW - 226031

Press any key to continue....
```

Fig: Information Screen

```
[COLLEGE DATABASE MANAGEMENT SYSTEM]
IIMT COLLEGE OF ENGINEERING, GREATER NOIDA

~~~~~ Login Page ~~~~~

Enter your username: IIMTCOE
Enter your password: 0216

LOGIN SUCCESSFUL!....
Press any key to continue....
```

Fig: Multi User Login System

```
=====
[COLLEGE DATABASE MANAGEMENT SYSTEM]
IIMT COLLEGE OF ENGINEERING, GREATER NOIDA
=====

~~~~~ WELCOME TO THE MAIN MENU ~~~~~

PRESS[1]-----> ADD RECORD IN DATABASE
PRESS[2]-----> DELETE RECORD IN DATABASE
PRESS[3]-----> UPDATE RECORD IN DATABASE
PRESS[4]-----> DISPLAY RECORD IN DATABASE
PRESS[5]-----> SEARCH RECORD IN DATABASE
PRESS[6]-----> UPDATE USERNAME AND PASSWORD
PRESS[7]-----> ADD NEW USERNAME AND PASSWORD
PRESS[8]-----> REMOVE USERNAME AND PASSWORD
PRESS[0]-----> EXIT FROM DATABASE SYSTEM

ENTER YOUR OPTION:
```

Fig: Main Menu

```
~~~~~ ENTER INFORMATION OF STUDENT ~~~~~

Enter Full Name: XXXXXXXX
Enter Father's Name: XXXXXXXX
Enter University Roll Number: XXXXXXXX
Enter Address: XXXXXXXX
Enter Contact: XXXXXXXXX
Enter Branch and Year: XXXXXXXX
Enter SGPA of 8 semesters

SGPA 1 semester: 9
SGPA 2 semester: 9
SGPA 3 semester: 9
SGPA 4 semester: 9
SGPA 5 semester: 9
SGPA 6 semester: 9
SGPA 7 semester: 9
SGPA 8 semester: 9

DO YOU WANT TO ENTER ANOTHER STUDENT INFORMATION....(Y/N)
```

Fig: Add Student Record

```
~~~~~ STUDENT RECORD DISPLAY PAGE ~~~~~

FULL NAME: r
FATHER'S NAME: r
UNIVERSITY ROLL NO: r
ADDRESS: r
CONTACT: r
BRANCH AND YEAR: r
SGPA: |1.00||1.00||1.00||1.00||1.00||1.00||1.00||1.00|
CGPA: 1.00

=====
=====

FULL NAME: w
FATHER'S NAME: w
```

Fig: Show all record present in database

```
[COLLEGE DATABASE MANAGEMENT SYSTEM]
IIMT COLLEGE OF ENGINEERING, GREATER NOIDA

=====
=====

RECORD DELETED FOR

FULL NAME: r
FATHER'S NAME: r
UNIVERSITY ROLL NUMBER: r
ADDRESS: r
CONTACT: r
BRANCH AND YEAR: r

Press any key to continue....
```

Fig: Record Deletion from Database

```
C:\Users\gift\Desktop\Mini Project 2nd Year\MADHAV CHANDOK 1902160130023.exe"
IIMT COLLEGE OF ENGINEERING, GREATER NOIDA
=====
~~~~~ DETAILS OF STUDENT ~~~~~
FULL NAME: w
FATHER'S NAME: w
UNIVERSITY ROLL NO: w
ADDRESS: w
CONTACT: w
BRANCH AND YEAR: w
SGPA: |1.00|2.00|34.00|34.00|6.00|7.00|8.00|8.00|
CGPA: 12.50

DO YOU WANT TO SEARCH FOR ANOTHER STUDENT RECORD....(Y/N)
=====
```

Fig: Search details in database

