A REPORT OF OPERATING SYSYTEMS LAB PROJECT ON

“STUDENT QUERY MANAGEMENT SYSTEM”

at

[ MODEL INSTITUTE OF ENGINEERING AND TECHNOLOGY]

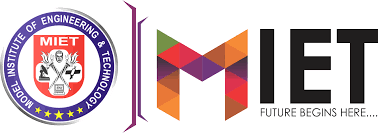
SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THE

AWARD

OF THE DEGREE

**BACHELOR OF TECHNOLOGY**

**(Computer Engineering)**

****

**SUBMITTED BY:** Akshat Amla, Ujjwal Maletha, Vishal Tickoo, Vivek Koul, Rakshit Bhari

**ROLL NO:** 2021a1r064,2021a1r068, 2021a1r070, 2021a1r080, 2021a1r084

**BRANCH:** C.S.E

**SEMESTER:** 3rd SEMESTER

ACKNOWLEDGEMENT

I would like to express my gratitude and appreciation to all those who gave me the possibility to complete this report. Special thanks is due to my Assistant Professor “Dr Swati Goel” whose help, stimulating suggestions and encouragement helped me in all time of fabrication process and in writing this report. I also sincerely thanks for the time spent proofreading and correcting my many mistakes. I would also like to acknowledge with much appreciation the crucial role of the staff in Operating system Lab, who gave me a permission to use the lab PC and other hardware devices to complete my project. Many thanks go to the all lecturer and supervisors who have given their full effort in guiding the team in achieving the goal as well as their encouragement to maintain our progress in track. My profound thanks go to all classmates, especially to my friends for spending their time in helping and giving support whenever I need it in fabricating my project

would like to express my gratitude and appreciation to all those who

gave me the possibility to complete this report. Special thanks is due to my

supervisor Mr. Hazami bin Che Hussain whose help, stimulating suggestions

and encouragement helped me in all time of fabrication process and in writing

this report. I also sincerely thanks for the time spent proofreading and correcting

my many mistakes.

I would also like to acknowledge with much appreciation the crucial role

of the staff in Mechanical Laboratory, who gave me a permission to use the lab

equipment and also the machine and to design the drawing and giving a

permission to use all the necessary tools in the laboratory.

Many thanks go to the all lecturer and supervisors who have given their

full effort in guiding the team in achieving the goal as well as their

encouragement to maintain our progress in track. My profound thanks go to all

classmates, especially to my friends for spending their time in helping and

giving support whenever I need it in fabricating my project

AKSHAT AMLA

3rd Sem

DATE:

23/12/2022

TABLE OF CONTENTS

[INTRODUCTION: 4](#_Toc122718570)

[PROJECT PROFILE: 4](#_Toc122718571)

[PROJECT DESCRIPTION: 4](#_Toc122718572)

[FLOWCHART: 5](#_Toc122718573)

[TECHNICAL DETAILS (Coding): 5](#_Toc122718574)

[OUTPUT 16](#_Toc122718575)

[BIBLIOGRAPHY 18](file:///C:\Users\gaura\Downloads\O.S%20Report.docx#_Toc122718576)

PROJECT SUMMARY

# INTRODUCTION:

The Round-robin scheduling algorithm is a kind of pre-emptive First come First Serve CPU Scheduling algorithm where each process in the ready state gets the CPU in a cyclic way (turn by turn) for a certain time slice defined in the system which is called time quantum. It is the oldest scheduling algorithm, which is mainly used for multitasking**.**

# PROJECT PROFILE:

Project title: ”STUDENT/FACYLTY QUERY HANDLING SYSTEM ”

College: MIET,Jammu   
Platform Language: C

Guide: Dr. Swati Goel

Dept.: Computer Science

MIET,Jammu(Autonomous)

SUBMITTED BY: Akshat Amla(2021a1r064)

Ujjwal Maletha(2021a1r068)

Vishal Tickoo(2021a1r070)

Vivek Koul(2021a1r080)

Rakshit Bhari(2021a1r084)

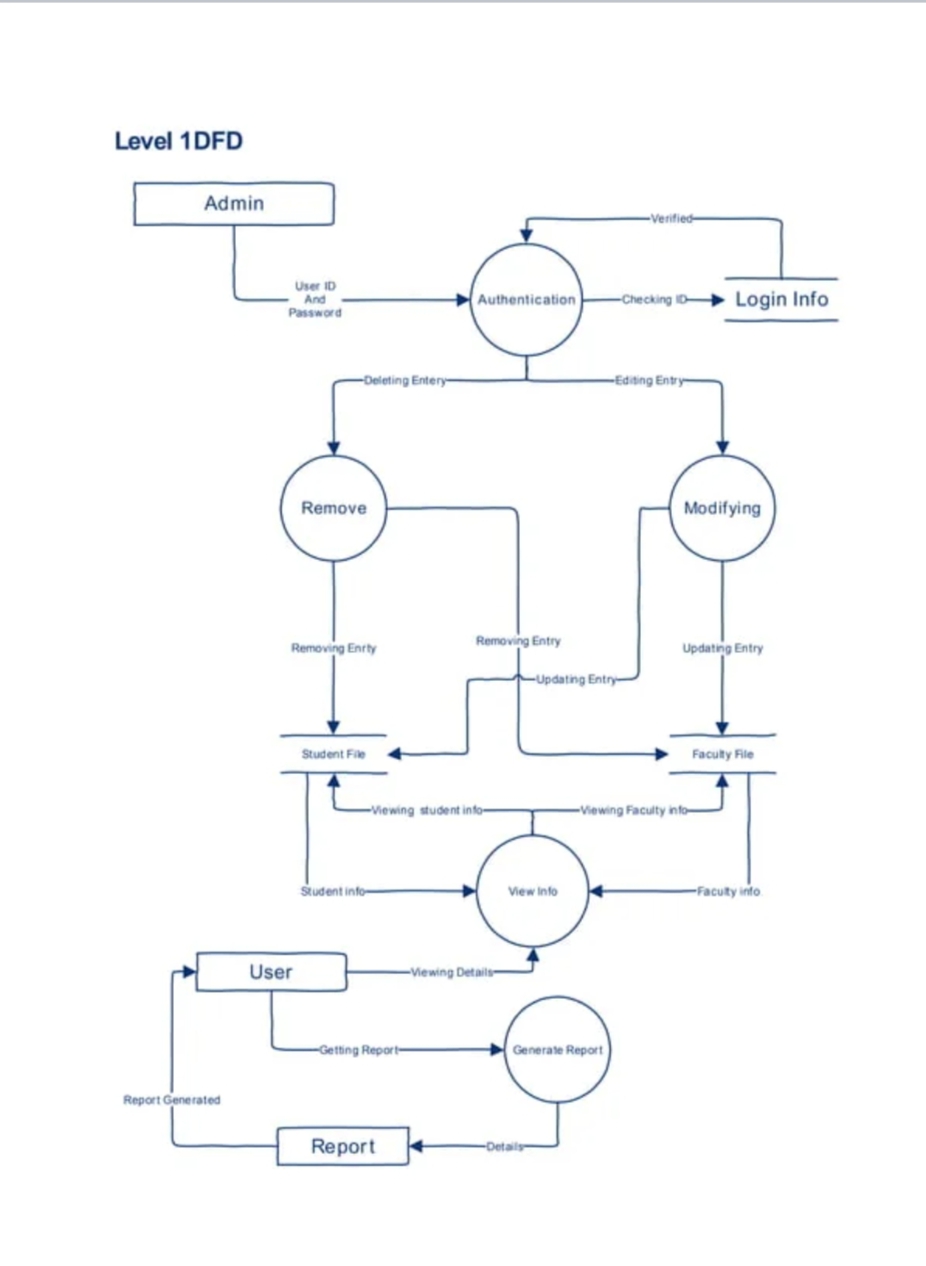
B.E 3rd Semester

MIET Jammu (Autonomous)

# PROJECT DESCRIPTION:

Design an online system which can handle student queries. Since there can be multiple requests at any time he wishes to dedicate a fixed amount of time to every request so that everyone gets a fair share of his time. He will log into the system from 10am to 12am only. He wants to have separate requests queues for students and faculty. Implement a strategy for the same. The summary at the end of the session should include the total time he spent on handling queries and average query time.

# FLOWCHART:



# 

# TECHNICAL DETAILS (Coding):

#include<stdio.h>

#include<conio.h>

#include<stdlib.h>

#include<windows.h>

#include<string.h>

void Admin\_login();

void print\_queries();

void Raise\_Query();

void Resolve\_Query();

void Resolved\_Queries();

void gotoxy(int ,int );

void menu();

void add();

void view();

void search();

void modify();

void deleterec();

struct student

{

char name[20];

char mobile[10];

int rollno;

char course[20];

char branch[20];

};

int main()

{

system("cls");

gotoxy(45,5);

system("color 74");

printf("<--:Student Record Management System:-->");

gotoxy(1,30);

printf("Press any key to continue.");

getch();

menu();

return 0;

}

void Raise\_Query(){

char name[30],query[50];

int choice=1;

FILE \*fp = fopen("Query.txt","a+");

while(choice){

printf("\nWhat is your name?");

fflush(stdin);

gets(name);

printf("Write your query: ");

fflush(stdin);

gets(query);

fprintf(fp,"%s : %s\n",name,query);

printf("\n\nWant to enter more queries?(1/0): ");

scanf("%d",&choice);

}

fclose(fp);

getch();

exit(0);

}

void Resolve\_Query(){

char Solution[100],Read\_Query[100];

FILE \* fp1 = fopen("Query.txt","r");

FILE \* fp2 = fopen("Solution.txt","a+");

int i=0;

while(!feof(fp1)){

fgets(Read\_Query,100,fp1);

printf("\n\n%d query is: %s",i+1,Read\_Query);

printf("\nWrite down your reply......\n");

fflush(stdin);

gets(Solution);

fprintf(fp2,"%s\n",Solution);

i++;

}

fclose(fp1);

fclose(fp2);

}

void Resolved\_Queries(){

FILE \* fp = fopen("Solution.txt","r");

char Read[100];

while(!feof(fp)){

fgets(Read,100,fp);

printf("%s",Read);

}

fclose(fp);

getch();

}

void print\_queries() {

system("cls");

FILE \* fp = fopen("Query.txt","r");

char str[100];

while(!feof(fp)){

fgets(str,100,fp);

printf("%s\n",str);

}

fclose(fp);

printf("\n\tAll the queries have been printed Successfully......");

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

getch();

}

void menu()

{

int choice,n;

system("cls");

system("color 30");

gotoxy(30,15);

printf("Want to login as a Faculty or Student???? ");

gotoxy(30,18);

printf("Enter 1 for Faculty and 0 otherwise.....");

scanf("%d",&n);

if(n==1){

Admin\_login();

system("color 56");

system("cls");

gotoxy(10,3);

printf("<--:MENU:-->");

gotoxy(10,5);

printf("Enter appropriate number to perform following task.");

gotoxy(10,7);

printf("1 : Add Record.");

gotoxy(10,8);

printf("2 : View Record.");

gotoxy(10,9);

printf("3 : Search Record.");

gotoxy(10,10);

printf("4 : Modify Record.");

gotoxy(10,11);

printf("5 : Delete.");

gotoxy(10,12);

printf("6 : Print Queries.");

gotoxy(10,13);

printf("7 : Resolve query");

gotoxy(10,14);

printf("8 : Exit.");

gotoxy(10,17);

printf("Enter your choice.");

scanf("%d",&choice);

switch(choice)

{

case 1:

add();

break;

case 2:

view();

break;

case 3:

search();

break;

case 4:

modify();

break;

case 5:

deleterec();

break;

case 6:

print\_queries();

break;

case 7:

Resolve\_Query();

break;

case 8: exit(0);

break;

default:

gotoxy(10,17);

printf("Invalid Choice.");

}

}

else{

system("cls");

system("color 56");

printf("\n\n\t\t\tWELCOME TO STUDENT PORTAL\n\n");

printf("\n\n<--:MENU:-->");

gotoxy(10,5);

printf("\n\nEnter appropriate number to perform following task.");

gotoxy(10,7);

printf("\n\n1 . SEARCH RECORD");

gotoxy(10,10);

printf("\n\n2. TO RAISE A QUERY");

printf("\n\n3. VIEW RESOLVED QUERIES.");

printf("\n\n4. EXIT.");

printf("\n\nEnter your choice.");

scanf("%d",&choice);

switch(choice)

{

case 1:

search();

break;

case 2:

Raise\_Query();

break;

case 3: Resolved\_Queries();

break;

case 4: exit(0);

break;

default:

gotoxy(10,17);

printf("Invalid Choice.");

}

}

}

void Admin\_login()//DEFINITION OF FUNCTION Admin\_login()

{

char c,A[15],U[15];//DECLARATION OF ARRAYS(A,U) & CHARACTER c

int i=0;

static int count=0;//STATIC VARIABLE RETAINS ITS VALUE THROUGHOUT THE PROGRAM.

system("pause");//FOR HOLDING OUTPUT SCREEN

system("cls");

printf("\n\n\n\n\t\t\t\n\t\t\t\t\t<------WELCOME TO ADMIN SECTION------->\n\n");

system("color 80");

printf("\t\t\t\tUSER ID: ");//

fflush(stdin);//IT IS USED TO CLEAR THE BUFFER

gets(U);//USED TO INPUT STRING

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

while((c=getch())!=13) //LOGIC TO READ PASSWORD IN A SECURED WAY

{

A[i]=c;

i++;

printf("\*");

}

A[i]='\0';//INSERTING NULL AT THE END OF THE STRING

if(strcmp(U,"1234")==0 && strcmp(A,"1234")==0)//COMPARRING WHETHER USERNAME AND PASSWORD IS SAME OR NOT

{

printf("\n\t\t\t\tLOGIN SUCCESSFUL! ");

getch();

return;

}

else//THIS BLOCK WILL EXECUTE IF PASSWORD OR USERNAME IS WRONG!

{

printf("\n\n\t\t\t\tINVALID USERNAME OR PASSWORD!...\n");

count++;

system("pause");

if(count>3)

exit(1);

Admin\_login();

}

}

void add()

{

FILE \*fp;

struct student std;

char another ='y';

system("cls");

fp = fopen("record.txt","ab+");

if(fp == NULL){

gotoxy(10,5);

printf("Error opening file");

exit(1);

}

fflush(stdin);

while(another == 'y')

{

gotoxy(10,3);

printf("<--:ADD RECORD:-->");

gotoxy(10,5);

printf("Enter details of student.");

gotoxy(10,7);

printf("Enter Name : ");

gets(std.name);

gotoxy(10,8);

printf("Enter Mobile Number : ");

gets(std.mobile);

gotoxy(10,9);

printf("Enter Roll No : ");

scanf("%d",&std.rollno);

fflush(stdin);

gotoxy(10,10);

printf("Enter Course : ");

gets(std.course);

gotoxy(10,11);

printf("Enter Branch : ");

gets(std.branch);

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

gotoxy(10,15);

printf("Want to add of another record? Then press 'y' else 'n'.");

fflush(stdin);

another = getch();

system("cls");

fflush(stdin);

}

fclose(fp);

gotoxy(10,18);

printf("Press any key to continue.");

getch();

menu();

}

void view()

{

FILE \*fp;

int i=1,j;

struct student std;

system("cls");

gotoxy(10,3);

printf("<--:VIEW RECORD:-->");

gotoxy(10,5);

printf("S.No Name of Student Mobile No Roll No Course Branch");

gotoxy(10,6);

printf("--------------------------------------------------------------------");

fp = fopen("record.txt","rb+");

if(fp == NULL){

gotoxy(10,8);

printf("Error opening file.");

exit(1);

}

j=8;

while(fread(&std,sizeof(std),1,fp) == 1){

gotoxy(10,j);

printf("%-7d%-22s%-12s%-9d%-12s%-12s",i,std.name,std.mobile,std.rollno,std.course,std.branch);

i++;

j++;

}

fclose(fp);

gotoxy(10,j+3);

printf("Press any key to continue.");

getch();

menu();

}

void search()

{

FILE \*fp;

struct student std;

char stname[20];

system("cls");

gotoxy(10,3);

printf("<--:SEARCH RECORD:-->");

gotoxy(10,5);

printf("Enter name of student : ");

fflush(stdin);

gets(stname);

fp = fopen("record.txt","rb+");

if(fp == NULL){

gotoxy(10,6);

printf("Error opening file");

exit(1);

}

while(fread(&std,sizeof(std),1,fp ) == 1){

if(strcmp(stname,std.name) == 0){

gotoxy(10,8);

printf("Name : %s",std.name);

gotoxy(10,9);

printf("Mobile Number : %s",std.mobile);

gotoxy(10,10);

printf("Roll No : %d",std.rollno);

gotoxy(10,11);

printf("Course : %s",std.course);

gotoxy(10,12);

printf("Branch : %s",std.branch);

}

}

fclose(fp);

gotoxy(10,16);

printf("Press any key to continue.");

getch();

menu();

}

void modify()

{

char stname[20];

FILE \*fp;

struct student std;

system("cls");

gotoxy(10,3);

printf("<--:MODIFY RECORD:-->");

gotoxy(10,5);

printf("Enter name of student to modify: ");

fflush(stdin);

gets(stname);

fp = fopen("record.txt","rb+");

if(fp == NULL){

gotoxy(10,6);

printf("Error opening file");

exit(1);

}

rewind(fp);

fflush(stdin);

while(fread(&std,sizeof(std),1,fp) == 1)

{

if(strcmp(stname,std.name) == 0){

gotoxy(10,7);

printf("Enter name: ");

gets(std.name);

gotoxy(10,8);

printf("Enter mobile number : ");

gets(std.mobile);

gotoxy(10,9);

printf("Enter roll no : ");

scanf("%d",&std.rollno);

gotoxy(10,10);

printf("Enter Course : ");

fflush(stdin);

gets(std.course);

gotoxy(10,11);

printf("Enter Branch : ");

fflush(stdin);

gets(std.branch);

fseek(fp ,-sizeof(std),SEEK\_CUR);

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

break;

}

}

fclose(fp);

gotoxy(10,16);

printf("Press any key to continue.");

getch();

menu();

}

void deleterec()

{

char stname[20];

FILE \*fp,\*ft;

struct student std;

system("cls");

gotoxy(10,3);

printf("<--:DELETE RECORD:-->");

gotoxy(10,5);

printf("Enter name of student to delete record : ");

fflush(stdin);

gets(stname);

fp = fopen("record.txt","rb+");

if(fp == NULL){

gotoxy(10,6);

printf("Error opening file");

exit(1);

}

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

if(ft == NULL){

gotoxy(10,6);

printf("Error opening file");

exit(1);

}

while(fread(&std,sizeof(std),1,fp) == 1){

if(strcmp(stname,std.name)!=0)

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

}

fclose(fp);

fclose(ft);

remove("record.txt");

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

gotoxy(10,10);

printf("Press any key to continue.");

getch();

menu();

}

void gotoxy(int x,int y)

{

COORD c;

c.X=x;

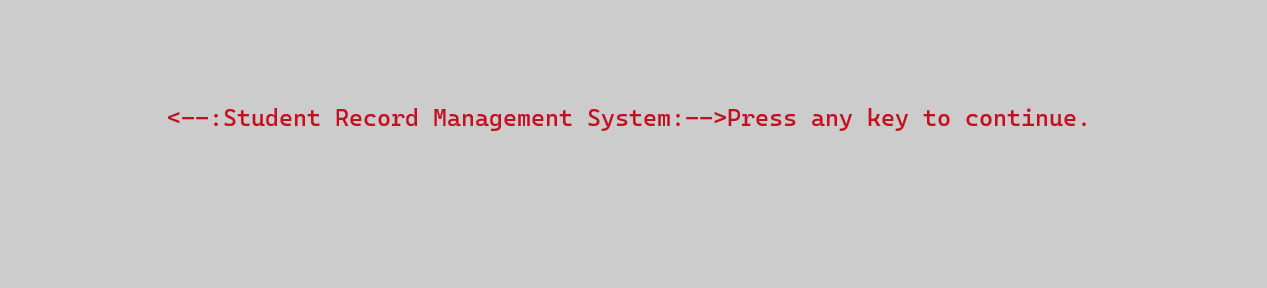
c.Y=y;

SetConsoleCursorPosition(GetStdHandle(STD\_OUTPUT\_HANDLE),c);

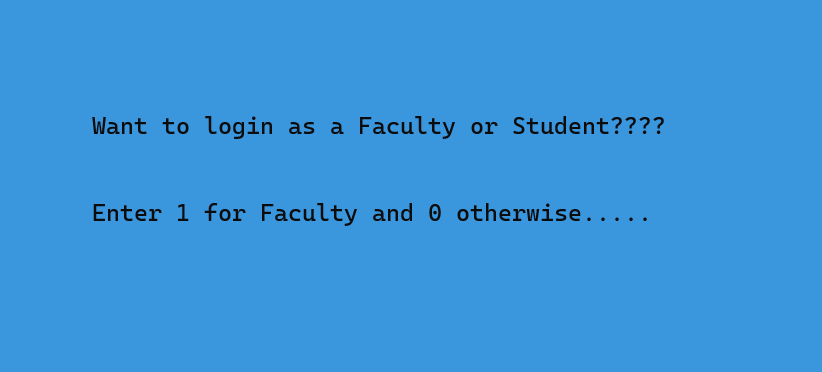
}

## OUTPUT

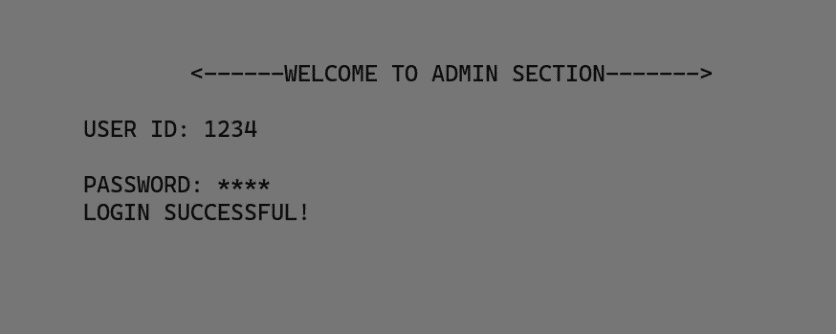
Home screen :

****

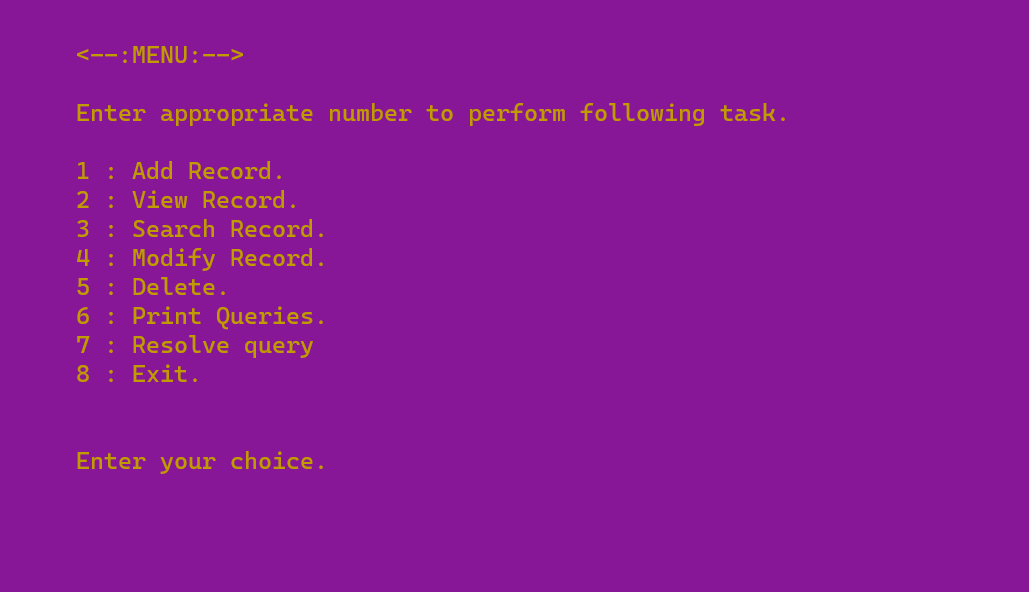
Asking for whether faculty is loging in or student:

****

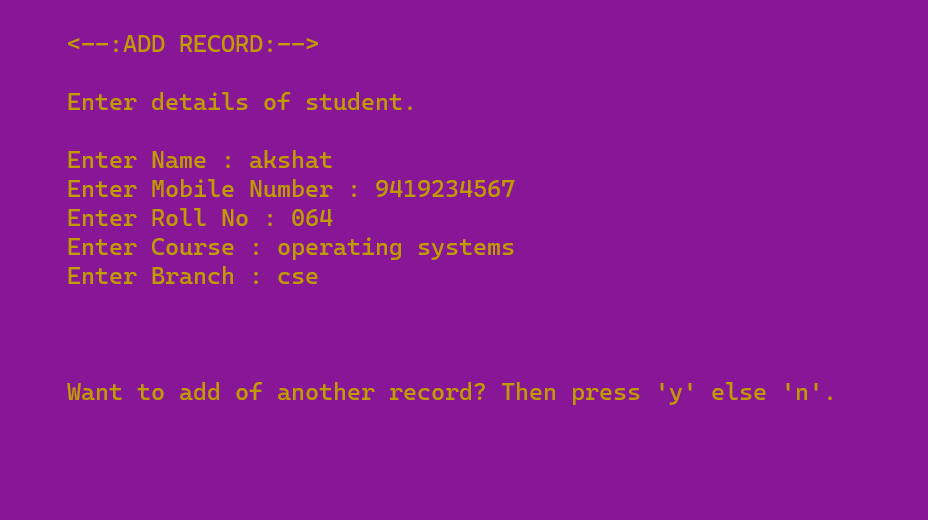
Authentication screen:

****

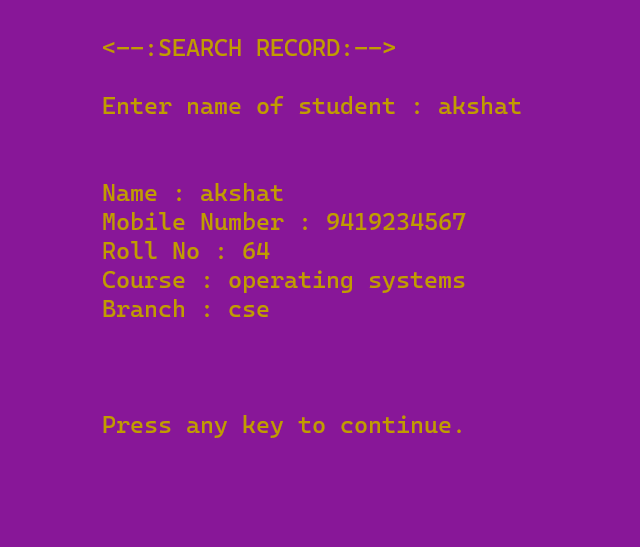
Faculty Menu:

****

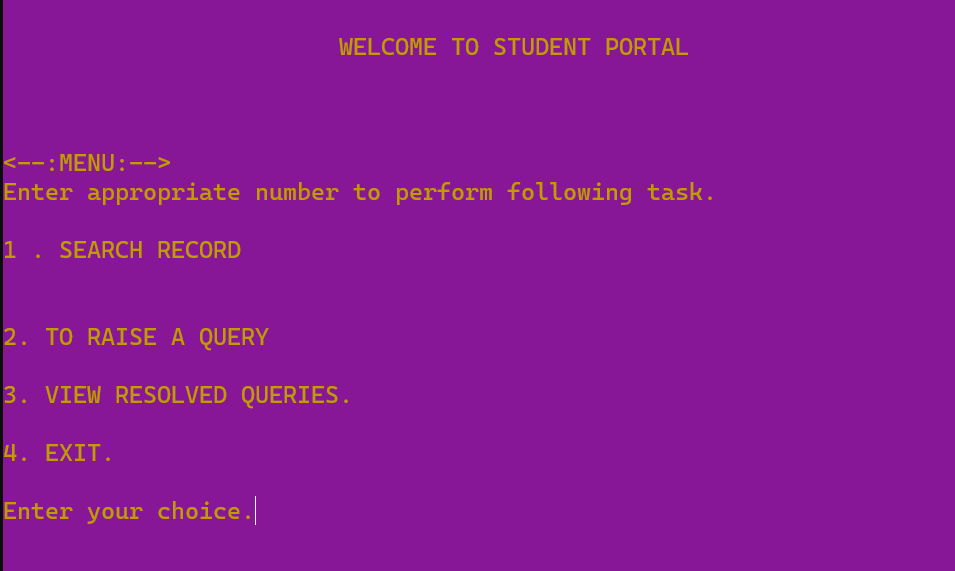
Adding Records:

****

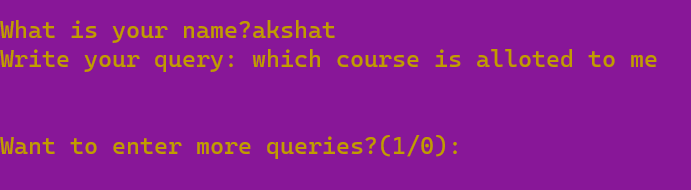
Searching Records:

****

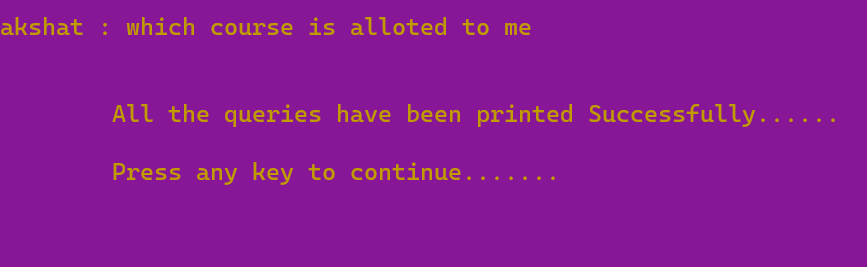
Student Portal:

****

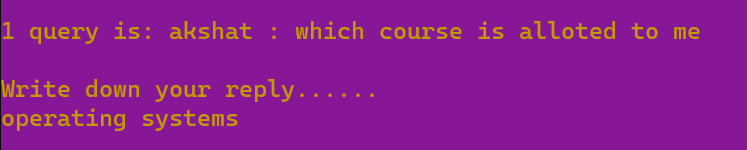
Raising Query:

****

Printing Query:

****

Resolving Query:

****

# BIBLIOGRAPHY

BOOK:

* Operating System Concepts

~By P.B Galvin

* Operating Systems

~By Archer J Harris

REFERENCES:

* <https://www.geeksforgeeks.org/>
* <https://www.javatpoint.com/>
* <https://www.tutorialspoint.com/index.htm>
* <https://www.gatevidyalay.com/>