



THAPATHALI CAMPUS
Institute of Engineering

A PROJECT REPORT

ON

ENTRANCE MANGEMENT SYSTEM

IN PARTIAL FULLFILLMENT OF THE REQUIREMENTS FOR THE
COURSE OF CT 401 COMPUTER PROGRAMMING

BACHELOR OF ELECTRONICS, COMMUNICATION AND
INFORMATION ENGINEERING

SUBMITTED BY

ARJUN CHHETRI THA075BEI007

NIRAJ DUWAL THA075BEI 028

SURAJ AGRAHARI THA075BEI043

SUBMITTED TO

DEPARTMENT OF ELECTRONICS AND COMPUTER
ENGINEERING ,THAPATHALI CAMPUS

March 7, 2019

DECLARATION

We hereby declare that the project work report entitled “ ENTRANCE MANGEMENT SYSTEM” submitted for the partial fulfillment of the requirements for the course of CT 401 Computer Programming is our original work and the Project Work Report has not formed the basis for the award of any degree, diploma, or other similar titles.

Signature

Name of Student: ARJUN CHHETRI

Signature

Name of Student: NIRAJ DUWAL

Signature

Name of Student: SURAJ AGRAHARI

ACKNOWLEDGEMENTS

Without encouragement and inspiration, it is very difficult to build anything. A suitable method for entrance examination was felt to be necessary from a long time although there are a lot of method to face this entrance examination. our project “ENTRANCE MANAGEMENT SYSTEM” is one of the most reliable, secure and fastest method to fulfill user needs. it is based on traditional platform and question along with flooring modernized technique to digitalized the educational system.

Its main inspiration is taken from IOE entrance examination. I, Suraj agrahari, along with other two friends who had recently faced entrance examination have thoroughly discussed about this topic. drawback of ioe exam, some special feature with proper presentation is done which provides some extend relief to eyes. Special support is taken from ‘learning c by examples’, ‘secret of c’ and website of ioe entrance examination(www.ioe.edu.np). Special thanks to Er. Rama bastola for proper guidance, positive feedback, suggestion and encouragement.

At last but not the least we will like to thank our parents (The living god) who toil hard and their prayer invisibly saves us from any problem.

ABSTRACT

Entrance Management system is a program which solves the practical life problem related with entrance in different level of education. The program is designed for the student who want to upgrade their level of study by joining in bachelor especially. It is the most systematic, well managed, easiest, safe, convenient, trustable and quick process to fulfill the main objective and obtain the is necessary fruitful result steps in simple.

This program mainly deals with storing students record via form, taking examination in order to judge the students' knowledge, publish students results and suggests them the college according to their rank.

Key Words:

ID, Mark, dob, name, rank

TABLE OF CONTENTS

DECLARATION.....	1
ACKNOWLEDGEMENTS	2
ABSTRACT	3
LIST OF FIGURES	6
LIST OF TABLES	7
CHAPTER 1: INTRODUCTION	8
1.1 Background	8
1.2 Problem Statement.....	9
1.3 Objectives	9
1.4 Applications	9
1.5 Project Features	10
1.6 Feasibility Analysis	11
1.6.1 Economic Feasibility	11
1.6.2 Technical Feasibility	11
1.6.3 Operational Feasibility	11
1.7 System Requirement.....	11
1.7.1 Software Requirement	11
1.7.2 Hardware Requirement.....	12
CHAPTER 2: LITERATURE REVIEW	13
2.1 Entrance management system.....	13
2.1.1 form.....	13
2.1.2 Modify	13
2.1.3 Entrance examination.....	13

2.1.4 Display student record.....	14
2.1.5 Search record.....	14
2.1.6 Rank of student.....	14
CHAPTER 3: DESIGN & METHODOLOGY.....	15
3.1 Entrance management system.....	15
3.1.1 form.....	15
3.1.2 Modify	16
3.1.3 Entrance examination	16
3.1.4 Display student record.....	17
3.1.5 Seach record.....	17
3.1.6 Rank of student.....	17
CHAPTER 4: IMPLEMENTATION AND RESULT	20
4.1 Implementation Detail	20
4.2 Result Analysis.....	20
CHAPTER 5: CONCLUSION & FURTHER WORK.....	23
5.1 Conclusion	23
5.2 Further Works	23
REFERENCES.....	24
APPENDICES.....	25

LIST OF FIGURES

FIGURE	PAGE
Figure 3.1: Block diagram of the project-----	15
Figure 3.2: Block diagram for form-----	16
Figure 4: System flow diagram-----	18

LIST OF TABLES

TABLE	PAGE
Table 3.1.6: Ranking table-----	18
Table 1.2: -----	3
Table 2.1: -----	7

CHAPTER 1: INTRODUCTION

1.1 Background

In this 21st century with the increase in the pace of development in the technological field. Nepal is also heading toward the development in the same way especially in the educational field. Nepal is introducing new technique (example grading system, technical training etc.) for better technical development. with the increase in the need of technically trained, sharp minded, logical thinker human resource entrance has become the backbone for many universities and college. In our Nepal there is bad tradition of nepotism, baptism, politics to overcome this problem this problem entrance can be an effective solution.

There are many reasons justify the idea of making universities mandatory for all students to take an entrance exam. First, an entrance exam can determine the student's tendencies towards the collage or career they pursuit. There are many students applying to a wrong university or career because of their parents' desire, friends' decision, or financial aid and scholarships offered by such a university. If a student really likes to enter a certain university, he'll try his best and study really hard to fulfill his wish. In addition, an entrance exam can help the university choose the best students where they get scholarships from it and from governments, companies... so that they can spend their time in studying or innovations to be the best to help our world. Finally, an entrance exam can balance and improve the work and environment. It can also push the students into being more serious about their career and future. It can never hinder their dreams even though many feel it's not fair for below average students. They might think they're not going to pass to get into the university of their dream, but a little more determination will drive these thoughts right away

In this program, entrance and its record student's student from +2 and diploma background can show their knowledge level through entrance examination. students who are really dedicated to this to their respective career option will show their maximum effort to face this challenge. Traditionally Nepal's education government is taking paper examination for almost every subject (one of expectation is entering) which is tedious, requires a large budget, man dory power, time consuming, maximum

error process where there is the chance of corruption. after discussing with many teachers and students who are taking and giving entrance the idea to develop this process has been developed. this program helps us to fill online form so we don't need to stand on queue, questions set up automatically so we don't need to worry about question leak, fast and no chance of corruption.

1.2 Problem Statement

The main problem of our country at present is that qualified students are not chosen for qualitative education instead they are chosen randomly and a few by their family name or background. now a day there are a few schools, college and university that choose students by the method of entrance, but in most cases it is manual (i.e. by hands and on paper) it is difficult to conduct such exam, manage answer sheets, check them and sort them. As well as publishing questions in such a huge quantity is expensive and there are most chances of getting leakage of these papers and similarly answer sheet may be lost. So by seeing our present context of our country we designed this program to conduct examination and select only qualified students regardless of their background.

1.3 Objectives

Its main objective is enlightened below:

- I. To store the data of candidate begin based on their forms record.
- II. To select the qualified candidate for the universities, college and job.
- III. To minimize the error and corruption for selection and maximize the comfort, accuracy and speed to publish the result.

1.4 Applications

This entrance and its record program can be applicable in different educational and professional field to select the best candidate for respective post or category which are listed below:

I. Engineering

II. MBBS Examination.

III. Ag and forestry

IV. BBS and BBA

V. Army

VI. PCL

VII. Job interview

VIII. Loksewa

IX. MCQ exam like lab exam.

1.5 Project Features

I. It publishes the form which collects the information of candidate and their validity.

II. It stores the data of candidate for future use.

III. one can modify his/her detail before entrance examination.

IV. student can withdrawal his/her name if s/he doesn't like to give exam by deleting his /her name

V. It take the entrance of 20 minute having full marks 10.

VI. It displays 10 question each having 1 marks along with 10% negative marking per each question

VIII. Special feature once the question is submitted cannot it cannot be recorrected.

IX. special feature it displays the marks scored just after the submission of paper.

X. it displays the rank wise result of candidate when the entrance examination of all candidate is completed.

1.6 Feasibility Analysis

1.6.1 Economic Feasibility

Economic feasibility analysis is the most commonly used method for determining the efficiency of a new project. It is also known as cost analysis. It helps in identifying profit against investment expected from a project. Cost and time are the most essential factors involved in this field of study. This program is economically highly feasible program snice it is written in c programming language in c compiler i.e. code block so economically it is highly satisfying with minimum time consumption

1.6.2 Technical Feasibility

Technical feasibility study is the complete study of the project in terms of input, processes, output, fields, programs and procedures. It is a very effective tool for long term planning and trouble shooting. The technical feasibility study should most

essentially support the financial information of an organization. It is technically well managed program whose source code can be easily modified for better improvement and for debugging any error.

1.6.3 Operational Feasibility

Operational feasibility refers to the measure of solving problems with the help of a new proposed system. It helps in taking advantage of the opportunities and fulfills the requirements as identified during the development of the project. It takes care that the management and the users support the project. it is very easy to use simply a person who understand English language can use this program. data given to this program can be easily resurrect while filling form also at the time of exam operation is very easy concept of how to tick and submit is required.

1.7 System Requirement

1.7.1 Software Requirement

This program does not require a lot of software .it can perform pretty well by using code block or DEV-C++ as its programming language is c.

1.7.2 Hardware Requirement

This program is economically feasible program so it doesn't require a lot of hardware similarly hardware required is a laptop. if laptop is unavailable then the hardware required is desktop, CPU, and mouse. In Simple laptop of i3 intel core, having 2 gb with low or no graphic or desktop having low ram about 400mb can be easily used to run this program.

CHAPTER 2: LITERATURE REVIEW

2.1 Entrance Management System.

Literally entrance means act of entering somewhere or door of anything. entrance and its result is program which carries a great literal meaning, it is mainly designed for student. it helps them to achieve their goal to fulfill their dreams. literally it symbolizes that educational institute are like a dreams of the students which they can fulfill through this entrance examination. in this program candidate can fight for their dreams that store the record and publish as a sweet fruit to them to their toil. This program can be operated in small budge unlike present IOE examination which needs networking, a large amount of budge

2.1.1 Form

Form is the one of the most beginning, primary and initial stage of any program which collect the essential information for future use.in this program form is like a backbone of the program. Unless or until the form is filled other process cannot be beginning. In form general information of the candidate like name, address, contact address is taken which helps the examiner to check their validity.

2.1.2 Modify

Modify is the user friendly portion in which candidate can modify their personal which they have mistakenly entered wrong while filling form. In this portion one has to enter their full details once again which increase it accuracy level because while filling the same form for second time one will be more careful.

2.1.3 Entrance examination

Education sector has been seen growing phenomenally across the globe and its prospects in Nepal too is very bright some student begin pursuing the engineering course soon after 10th as diploma and some candidate take up the route after 12th for bachelor in both the cases entrance plays a vital role.in context of our country Nepal nowadays the entrance is flooring in each and every places for example like engineering, MBBS etc.

2.1.4 Display student record

In this portion of program, one can check their full detail which he/she has entered while filling form. In this part all the information of all the student who have involve in this entrance examination is included .one can check this her marks obtain after giving entrance exam.

2.1.5 Search

Search is a service given by this program in which one can just check their full details like name, date of birth, home, phone number and marks obtain just by entering their identification document number.it makes user or any person who is trying to seek the information of other people easier to allocate that person without any hindrance.

2.1.6 Rank of student

Result is the consequence of any problem or last stage of any program. It is the final where the time to spread the pursuit of happiness who toil hard to achieve the goal and their luck favors too. In this part marks scored by student along with their rank is displayed viewing their roll number so only the authorized person can know marks and rank.

CHAPTER 3: DESIGN & METHODOLOGY

3.1 Entrance Management:

Entrance Management is the name of our project. At the beginning our project it asks the details of the applicant and if the conditions are met to get admitted in the college then only the applicant is allowed to fill the form. After filling the form examination is taken and the result isn't obtained manually instead it is processed by the project itself, the result/ranking of the student is published in ascending order. Based on their ranking they are divided in different colleges for their higher studies. The block diagram (i.e.: fig. 3.1) below shows the details of our project.

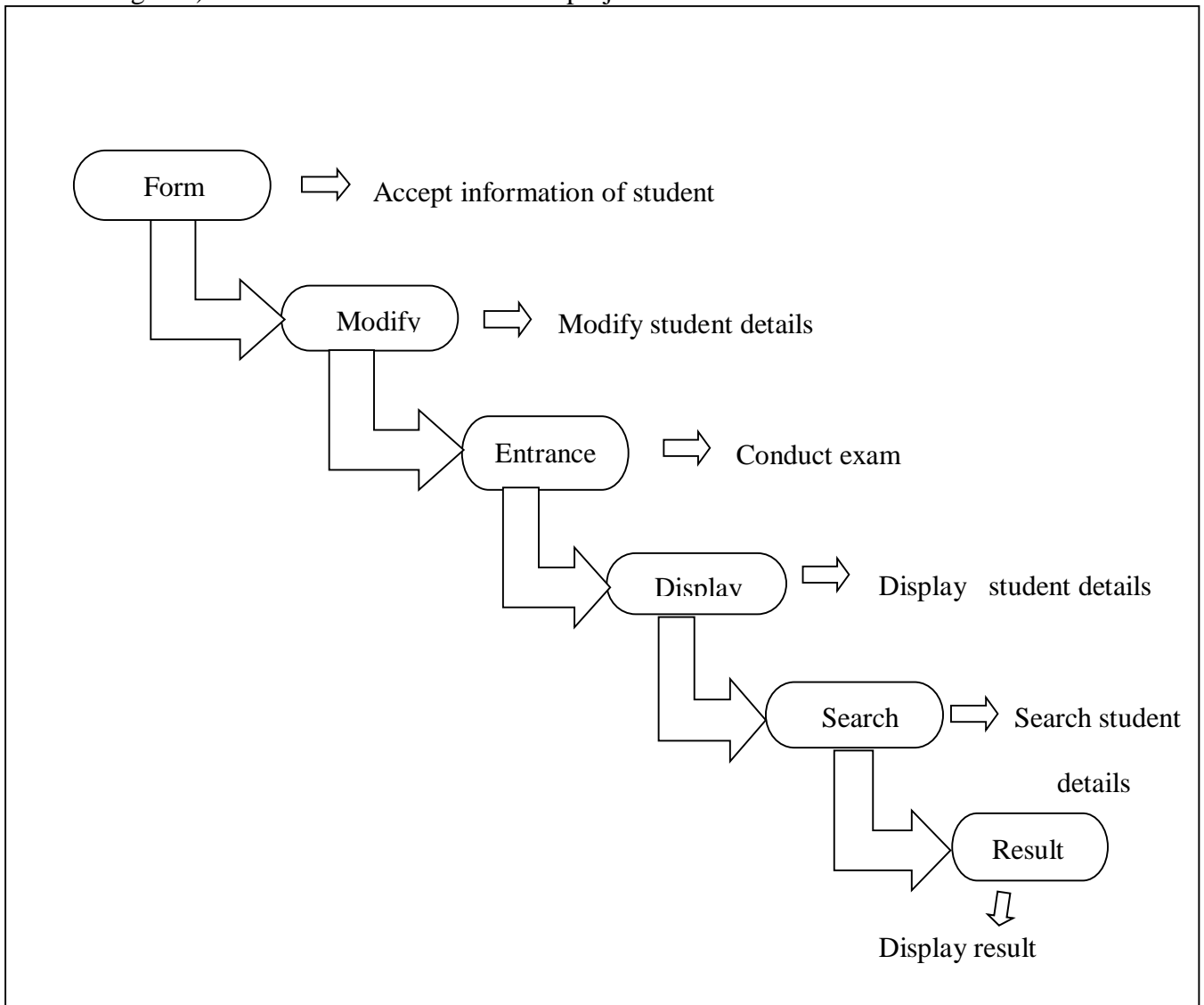


Fig 3.1: block diagram of the project

3.1.1 Form:

Form is an important part of the project and the initial step of the project. In this part of our project student's information's such as name, address, document identification no., contact no etc. is asked and is stored. If any unsuitable condition is met in the form of student, then the form is terminated and the student won't be allowed to take the examination. The block diagram (i.e. fig 3.2) below explains the parts of the form.

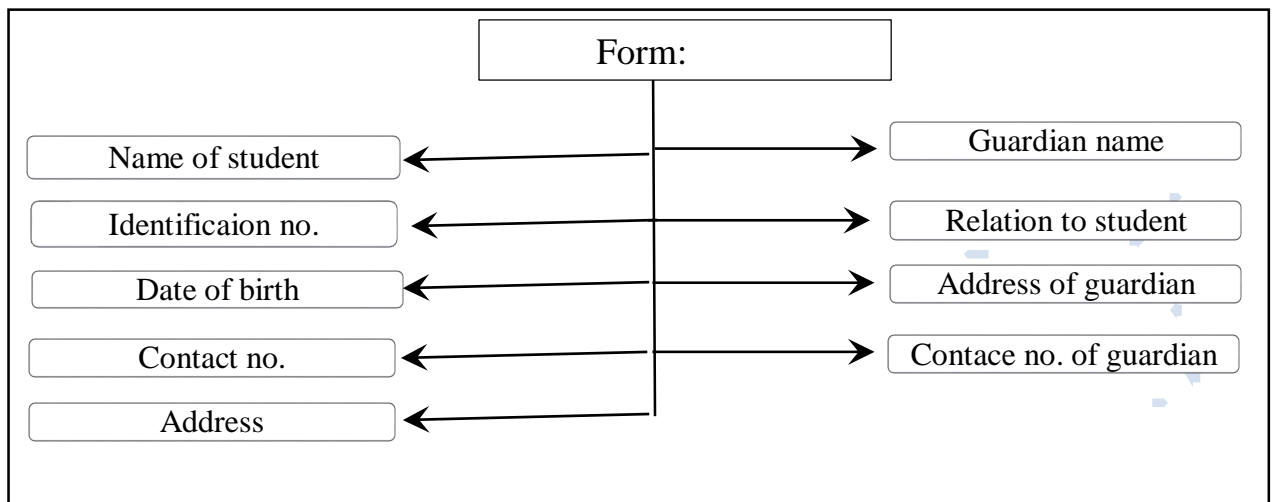


Fig 3.2: Block diagram for form

3.1.2 Modify:

It is the second part of our project. In this part of project, the student who filled the form can modify their information if he/she found mistake in their details. For this you have to enter your identification number that you entered while form filling and have to pay Rs 500. Then you are allowed to enter your new details, i.e. name, address, document identification no., contact no etc.

3.1.3 Entrance:

It is the third and most important part of project. In this part of project, the students who filled the form are allowed to give take the examination. The exam is of 10 marks for

10 questions and has to be filled in 20 minutes. In the question paper there are 10 questions of 1 mark. If any student found cheating, she/he is dismissed from the exam. Students can submit the paper any time and the paper is submitted automatically after 20 minutes. The question once attempted cannot be reappeared again.

3.1.4 Display Record:

It is the displaying part of our project. Student who filled the form are display in this part. If the student attempt entrance exam the marks will display as he/she scored otherwise marks will be displayed 0.00. Thus it is a displaying part so we couldn't enter anything, it is the reading part of our project, which display all the student details who filled up the form.

3.1.5 Search Record:

It is the fifth part of our project. This part help student to find their details by entering their identification number. In display record it displays all the student record so it is time consuming for student to find their details. Search record easily find the student details only by entering their identification number.

3.1.6 Rank of student:

It is the last and final part of the project. Only those students who attempt entrance exam will be displayed in result. It displays identification number including total marks and rank. Rank will be displayed by comparing their obtained marks.

* List of the student who passed in entrance examination including with total marks and rank.

S. N	Identification No:	Total Marks	Rank
1	290173014	8.8	1
2	290175115	7.2	2
3	290174165	5.4	3
4			
5			

Table 3.1.6: Ranking table

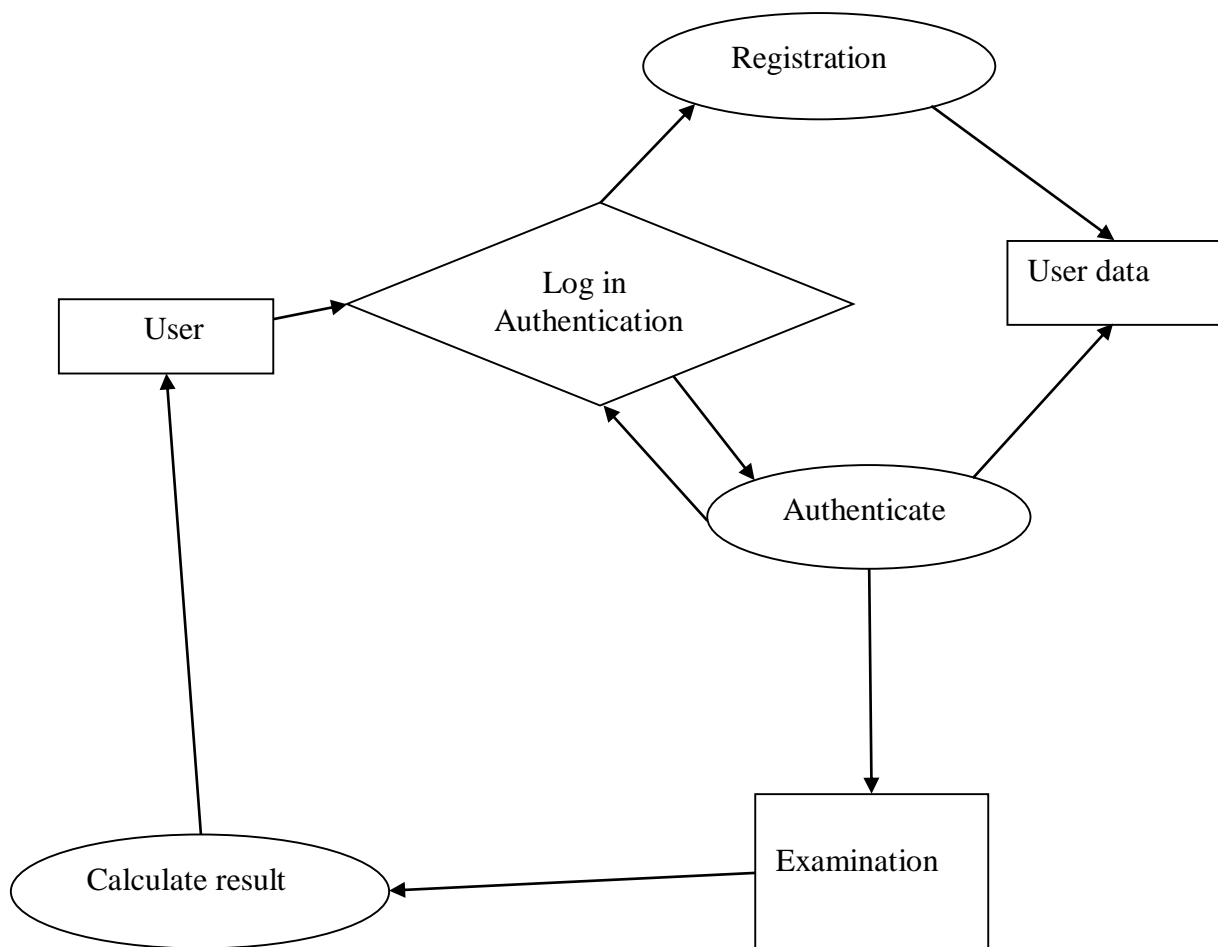


Fig 4: System-Flow Diagram

CHAPTER 4: IMPLEMENTATION AND RESULT

4.1 Implementation

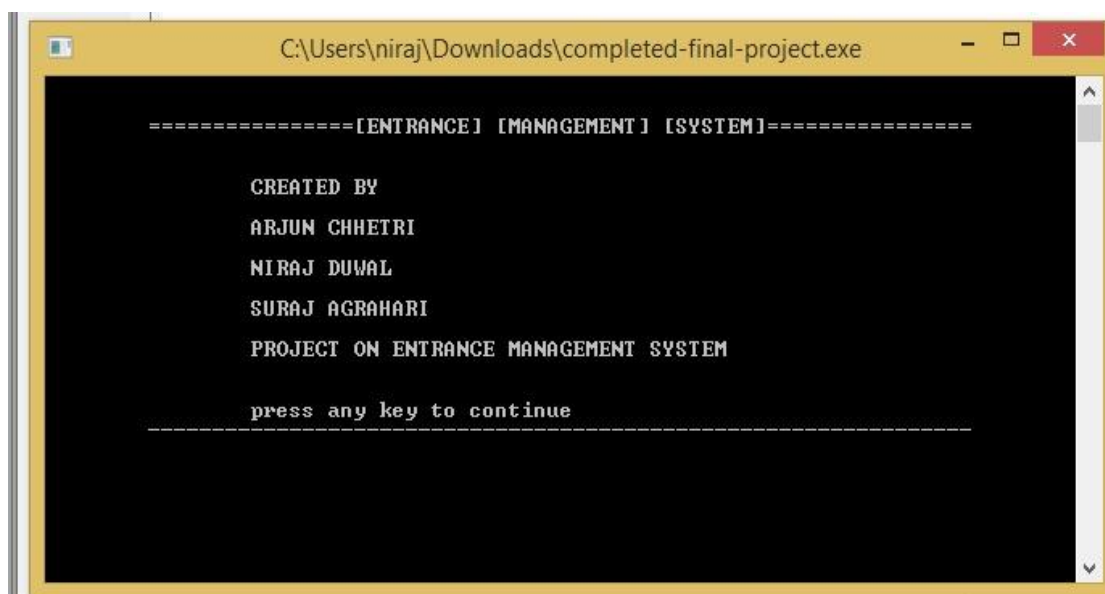
4.1.1 Introduction:

Implementation is the stage of the project when the theoretical design is turned out into a working system. Thus, it can be considered to be the most critical stage in achieving a successful new system and in giving the user, confidence that the new system will work and be effective.

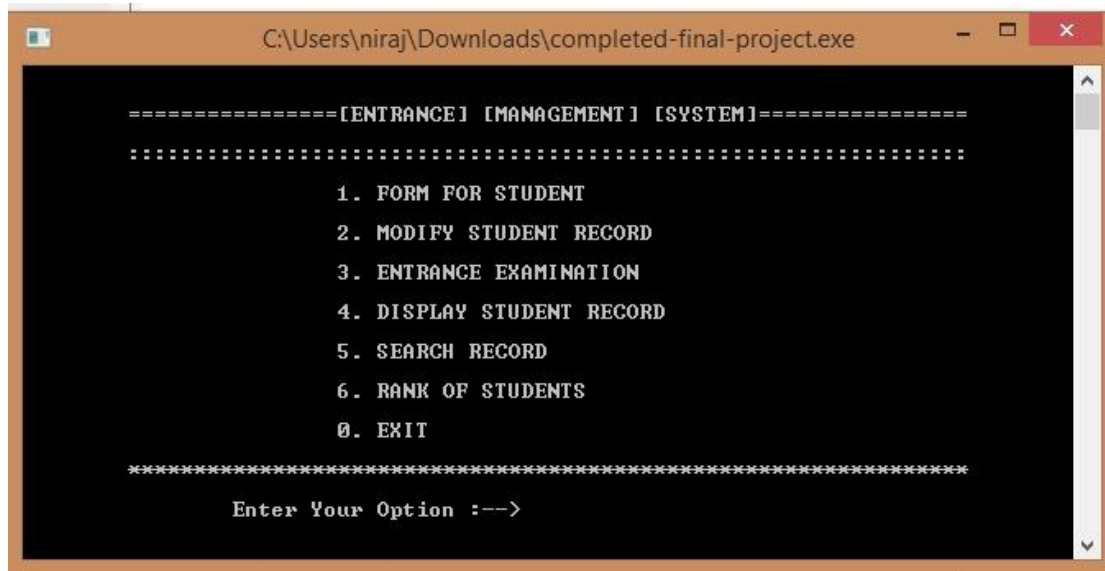
The implementation stage involves careful planning, investigation of the existing system and its constraints on implementation, designing of methods to achieve changeover and evaluation of changeover methods.

Some screenshots:

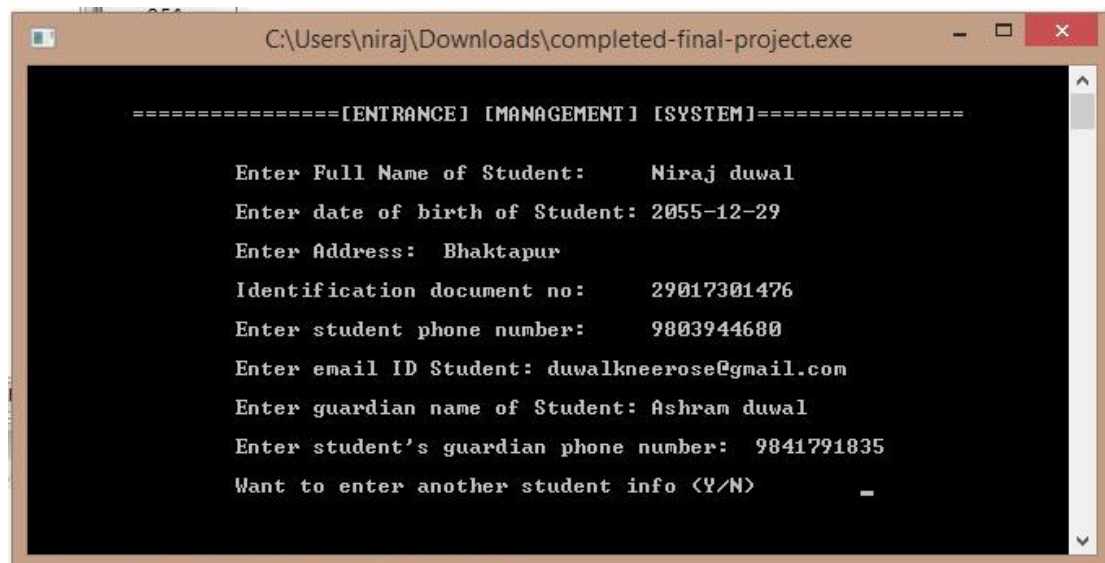
FIRST DISPLAY



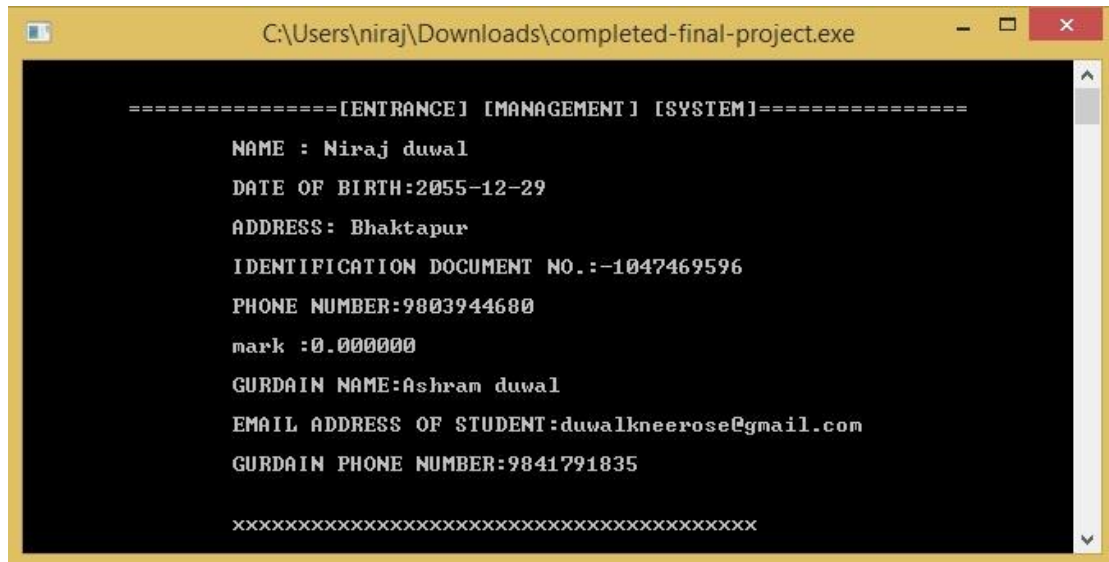
MAIN SCREEN



FORM FOR STUDENT



DISPLAY STUDENT RECORD



4.2 Result Analysis

It is the last and one of the most important part whose main goal is to display the result, capacity and validity .In the part of project , rank of student along with their marks is displayed . marks and rank is displayed via identification document number beside this one can access to their all data including mark in display part of exam in mass in list. Also one can glance his\her data individually in search portion

CHAPTER 5: CONCLUSION & FURTHER WORK

5.1 Conclusion

Since we are taking exam of student electronically in the” Entrance Management System”, data will be secured. Using this application, we can retrieve student information with a single click. Thus, processing information will be faster. It guarantees accurate maintenance of student details. It easily reduces the paper question task and thus reduces the time for calculation, ranking, form filling and increases accuracy speed.

5.2 Further Works

Many different adaptations, tests, and experiments have been left for the future due to lack of time (i.e. the experiments with real data are usually very time consuming, requiring even days to finish a single run). Future work concerns deeper analysis of particular mechanisms, new proposals to try different methods, or simply curiosity. we can add more questions as we like and marking according to the question. Due to lack of time we add only 10 questions. Our intention is to add 140 question, where 60 questions are short carrying 1 mark each and 40 are long question carrying 2 marks each.

REFERENCES

- [1] Braian w. Kernighan and dennis M.ritchie, “The C Programming Language”, Second Edition.
- [2] Robort lafore , “ The waite groups C programming using turbo C++” second Edition
- [3] Yashavant kanetkar , “let us C” ,seventh edition
- [4] AI Kelly and Ira pohl, “A Book on C”, fourth edition
- [5] [http://www.wikipedia.org/wiki/literature review](http://www.wikipedia.org/wiki/literature_review), 2019 Jan 26
- [6] <http://www.ioe.edu.np>
- [7] Daya Sagar Baral, Duwakar Baral, Sharad Kumar Ghumire, “The Secrets of C Programming Language”

APPENDICES

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

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

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

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

```
#include<stdbool.h>
```

```
#include<time.h>
```

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

```
#include<ctype.h>
```

```
#define Student struct Stud
```

```
void add (FILE * fp); //to add to list
```

```
void exam (FILE *fp) ;//for entrance examination
```

```
void show_record ();
```

```
void modify (FILE * fp) ;//to modify a record
```

```
void displayList(FILE * fp);//display whole list
```

```
void searchRecord(FILE *fp);//find a particular record
```

```
void printChar(char ch,int n);//printing a character ch n times
```

```
void printHead();//printing head line for each screen
```

```
void sort(FILE *fp);//sorting
```

```
struct Stud
```

```
{
```

```
    char name[100];
```

```
    char dob[20];
```

```
    char address[50];
```

```
    int ID;
```

```
    float Mark;
```

```
    int n;
```

```
    char ph[20];
```

```
    char gname[100];
```

```
    char email[100];
```

```
    char gph[20];
```

```
};
```

```
int main()
```

```
{
```

```
    FILE * fp;
```

```
Student s;

int option;

char another;

if((fp=fopen("D:\\student.txt","rb+"))==NULL)

{

    if((fp=fopen("D:\\student.txt","wb+"))==NULL)

    {

        printf("can't open file");

        return 0;

    }

}

printHead();

printf("\n\n\t\tCREATED BY");

printf("\n\n\t\tARJUN CHHETRI");

printf("\n\n\t\tNIRAJ DUWAL");

printf("\n\n\t\tSURAJ AGRAHARI");

printf("\n\n\t\tPROJECT ON ENTRANCE MANAGEMENT SYSTEM");

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

```

printChar('-',64);

getch();

while(1)

{

    printHead();

    printf("\n\t");

    printChar(':',64);

    printf("\n\n\t\t1. FORM FOR STUDENT");//ADD

    printf("\n\n\t\t2. MODIFY STUDENT RECORD");

    printf("\n\n\t\t3. ENTRANCE EXAMINATION");//MODIFY

    printf("\n\n\t\t4. DISPLAY STUDENT RECORD");//DISPLAY

    printf("\n\n\t\t5. SEARCH RECORD");// RECORD

    printf("\n\n\t\t6. RANK OF STUDENTS");//SORTING OF RANK OF
STUDENT

    printf("\n\n\t\t0. EXIT");//TERMINATE THE PROGRAM

    printf("\n\t");

    printf("\n\t");

    printChar('*',64);

    printf("\n\n\t\tEnter Your Option :--> ");

```

```
scanf("%d",&option);

switch(option)

{

    case 0: return 1;

        break;

    case 1: add(fp);

        break;

    case 2: modify(fp);

        break;

    case 3: exam(fp);

        break;

    case 4: displayList(fp);

        break;

    case 5: searchRecord(fp);

        break;

    case 6: sort(fp);

        break;

    default: printf("\n\t\tYou Pressed wrong key");
```

```

        printf("\n\t\tProgram terminated");

        getch();

        exit(0)

    }

}

return 1;

}

//----printing character ch at n times -----

void printChar(char ch,int n)

{

    while(n--)

    {

        putchar(ch);

    }

}

//-----Printing Head Line of the program -----

void printHead()

{ system("cls");

```



```

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

printChar('=',16);

printf("[ENTRANCE] [MANAGEMENT] [SYSTEM]");

printChar('=',16);

printf("\n");

}

// =====sorting=====

void sort(FILE * fp)

{

    printHead();

    Student s;

    int i,b,j,troll,siz=sizeof(s),a=0;

    float tmark;

    struct sort

    {

        int roll;

        float mark;

```

```

}rank[a];

rewind(fp);

while((fread(&s,siz,1,fp))==1)

{

    a+=1;

}

rewind(fp);

b=0;

while((fread(&s,siz,1,fp))==1)

{

    rank[b].roll=s.ID;

    rank[b].mark=s.Mark;

    b+=1;

}

for (i=0;i<a;i++)

{

    for(j=0;j<a-1;j++)

    {

```

```

        if (rank[j].mark<rank[j+1].mark)

        {

            tmark=rank[j].mark;

            rank[j].mark=rank[j+1].mark;

            rank[j+1].mark=tmark;


            troll=rank[j].roll;

            rank[j].roll=rank[j+1].roll;

            rank[j+1].roll=troll;

        }

    }

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

    printChar('*',64);

    printf("\n\n\t\tRanking\t\t\tRoll No.\t\tMarks\n");

    for (i=0;i<a;i++)

    {

        printf("\n\t\t%d\t\t%d\t\t%f",i+1,rank[i].roll,rank[i].mark);

```

```

    }

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

    printChar('~',64);

    getch();

}

// =====FILLING STUDENT FORM=====

void add(FILE * fp)

{

    printHead();

    char another='y';

    Student s;

    fseek(fp,0,SEEK_END);

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

    printf("\t\t");

    printChar('~',85);

    puts("\n\t\t*POINTS TO REMEMBER WHILE FILLING FORM");

    puts("\n\t\ti. One should pay proper attention while filling the form.");

    puts("\n\t\tii. phone number and identification document number should be
numeric (<10) i.e character");

```

```

puts("\n\t\t from 0 to 9.");

puts("\n\t\tiii.one should remember their identification document number as it
is the key factor");

puts("\n\t\t for other steps ");

puts("\n\t\tiv. one can modify their form record in modify portion with plenty
of rs 500");

puts("\n\t\tv. identification document number should be last exam symbol
number or ");

puts("\n\t\t citizenship number without any special sign(eg-./_) ");

printf("\t\t");

printChar('~',85);

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

getch();

printHead();

while(another=='y'||another=='Y')

{

printf("\n\n\t\tEnter Full Name of Student:\t");

fflush(stdin);

fgets(s.name,100,stdin);

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

```

```
printf("\n\n\t\tEnter date of birth of Student:\t");
```

```
fflush(stdin);
```

```
fgets(s.dob,20,stdin);
```

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

```
printf("\n\n\t\tEnter Address:\t");
```

```
fflush(stdin);
```

```
fgets(s.address,50,stdin);
```

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

```
printf("\n\n\t\tIdentification document no: \t");
```

```
scanf("%d",&s.ID);
```

```
s.Mark=0;
```

```
s.n=0;
```

```
printf("\n\n\t\tEnter student phone number:\t");
```

```
fflush(stdin);
```

```
fgets(s.ph,20,stdin);
```

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

```
printf("\n\n\t\tEnter email ID Student:\t");
```

```

fflush(stdin);

fgets(s.email,100,stdin);

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

printf("\n\n\tEnter guardian name of Student:\t");

fflush(stdin);

fgets(s.gname,100,stdin);

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

printf("\n\n\tEnter student's guardian phone number:\t");

fflush(stdin);

fgets(s.gph,20,stdin);

s.gph[strlen(s.gph)]='\0';

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

printf("\n\n\tWant to enter another student info (Y/N)\t");

fflush(stdin);

another=getche();

}

}

```

//=====ENTRANCE EXAMINATION=====

```
void exam(FILE *fp)

{

    printHead();

    int tempRoll,flag,siz,r;

    Student s;

    float countr;

    char another,cha;

    siz=sizeof(s);

    fseek(fp,0,SEEK_END);

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

    printf("\t\t");

    printChar('~',85);

    puts("\n\t\t*POINTS TO REMEMBER WHILE GIVING EXAMINATION ");

    puts("\n\t\ti. One should enter their identification document number to began the
entrance exam.");

    puts("\n\t\tii. Entrance is taken for 20 minute having total 10 marks.");

    puts("\n\t\tiii. Examinee can skip any question by pressing's' ");

    puts("\n\t\tiv. Once skipped questions cant be reattempted.");
```



```

puts("\n\t\tv. It consist of 10 question each having 1 marks.");

puts("\n\t\tvi. There is the negative marking of 10% marks per each wrong
answer.");

puts("\n\t\tvii. Once the answer is submitted cannot be recorrected.");

puts("\n\t\tviii.Total score is displayed when all the answer are submitted.");

printf("\t\t");

printChar('~',85);

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

getch();

printHead();

printf("\n\n\t\t Do you want to take exam (Y/N)\t");

fflush(stdin);

another=getch();

if (another=='y' || another=='Y')

{

printf("\n\n\t\tEnter Identification Number of Student: ");

scanf("%d",&tempRoll);

rewind(fp);

```

```

while((fread(&s,siz,1,fp))==1)

{

if(s.ID==tempRoll&& s.n==0)

    {

        flag=1;

        break;

    }

}

if (flag==1)

    { countr=0.0;

    int i,m;

    bool arr[10]={0};

    int A[10];

    // srand(time(NULL));

    time_t t;

    srand((unsigned)time(&t));

    for (i=0;i<10;i++)

        {int r = rand()%10;

```

```

    if(!arr[r])

    A[i]= r;

    else

    i--;

    arr[r]=1;

    }

for(i=0;i<10;i++)

    {

        //system("cls");

        // A[i]=i;

    switch(A[i])

    {

        case 1:

        a1:

        system("cls");

        printf("\n\n\tQ.The word ‘engineer’ has its primary stress on its
        _____ syllable:\n");

        printf("\n\n\t(A) first \t\t(B) second \n\t(C) third \t\t(D) fourth");

        //ans c

```

```

cha=getch();

if (cha=='c' || cha=='C')

{

    countr++;

    break;

}           else

if(cha=='a' || cha=='A' || cha=='b' || cha=='B' || cha=='d' || cha=='D')

{

    countr-=.1;

    break;

} else if(cha=='s' || cha=='S')

{

    countr=countr+0;

    break;

}

else{

    printf("\n\n\n\n\t\t\t<error in choosing option:>");

    getch();

    goto a1;

```

```
}
```

case 2:

```
system("cls");
```

a2:

```
printf("\n\n\n\tQ.Two and two _____ four.");
```

```
printf("\n\n\n\t(A) makes \t\t\t(B) make \n\t(C) have made \t\t\t(D) will  
be made ");
```

```
// ans a
```

```
cha=getch();
```

```
if (cha=='a' || cha=='A')
```

```
{
```

```
    countr++;
```

```
    break;
```

```
}
```

```
else if(cha=='s' || cha=='S')
```

```
{
```

```
    countr=countr+0;
```

```
    break;
```

```
}
```

```

else
if(cha=='c' || cha=='C' || cha=='b' || cha=='B' || cha=='d' || cha=='D')

{

    countr-=.1;

    break;

}

else{

    printf("\n\n\n\t\t\t<error in choosing option:>");

    getch();

    goto a2;

}

case 3:

system("cls");

a3:

printf("\n\n\n\t\tQ.pH of 0.2 N H2SO4 is:");

printf("\n\n\n\t(A) 0.69 \t\t(B) 1.2 \n\t(C) 0.76 \t\t(D) 0.56");

// ans a

cha=getch();

if (cha=='a' || cha=='A')

```

```

        {

            countr++;

            break;

        }        else
if(cha=='c' || cha=='C' || cha=='b' || cha=='B' || cha=='d' || cha=='D')

        {

            countr-=.1;

            break;

        }

else if(cha=='s' || cha=='S')

        {

            countr=countr+0;

            break;

        }

else{

            printf("\n\n\n\n\t\t\t<error in choosing option:>");

            getch();

            goto a3;

        }

```

case 4:

```
system("cls");
```

a4:

```
printf("\n\n\n\tQ.Catalytic oxidation of ammonia in air forms:");
```

```
printf("\n\n\n\t(A)NO2 \t\t\t(B) N2O5 \n\t\t(C) N2O \t\t(D) NO");
```

```
//ans d
```

```
cha=getch();
```

```
if (cha=='d' || cha=='D')
```

```
{
```

```
    countr++;
```

```
    break;
```

```
}
```

```
else if(cha=='s' || cha=='S')
```

```
{
```

```
    countr=countr+0;
```

```
    break;
```

```
}
```

```
else
```

```
if(cha=='a' || cha=='A' || cha=='b' || cha=='B' || cha=='c' || cha=='C')
```

```
{
```



```

        countr-=.1;

        break;

    }

    else{

        printf("\n\n\n\n\t\t\t<error in choosing option:>");

        getch();

        goto a4;

    }

    case 5:

        system("cls");

        a5:

        printf("\n\n\n\t\tQ.The magnetism of the magnet is due to:");

        printf("\n\n\n\t\t(A) the spin motion of electron \n\t\t(B) cosmic ray
\n\t\t(C) the earth \n\t\t(D) pressure of big magnet inside the earth ");

        //ans a

        cha=getch();

        if (cha=='a'||cha=='A')

        {

            countr++;

```

```

        break;

    }else if(cha=='s'||cha=='S')

    {

        countr=countr+0;

        break;

    }

    else

if(cha=='d'||cha=='D'||cha=='b'||cha=='B'||cha=='c'||cha=='C')

    {

        countr-=.1;

        break;

    }

else{

    printf("\n\n\n\n\t\t\t<error in choosing option:>");

    getch();

    goto a5;

}

```

case 6:

```
system("cls");
```

a6:

```
printf("\n\n\n\tQ.The maximum percentage of ingredients in cement is  
that of:");
```

```
printf("\n\n\t(A) magnesia \t\t(B) lime \n\t(C) alumina \t\t(D) silica");
```

```
//ans b
```

```
cha=getch();
```

```
if (cha=='b'||cha=='B')
```

```
{
```

```
    countr++;
```

```
    break;
```

```
}
```

```
else if(cha=='s'||cha=='S')
```

```
{
```

```
    countr=countr+0;
```

```
    break;
```

```
}
```

```

else
if(cha=='a' || cha=='A' || cha=='d' || cha=='D' || cha=='c' || cha=='C')

{

    countr-=.1;

    break;

}

else{

    printf("\n\n\n\n\t\t\t<error in choosing option:>");

    getch();

    goto a6;

}

case 7:

system("cls");

a7:

printf("\n\n\n\tQ.The second derivative of  $f(x) = 1/x$  at point (1,1) is
equal to:");

printf("\n\n\t(A) 1 \t\t\t(B) -1 \n\t(C) 2 \t\t\t(D) -2");

//ans c

cha=getch();

```

```

if (cha=='c' || cha=='C')

{

    countr++;

    break;

}

else if(cha=='s' || cha=='S')

{

    countr=countr+0;

    break;

}

else

if(cha=='a' || cha=='A' || cha=='b' || cha=='B' || cha=='d' || cha=='D')

{

    countr-=.1;

    break;

}

else{

    printf("\n\n\n\n\t\t\t<error in choosing option:>");

    getch();

```

```

        goto a7;

    }

case 8:

system("cls");

a8:

printf("\n\n\n\tQ.Which of the following is a logic gates? ");

printf("\n\n\tA.PUT\t\tB.THEN\n\tC.NOR\t\tD.WHEN\n");

//ans c

cha=getch();

if (cha=='c' || cha=='C')

{

    countr++;

    break;

}

else if(cha=='s' || cha=='S')

{

    countr=countr+0;

    break;

```

```

    }

    else

if(cha=='a'||cha=='A'||cha=='b'||cha=='B'||cha=='d'||cha=='D')

    {

        countr-=.1;

        break;

    }

else{

    printf("\n\n\n\n\t\t\t<error in choosing option:>");

    getch();

    goto a8;

}

case 9:

system("cls");

a9:

    printf("\n\n\n\tQ.:If the line  $2x+3y+4+k(-x+y+5)=0$  is horizontal then
the value of k is:");

    printf("\n\n\t(A) 0 \t\t\t(B) 1 \n\t(C) 3 \t\t\t(D) 2");

//ans d

```

```

cha=getch();

if (cha=='d' || cha=='D')

{

    countr++;

    break;

}

else if(cha=='s' || cha=='S')

{

    countr=countr+0;

    break;

}

else

if(cha=='a' || cha=='A' || cha=='b' || cha=='B' || cha=='c' || cha=='C')

{

    countr-=.1;

    break;

}

else{

    printf("\n\n\n\n\t\t\t<error in choosing option:>");

```



```

        getch();

        goto a9;

    }

    case 10:

        system("cls");

        a10:

        printf("\n\n\t\tQ.:IUPAC name of CH3-CH2 - CH2 -CH (OCH3) - CO
Br is:");

        printf("\n\t(A) 2- methoxypentanoyl bromide \t\t\t(B) 3-
methoxypentanoyl bromide \n\t\t\t\t(C) 3-methoxyhexanoyl bromide \t\t(D) 2-
methoxyhexanoyl bromide");

        //ans a

        cha=getch();

        if (cha=='a' || cha=='A')

        {

            countr++;

            break;

        }

        else if(cha=='s' || cha=='S')

        {

```

```

        countr=countr+0;

        break;

    }

    else

if(cha=='a' || cha=='A' || cha=='b' || cha=='B' || cha=='c' || cha=='C')

    {

        countr-=.1;

        break;

    }

    else{

        printf("\n\n\n\n\t\t\t<error in choosing option:>");

        getch();

        goto a10;

    }

} //s.ID;

}

fflush(stdin) //to hold marks

rewind(fp);

while((fread(&s,siz,1,fp))==1)

```

```

{

    if(s.ID==tempRoll)

        {

            flag=1;

            break;

        }

}

if(flag==1)

{

    fseek(fp,-siz,SEEK_CUR);

    s.Mark=countr;

    s.n=1;                                     //to check exam given or not

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

}

system("cls");

printf("\n\n\n\n\t\tCongratulations %s you scored=%f\n",s.name,s.Mark);

printf("\t\t\tBest of luck for result\n");

printf("\t\t\tpress any key to go homepage\t");

```

```

fflush(stdin);

}

else

{

    printf("\n\t\t EITHER YOU HAVENOT REGISTER YET OR YOU HAVE
ALREADY GIVEN EXAM\n");

}

getch();

}

}

//=====MODIFY A RECORD =====

void modify(FILE * fp)

{

    printHead();

    Student s;

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

    float cgpa;

    printf("\n\n\t Precaution:one shouldn't enter a string character while entering ID
number");

```

```
printf("\n\n\tEnter identification document Number of Student to MODIFY the  
Record : ");
```

```
scanf("%d",&tempRoll);
```

```
rewind(fp);
```

```
while((fread(&s,siz,1,fp))==1)
```

```
{
```

```
    if(s.ID==tempRoll)
```

```
    { flag=1;
```

```
      break;
```

```
    }
```

```
}
```

```
if(flag==1)
```

```
{
```

```
    fseek(fp,-siz,SEEK_CUR);
```

```
    printf("\n\n\tRECORD FOUND");
```

```
    printf("\n\n\tEnter New Data for the Record");
```

```
    printf("\n\n\tEnter Full Name of Student\t");
```

```
    fflush(stdin);
```

```
    fgets(s.name,100,stdin);
```

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

printf("\n\n\t\tEnter date of birth of Student\t");

fflush(stdin);

fgets(s.dob,20,stdin);

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

printf("\n\n\t\tEnter Address\t");

fflush(stdin);

fgets(s.address,50,stdin);

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

printf("\n\n\t\tEnter student phone number\t");

fflush(stdin);

fgets(s.ph,20,stdin);

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

printf("\n\n\t\tEnter email ID Student\t");

fflush(stdin);

fgets(s.email,100,stdin);

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

```

printf("\n\n\tEnter guardian name of Student\t");

fflush(stdin);

fgets(s.gname,100,stdin);

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

printf("\n\n\tEnter student's guardian phone number\t");

fflush(stdin);

fgets(s.gph,20,stdin);

s.gph[strlen(s.gph)]='\0';

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

}

else

{

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

}

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

system("pause");

}

```

```
//=====DISPLAY THE LIST =====
```

```
void displayList(FILE * fp)
```

```
{ printHead();
```

```
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\t\tDATE OF BIRTH:%s",s.dob);
```

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

```
printf("\n\t\tIDENTIFICATION DOCUMENT NO.:%d",s.ID);
```

```
printf("\n\t\tPHONE NUMBER:%s",s.ph);
```

```
printf("\n\t\tmark :%f",s.Mark);
```

```
printf("\n\t\tGURDAIN NAME:%s",s.gname);
```

```
printf("\n\t\tEMAIL ADDRESS OF STUDENT:%s",s.email);
```

```
printf("\n\t\tGURDAIN PHONE NUMBER:%s",s.gph);
```

```
printf("\n\n\t");
```



```

        printChar('x',40);

        printf("\n\n");

    }

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

    printChar('*',65);

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

    system("pause");

}

//=====SEARCH=====

void searchRecord(FILE *fp)

{

    printHead();

    int tempRoll,flag,siz,i;

    Student s;

    char another='y';

    siz=sizeof(s);

    while(another=='y'||another=='Y')

    {

```

```

printf("\n\n\tEnter Identification Number of Student to search the record : ");

scanf("%d",&tempRoll);

rewind(fp);

while((fread(&s,siz,1,fp))==1)

{

    if(s.ID==tempRoll)

        { flag=1;

          break;

        }

}

if(flag==1)

{

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

    printf("\n\t\tDATE OF BIRTH:%s",s.dob);

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

    printf("\n\t\tIDENTIFICATION DOCUMENT NO.: %d",s.ID);

    printf("\n\t\tPHONE NUMBER:%s",s.ph);

    printf("\n\t\tGURDAIN NAME:%s",s.gname);

```

```

printf("\n\t\tEMAIL OF STUDENT:%s",s.email);

printf("\n\t\tGURDAIN PHONE NUMBER:%s",s.gph);

}

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

printf("\n");

printChar('~',65);

printf("\n\n\t\tWant to enter another search (Y/N)");

fflush(stdin);

another=getch();

}

}

```