KENDRIYA VIDYALAYA, BAILEY ROAD, PATNA



COMPUTER SCIENCE PROJECT

TEST TAKING APPLICATION

SAIFUR RAHMAN, KUMAR SARTHAK, KISHAN AKHAURY  
XII B

BOARD Roll: 7634808, -793, -791

SESSION: 2015-16

CERTIFICATE

It is to certify that Saifur Rahman, Kumar Sarthak and Kishan Akhaury, students of class XII B has successfully completed the project entitled “Test Taking Application” under the guidance of Miss Sarwari Begum during the year 2015-16 in partial fulfilment of Computer Science Practical examination conducted by AISSCE, New Delhi.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Signature of External Signature of Computer

Examiner Teacher

Acknowledgement

I sincerely like to thank my Computer Teacher for her guidance, support and advice to complete my work successfully. I also thank Mr. S. Vallabhan for providing all the facilities in the Computer Lab. that helped me to complete this project.

I would also like to thank my parents for their continuous corporation in completing the project.

And, lastly I would thank my classmates for their corporation and support at different stage of project.

CONTENTS

Certificate

Acknowledgement

Contents

Requirements

Header Files

Interface Design

Code

Bibliography

INTRODUCTION

This Test Taking Application is automated system for taking MCQ type tests and generation of report card of applicants after examination.

This application is an automated system that keeps the track of records of marks of applicants easily and effectively. It is developed for solution of keeping record of questions, marks and automated report of student marks with rank and application settings.

This application takes record of MCQ type questions and applicants registered for the test.

The application can also be altered in different ways to change settings to suit different types of tests.

REQUIREMENTS

**HARDWARE REQUIREMENTS**

Tools/Platform

Processor : Intel Core I series

HDD : 100 MB

RAM : 512 MB

Language : Turbo C++ 3.0

Platform : MS-DOS

**WHY C++**

C++ is an object oriented language with all feature of object oriented. It also support file handling for data storage. Therefore it is very easy to use file with object oriented paradigm.

**SOFTWARE REQUIREMENT SPECIFICATIONS**

Introduction

Main objective of the system is to improve efficiency and quality of service by designing and developing user friendly, efficient system.

Information Description

The system will help the teacher for taking test, keeping marks, provide rank, registers the applicants and to edit details of applicants in the way that there will be minimal human interference and high user friendliness and less effort taking. Reducing duplication during manual processing and managing records. The user will be given sufficient menu options for data entry, report or other miscellaneous task.

The whole processing in old system is manual, which results in complexity

and at times result in unnecessary duplication and wastage of resources. Information retrieval is a time consuming process.

New system will provide more flexibility, easier to take test of the students and store their performance. Records can be stored, retrieved, deleted or updated with less effort. User interface is simple and operator requires less hardware/software knowledge.

HEADER FILES

iostream.h :- For basic input and output .

conio.h:- For using clrscr().

stdlib.h:- For randomise() function.

string.h:- To perform string related functifon.

fstream.h:-To perform file stream function.

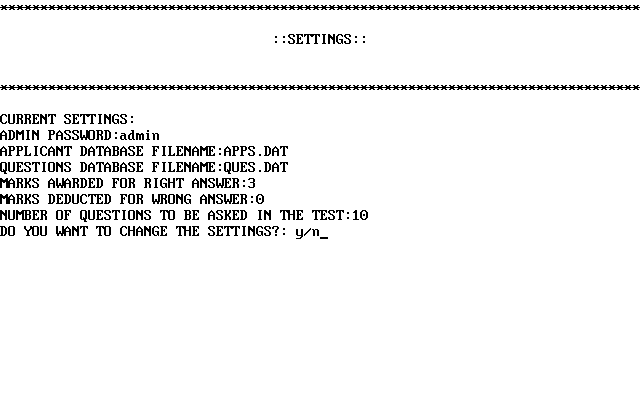
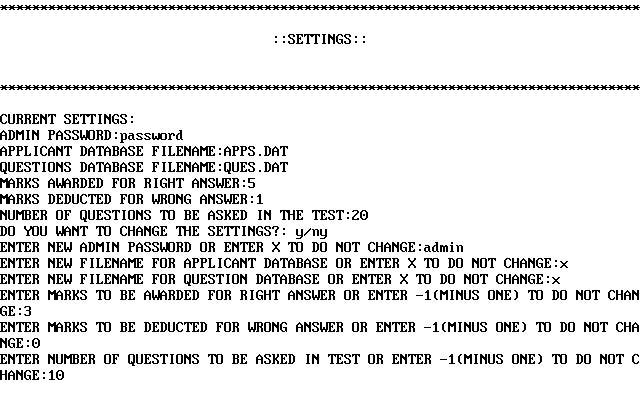
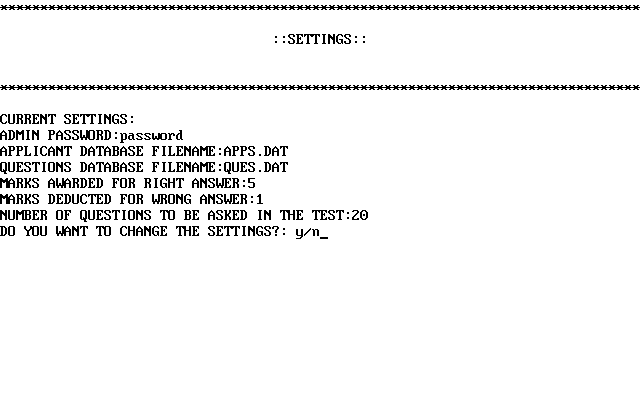
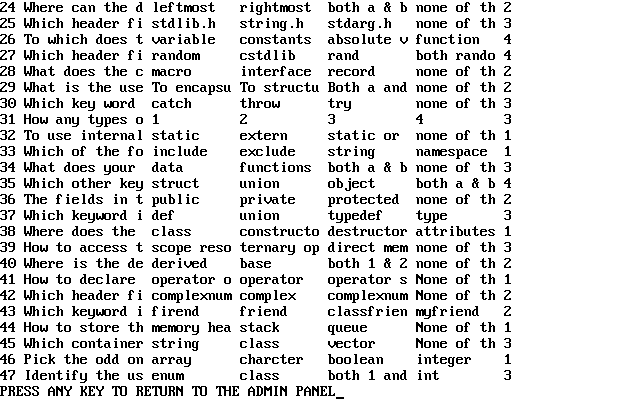
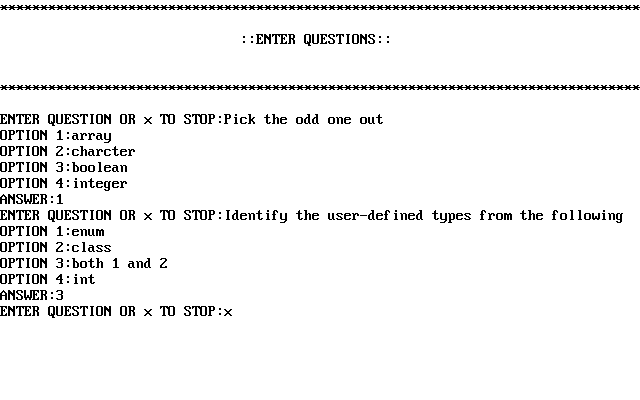
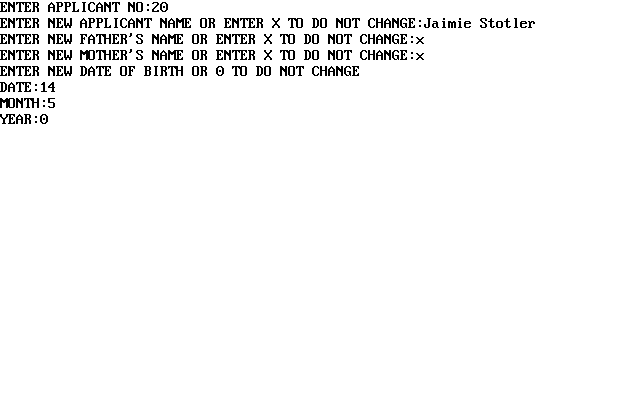
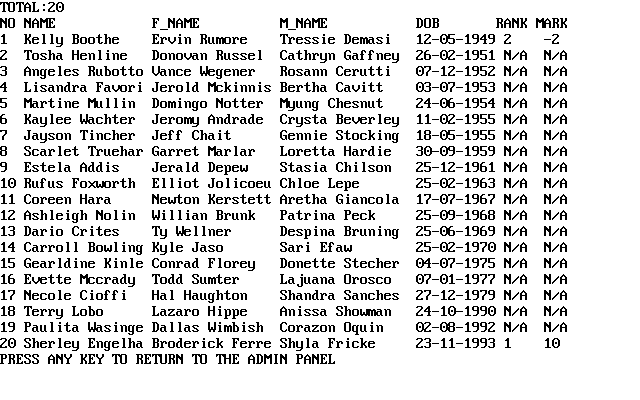
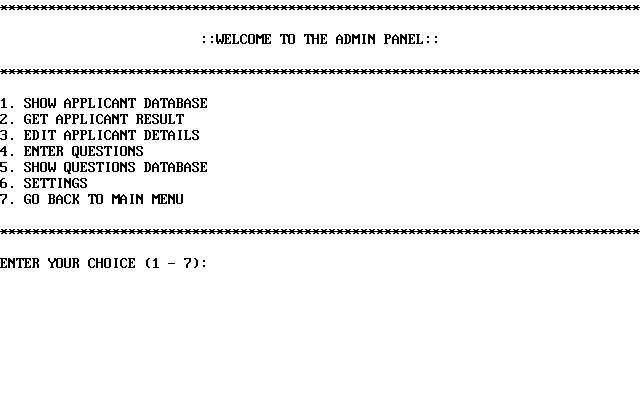
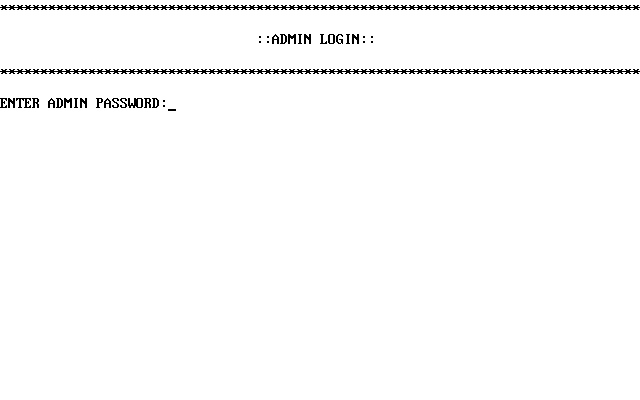
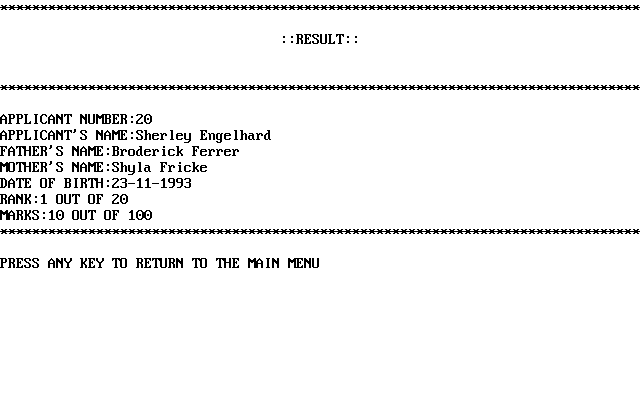
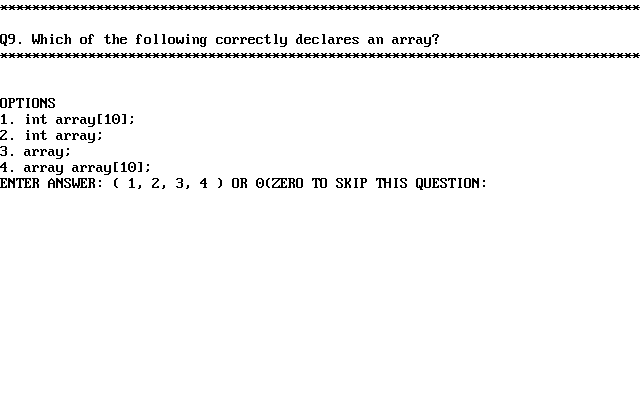
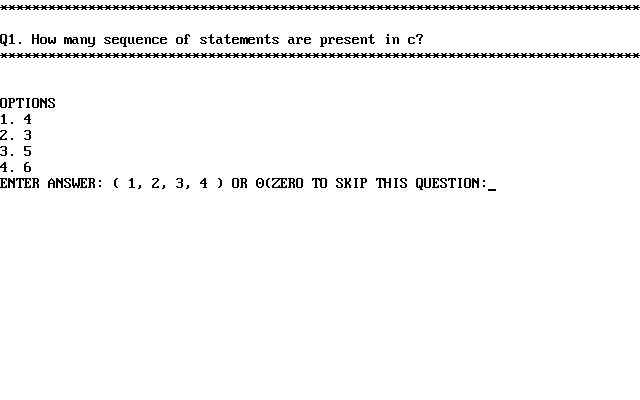
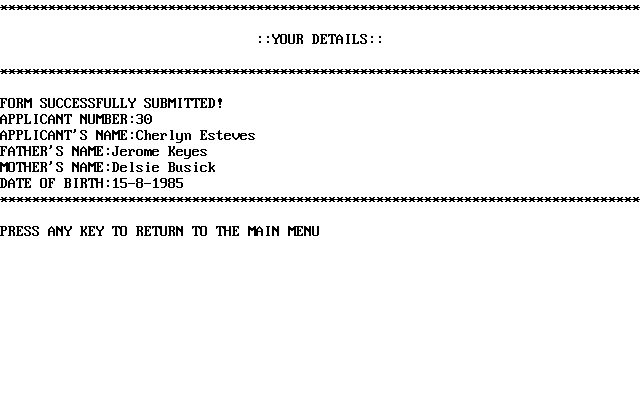
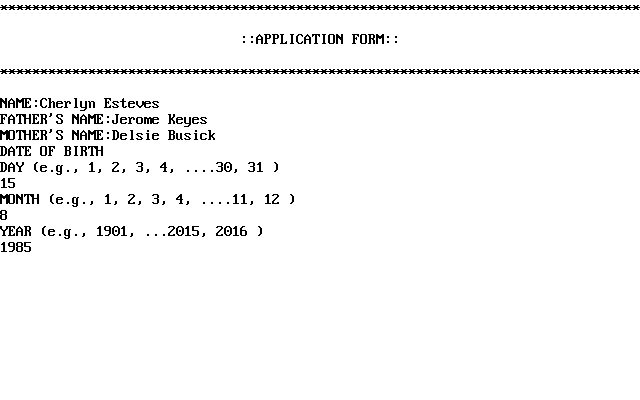
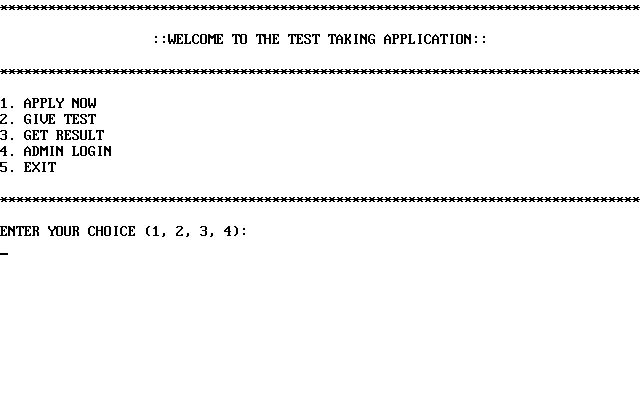
stdio.h:- For standard I/O operation.

time.h:-To use time in generating random numbers.

iomanip.h:-To format the output.

INTERFACE

DESIGN



CODE

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// HEADER FILE USED IN PROJECT

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

#include<iostream.h>

#include<conio.h>

#include<stdlib.h>

#include<string.h>

#include<fstream.h>

#include<stdio.h>

#include<time.h>

#include<iomanip.h>

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// FUNCTION PROTOTYPES

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void main();

void show\_applicant\_database();

void edit\_applicant();

void enter\_questions();

void show\_questions\_database();

void assign\_rank();

void settings();

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// GLOBAL VARIABLES

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

int main\_menu\_choice;

int dd,dm,dy;

char na[50], fa[50], ma[50];

int no\_of\_applicants=0;

int apno, get\_d, get\_m, get\_y;

char pass[20];

int admin\_menu\_choice;

int ed\_apno, ed\_new\_dob\_d, ed\_new\_dob\_m, ed\_new\_dob\_y;

char ed\_new\_name[50], ed\_new\_faname[50], ed\_new\_maname[50];

int pos;

int no\_of\_questions=0;

char add\_q[100], add\_o1[100], add\_o2[100], add\_o3[100], add\_o4[100];

int add\_a;

int t\_apno, t\_dob\_d, t\_dob\_m, t\_dob\_y;

int pos2;

int marks[500];

int pos3;

int negative\_mark=1;

int positive\_mark=5;

int no\_of\_ques\_to\_ask=20;

char password[30]="password";

char app\_file\_name[30]="APPS.DAT";

char ques\_file\_name[30]="QUES.DAT";

int new\_negative\_mark;

int new\_positive\_mark;

int new\_no\_of\_ques\_to\_ask;

char new\_password[30];

char new\_app\_file\_name[30];

char new\_ques\_file\_name[30];

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// CLASSES USED IN THE PROJECT

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// APPLICANTS CLASS (TO SAVE APPLICANTS)

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

class Applicants {

int applicant\_no;

char name[50];

char father\_name[50];

char mother\_name[50];

int dob\_day;

int dob\_mon;

int dob\_year;

int marks;

int rank;

public:

int get\_applicant\_no(){ return applicant\_no; }

int get\_dob\_d() { return dob\_day; }

int get\_dob\_m() { return dob\_mon; }

int get\_dob\_y() { return dob\_year; }

void get\_name() { cout<<name; }

void get\_faname() { cout<<father\_name; }

void get\_maname() { cout<<mother\_name; }

int get\_marks() {return marks;}

int get\_rank() {return rank;}

void set\_marks(int mk) { marks=mk; }

void set\_rank(int rn) { rank=rn; }

void set\_applicant\_no(int ap) { applicant\_no=ap; }

void set\_dob\_d(int dd) { dob\_day=dd; }

void set\_dob\_m(int dm) { dob\_mon=dm; }

void set\_dob\_y(int dy) { dob\_year=dy; }

void set\_name(char na[]){ strcpy(name,na); }

void set\_maname(char ma[]){ strcpy(mother\_name,ma); }

void set\_faname(char fa[]){ strcpy(father\_name,fa); }

void get\_short\_name(){

if(strlen(name)<15)

{ cout.width(15); cout.fill(' '); cout<<name; }

else

cout.write(name,15);

}

void get\_short\_faname(){

if(strlen(father\_name)<15)

{ cout.width(15); cout.fill(' '); cout<<father\_name; }

else

cout.write(father\_name,15);

}

void get\_short\_maname(){

if(strlen(mother\_name)<15)

{ cout.width(15); cout.fill(' '); cout<<mother\_name; }

else

cout.write(mother\_name,15);

}

};

void calc\_no\_of\_applicants(){

ifstream fin4calc;

fin4calc.open(app\_file\_name,ios::in|ios::binary);

fin4calc.seekg(0,ios::end);

no\_of\_applicants= fin4calc.tellg()/sizeof(Applicants);

fin4calc.close();

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// QUESTIONS CLASS (TO SAVE QUESTIONS)

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

class Questions{

int id;

char ques[100];

char opt1[100];

char opt2[100];

char opt3[100];

char opt4[100];

int ans;

public:

void set\_id(int i) { id=i; }

void set\_ques(char q[]) { strcpy(ques,q); }

void set\_opt1(char o1[]) { strcpy(opt1,o1); }

void set\_opt2(char o2[]) { strcpy(opt2,o2); }

void set\_opt3(char o3[]) { strcpy(opt3,o3); }

void set\_opt4(char o4[]) { strcpy(opt4,o4); }

void set\_ans(int a) { ans=a; }

int get\_id() { return id; }

void get\_ques() { cout<<ques;}

void get\_opt1() { cout<<opt1; }

void get\_opt2() { cout<<opt2; }

void get\_opt3() { cout<<opt3; }

void get\_opt4() { cout<<opt4; }

int get\_ans() { return ans; }

void get\_short\_ques(){

if(strlen(ques)<15)

{ cout.width(15); cout.fill(' '); cout<<ques; }

else

cout.write(ques,15);

}

void get\_short\_opt1(){

if(strlen(opt1)<10)

{ cout.width(10); cout.fill(' '); cout<<opt1; }

else

cout.write(opt1,10);

}

void get\_short\_opt2(){ if(strlen(opt2)<10)

{ cout.width(10); cout.fill(' '); cout<<opt2; }

else

cout.write(opt2,10);

}

void get\_short\_opt3(){

if(strlen(opt3)<10)

{ cout.width(10); cout.fill(' '); cout<<opt3; }

else

cout.write(opt3,10);

}

void get\_short\_opt4(){ if(strlen(opt4)<10)

{ cout.width(10); cout.fill(' '); cout<<opt4; }

else

cout.write(opt4,10);

}

};

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// STATE CLASS (TO SAVE SETTINGS)

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

class State{

int max\_marks;

int negative\_mark;

int positive\_mark;

int no\_of\_ques\_to\_ask;

char password[30];

char app\_file\_name[30];

char ques\_file\_name[30];

public:

int get\_negative\_mark(){ return negative\_mark; }

int get\_positive\_mark(){ return positive\_mark; }

int get\_no\_of\_ques\_to\_ask(){ return no\_of\_ques\_to\_ask; }

char\* get\_password(){ return password; }

char\* get\_app\_file\_name(){ return app\_file\_name; }

char\* get\_ques\_file\_name(){ return ques\_file\_name; }

void set\_negative\_mark(int nm){ negative\_mark=nm;}

void set\_positive\_mark(int pm){ positive\_mark=pm;}

void set\_no\_of\_ques\_to\_ask(int nq){ no\_of\_ques\_to\_ask=nq;}

void set\_password(char pw[]){ strcpy(password, pw); }

void set\_app\_file\_name(char af[]){ strcpy(app\_file\_name, af); }

void set\_ques\_file\_name(char qf[]){ strcpy(ques\_file\_name, qf); }

};

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// FUNCTION TO CALCULATE NUMBER OF QUESTIONS IN THE FILE

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void calc\_no\_of\_questions(){

ifstream fin4calc\_q;

fin4calc\_q.open(ques\_file\_name,ios::in|ios::binary|ios::ate);

no\_of\_questions= fin4calc\_q.tellg()/sizeof(Questions);

fin4calc\_q.close();

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// FUNCTION REGISTER APPLICANT TO THE FILE

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void apply\_now() {

Applicants t\_applicant;

ofstream fout;

t\_applicant.set\_marks(-500);

t\_applicant.set\_rank(-1);

clrscr();

fout.open(app\_file\_name, ios::app|ios::binary);

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<" ::APPLICATION FORM:: "<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<"NAME:";

gets(na);

t\_applicant.set\_name(na);

cout<<"FATHER'S NAME:";

gets(fa);

t\_applicant.set\_faname(fa);

cout<<"MOTHER'S NAME:";

gets(ma);

t\_applicant.set\_maname(ma);

cout<<"DATE OF BIRTH"<<endl;

do{

cout<<"DAY (e.g., 1, 2, 3, 4, ....30, 31 )"<<endl;

cin>>dd;

}

while(!((dd>0)&&(dd<32)));

t\_applicant.set\_dob\_d(dd);

do{

cout<<"MONTH (e.g., 1, 2, 3, 4, ....11, 12 )"<<endl;

cin>>dm;

}

while(!((dm>0)&&(dm<13)));

t\_applicant.set\_dob\_m(dm);

do{

cout<<"YEAR (e.g., 1901, ...2015, 2016 )"<<endl;

cin>>dy;

}

while(!((dy>1900)&&(dy<3000)));

t\_applicant.set\_dob\_y(dy);

calc\_no\_of\_applicants();

t\_applicant.set\_applicant\_no(no\_of\_applicants+1);

fout.write((char\*)&t\_applicant, sizeof(t\_applicant));

fout.close();

calc\_no\_of\_applicants();

clrscr();

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<" ::YOUR DETAILS:: "<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<"FORM SUCCESSFULLY SUBMITTED!"<<endl;

cout<<"APPLICANT NUMBER:"<<t\_applicant.get\_applicant\_no()<<endl;

cout<<"APPLICANT'S NAME:";

t\_applicant.get\_name();

cout<<endl;

cout<<"FATHER'S NAME:";

t\_applicant.get\_faname();

cout<<endl;

cout<<"MOTHER'S NAME:";

t\_applicant.get\_maname();

cout<<endl;

cout<<"DATE OF BIRTH:"<<t\_applicant.get\_dob\_d()<<"-"<<t\_applicant.get\_dob\_m()<<"-"<<t\_applicant.get\_dob\_y()<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<"PRESS ANY KEY TO RETURN TO THE MAIN MENU";

getch();

main();

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// FUNCTION TO TAKE THE TEST OF REGISTERED APPLICANTS

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void give\_test(){

Applicants test\_applicant;

fstream fin;

fin.open(app\_file\_name,ios::in|ios::out|ios::binary);

clrscr();

calc\_no\_of\_questions();

if(no\_of\_questions<no\_of\_ques\_to\_ask)

{

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<"INSUFFICIENT QUESTION IN DATABASE. CONTACT ADMINISTRATOR"<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<"PRESS ANY KEY TO RETURN TO THE MAIN MENU";

getch();

main();

}

cout<<"ENTER YOUR APPLICANT NUMBER:"<<endl;

cin>>t\_apno;

cout<<"ENTER DATE OF BIRTH"<<endl;

cout<<"ENTER DAY:";

cin>>t\_dob\_d;

cout<<"ENTER MONTH:";

cin>>t\_dob\_m;

cout<<"ENTER YEAR:";

cin>>t\_dob\_y;

int flag=0;

while (!fin.eof())

{

pos2=fin.tellg();

fin.read((char\*)&test\_applicant,sizeof(test\_applicant));

if(test\_applicant.get\_applicant\_no()==t\_apno&&test\_applicant.get\_dob\_d()==t\_dob\_d&&test\_applicant.get\_dob\_m()==t\_dob\_m&&test\_applicant.get\_dob\_y()==t\_dob\_y)

{

flag=1;

cout<<endl;

break;

}

}

if(flag==1)

{

Questions t\_obj;

ifstream t\_ques;

t\_ques.open(ques\_file\_name,ios::in|ios::binary);

time\_t t;

int seedval= (unsigned) time (&t);

srand(seedval);

int marks=0;

int qno;

int ans;

for(int i=0; i<no\_of\_ques\_to\_ask;i++)

{

qno=rand()%no\_of\_questions+1;

while(t\_ques.read((char\*)&t\_obj, sizeof(t\_obj)))

{

if(t\_obj.get\_id()==qno)

{

clrscr();

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<"Q"<<i+1<<". ";

t\_obj.get\_ques();

cout<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<endl;

cout<<"OPTIONS"<<endl<<"1. ";

t\_obj.get\_opt1();

cout<<endl;

cout<<"2. ";

t\_obj.get\_opt2();

cout<<endl;

cout<<"3. ";

t\_obj.get\_opt3();

cout<<endl;

cout<<"4. ";

t\_obj.get\_opt4();

cout<<endl;

cout<<"ENTER ANSWER: ( 1, 2, 3, 4 ) OR 0(ZERO TO SKIP THIS QUESTION:";

cin>>ans;

if (ans==0)

{

;

}

else if(t\_obj.get\_ans()==ans)

{ marks = marks + positive\_mark;}

else{ marks = marks - negative\_mark;}

break;

}

}

t\_ques.seekg(0);

}

test\_applicant.set\_marks(marks);

fin.seekg(pos2);

fin.write((char\*)&test\_applicant,sizeof(test\_applicant));

t\_ques.close();

fin.close();

}

if(flag==0)

{

clrscr();

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<"APPLICANT NOT FOUND!"<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

}

assign\_rank();

cout<<"PRESS ANY KEY TO RETURN TO THE MAIN MENU";

getch(); main();

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// FUNCTION TO SHOW RESULT OF THE REGISTERED APPLICANTS

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void get\_result(){

Applicants t2\_applicant;

ifstream fin;

fin.open(app\_file\_name,ios::in|ios::binary);

clrscr();

cout<<"ENTER YOUR APPLICANT NUMBER:"<<endl;

cin>>apno;

cout<<"ENTER DATE OF BIRTH"<<endl;

cout<<"ENTER DAY:";

cin>>get\_d;

cout<<"ENTER MONTH:";

cin>>get\_m;

cout<<"ENTER YEAR:";

cin>>get\_y;

int flag=0;

while(fin.read((char \*)&t2\_applicant,sizeof(t2\_applicant))){

if(t2\_applicant.get\_applicant\_no()==apno&&t2\_applicant.get\_dob\_d()==get\_d&&t2\_applicant.get\_dob\_m()==get\_m&&t2\_applicant.get\_dob\_y()==get\_y)

{

clrscr();

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<" ::RESULT:: "<<endl<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<"APPLICANT NUMBER:"<<t2\_applicant.get\_applicant\_no()<<endl;

cout<<"APPLICANT'S NAME:";

t2\_applicant.get\_name();

cout<<endl;

cout<<"FATHER'S NAME:";

t2\_applicant.get\_faname();

cout<<endl;

cout<<"MOTHER'S NAME:";

t2\_applicant.get\_maname();

cout<<endl;

cout<<"DATE OF BIRTH:"<<t2\_applicant.get\_dob\_d()<<"-"<<t2\_applicant.get\_dob\_m()<<"-"<<t2\_applicant.get\_dob\_y()<<endl;

if(t2\_applicant.get\_rank()!=-1)

{ calc\_no\_of\_applicants();

cout<<"RANK:"<<t2\_applicant.get\_rank()<<" OUT OF "<<no\_of\_applicants<<endl; }

else { cout<<"RANK NOT ASSIGNED"<<endl; }

if(t2\_applicant.get\_marks()!=-500)

{ cout<<"MARKS:"<<t2\_applicant.get\_marks()<<" OUT OF "<<no\_of\_ques\_to\_ask\*positive\_mark<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

}

else { cout<<"MARKS NOT ASSIGNED"<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

}

flag=1;

break;

}

}

if(!flag){

clrscr();

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<"APPLICANT NOT FOUND!"<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

}

fin.close();

cout<<"PRESS ANY KEY TO RETURN TO THE MAIN MENU";

getch(); main();

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// FUNCTION TO GIVE ADMINISTRATOR CONTROLS TO THE USER

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void admin\_login(){

clrscr();

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<" ::ADMIN LOGIN:: "<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<"ENTER ADMIN PASSWORD:";

gets(pass);

if(strcmp(pass,password)==0)

{

clrscr();

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<" ::WELCOME TO THE ADMIN PANEL:: "<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<"1. SHOW APPLICANT DATABASE"<<endl;

cout<<"2. GET APPLICANT RESULT"<<endl;

cout<<"3. EDIT APPLICANT DETAILS"<<endl;

cout<<"4. ENTER QUESTIONS"<<endl;

cout<<"5. SHOW QUESTIONS DATABASE"<<endl;

cout<<"6. SETTINGS"<<endl;

cout<<"7. GO BACK TO MAIN MENU"<<endl<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<"ENTER YOUR CHOICE (1 - 7):";

cin>>admin\_menu\_choice;

switch(admin\_menu\_choice){

case 1 : show\_applicant\_database();

break;

case 2 : get\_result();

break;

case 3: edit\_applicant();

break;

case 4: enter\_questions();

break;

case 5 : show\_questions\_database();

break;

case 6 : settings();

break;

case 7 : main();

break;

default: cout<<"WRONG CHOICE!";

cout<<"PRESS ANY KEY TO RETURN TO THE MAIN MENU";

getch();

main();

}

}

else

{

clrscr();

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<"ACCESS DENIED!"<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<"PRESS ANY KEY TO RETURN TO THE MAIN MENU";

getch();

main();

}

getch();

main();

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// FUNCTION TO SHOW ALL ENTRIES IN THE APPLICANT FILE

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void show\_applicant\_database(){

Applicants t3\_applicant;

ifstream fin;

fin.open(app\_file\_name, ios::binary|ios::in);

calc\_no\_of\_applicants();

clrscr();

cout<<"TOTAL:"<<no\_of\_applicants<<endl;

cout.setf(ios::left);

cout<<setw(3)<<"NO"<<setw(16)<<"NAME"<<setw(16)<<"F\_NAME"<<setw(16)<<"M\_NAME"<<setw(11)<<" DOB"<<setw(5)<<"RANK"<<setw(5)<<"MARK"<<endl;

while(fin.read((char\*)&t3\_applicant,sizeof(t3\_applicant)))

{

cout<<setw(3)<<t3\_applicant.get\_applicant\_no();

cout<<setw(16);

t3\_applicant.get\_short\_name();

cout<<' ';

cout<<setw(16);

t3\_applicant.get\_short\_faname();

cout<<setw(16);

cout<<' ';

t3\_applicant.get\_short\_maname();

cout<<' ';

cout.setf(ios::right);

cout<<' '<<setw(2)<<setfill('0')<<t3\_applicant.get\_dob\_d()<<"-"<<setw(2)<<setfill('0')<<t3\_applicant.get\_dob\_m()<<"-"<<setw(4)<<setfill('0')<<t3\_applicant.get\_dob\_y()<<' '<<setfill(' ');

cout.setf(ios::left);

if(t3\_applicant.get\_rank()!=-1)

{ cout<<setw(5)<<t3\_applicant.get\_rank(); }

else { cout<<setw(5)<<"N/A"; }

if(t3\_applicant.get\_marks()!=-500)

{ cout<<setw(4)<<t3\_applicant.get\_marks()<<endl; }

else { cout<<setw(4)<<"N/A"<<endl; }

}

fin.close();

cout<<"PRESS ANY KEY TO RETURN TO THE ADMIN PANEL";

getch();

admin\_login();

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// FUNCTION TO EDIT DETAILS OF REGISTERED APPLICANTS

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void edit\_applicant(){

Applicants ed\_obj;

fstream ed\_file;

ed\_file.open(app\_file\_name,ios::in|ios::out|ios::binary);

clrscr();

cout<<"ENTER APPLICANT NO:";

cin>>ed\_apno;

int flag\_ed=0;

while(!ed\_file.eof())

{

pos=ed\_file.tellg();

ed\_file.read((char\*)&ed\_obj,sizeof(ed\_obj));

if(ed\_obj.get\_applicant\_no()==ed\_apno)

{

cout<<"ENTER NEW APPLICANT NAME OR ENTER X TO DO NOT CHANGE:";

gets(ed\_new\_name);

if((strcmp(ed\_new\_name,"x")!=0)&&(strcmp(ed\_new\_name,"X")!=0))

{

ed\_obj.set\_name(ed\_new\_name);

}

cout<<"ENTER NEW FATHER'S NAME OR ENTER X TO DO NOT CHANGE:";

gets(ed\_new\_faname);

if((strcmp(ed\_new\_faname,"x")!=0)&&(strcmp(ed\_new\_faname,"X")!=0))

{

ed\_obj.set\_faname(ed\_new\_faname);

}

cout<<"ENTER NEW MOTHER'S NAME OR ENTER X TO DO NOT CHANGE:";

gets(ed\_new\_maname);

if((strcmp(ed\_new\_maname,"x")!=0)&&(strcmp(ed\_new\_maname,"X")!=0))

{

ed\_obj.set\_maname(ed\_new\_maname);

}

cout<<"ENTER NEW DATE OF BIRTH OR 0 TO DO NOT CHANGE"<<endl;

cout<<"DATE:";

cin>>ed\_new\_dob\_d;

if(ed\_new\_dob\_d!=0)

{

ed\_obj.set\_dob\_d(ed\_new\_dob\_d);

}

cout<<"MONTH:";

cin>>ed\_new\_dob\_m;

if(ed\_new\_dob\_m!=0)

{

ed\_obj.set\_dob\_m(ed\_new\_dob\_m);

}

cout<<"YEAR:";

cin>>ed\_new\_dob\_y;

if(ed\_new\_dob\_y!=0)

{

ed\_obj.set\_dob\_y(ed\_new\_dob\_y);

}

ed\_file.seekg(pos);

ed\_file.write((char\*)&ed\_obj, sizeof(ed\_obj));

flag\_ed=1;

clrscr();

cout<<"APPLICANT DETAILS SUCCESSFULLY CHANGED!"<<endl;

break;

}

}

if(!flag\_ed)

{

clrscr();

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<"APPLICANT NOT FOUND!"<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

}

ed\_file.close();

clrscr();

cout<<"PRESS ANY KEY TO RETURN TO THE ADMIN PANEL";

getch();

admin\_login();

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// FUNCTION TO ENTER QUESTIONS INTO THE FILE

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void enter\_questions(){

Questions t\_question;

ofstream fout;

char ask='y';

calc\_no\_of\_questions();

clrscr();

fout.open(ques\_file\_name, ios::app|ios::binary);

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<" ::ENTER QUESTIONS:: "<<endl<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

while(ask!='n'){

cout<<"ENTER QUESTION OR x TO STOP:";

gets(add\_q);

if(strcmp(add\_q,"X")==0)

{

ask='n';

continue;

}

if(strcmp(add\_q,"x")==0)

{

ask='n';

continue;

}

t\_question.set\_ques(add\_q);

cout<<"OPTION 1:";

gets(add\_o1);

t\_question.set\_opt1(add\_o1);

cout<<"OPTION 2:";

gets(add\_o2);

t\_question.set\_opt2(add\_o2);

cout<<"OPTION 3:";

gets(add\_o3);

t\_question.set\_opt3(add\_o3);

cout<<"OPTION 4:";

gets(add\_o4);

t\_question.set\_opt4(add\_o4);

cout<<"ANSWER:";

cin>>add\_a;

t\_question.set\_ans(add\_a);

t\_question.set\_id(++no\_of\_questions);

fout.write((char\*)&t\_question, sizeof(t\_question));

}

fout.close();

calc\_no\_of\_questions();

clrscr();

cout<<"PRESS ANY KEY TO RETURN TO THE ADMIN PANEL";

getch();

admin\_login();

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// FUNCTION TO SHOW ALL ENTRIES IN THE QUESTION FILE

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void show\_questions\_database(){

Questions t3\_question;

ifstream fin;

fin.open(ques\_file\_name, ios::binary|ios::in);

calc\_no\_of\_questions();

clrscr();

cout<<"TOTAL:"<<no\_of\_questions<<endl;

cout.setf(ios::left);

cout<<setfill(' ')<<setw(3)<<"ID"<<setw(16)<<"QUES"<<setw(11)<<"OPT1"<<setw(11)<<"OPT2"<<setw(11)<<"OPT3"<<setw(11)<<"OPT4"<<setw(2)<<"ANS"<<endl;

while(fin.read((char\*)&t3\_question,sizeof(t3\_question)))

{

cout<<setw(3)<<t3\_question.get\_id();

cout<<setw(16);

t3\_question.get\_short\_ques();

cout<<' ';

cout<<setw(11);

t3\_question.get\_short\_opt1();

cout<<' ';

cout<<setw(11);

t3\_question.get\_short\_opt2();

cout<<' ';

cout<<setw(11);

t3\_question.get\_short\_opt3();

cout<<' ';

cout<<setw(11);

t3\_question.get\_short\_opt4();

cout<<' ';

cout<<setw(2);

cout<<t3\_question.get\_ans()<<endl;

}

fin.close();

cout<<"PRESS ANY KEY TO RETURN TO THE ADMIN PANEL";

getch();

admin\_login();

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// FUNCTION TO ASSIGN RANK TO APPLICANTS

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void assign\_rank(){

int big, pos, temp;

Applicants t\_app\_obj;

fstream t\_app\_file;

t\_app\_file.open(app\_file\_name, ios::in|ios::out|ios::binary);

int i=0;

int marks\_array[100];

while(t\_app\_file.read((char\*)&t\_app\_obj, sizeof(t\_app\_obj)))

{

marks\_array[i]=t\_app\_obj.get\_marks();

i++;

}

calc\_no\_of\_applicants();

for(i=0;i<no\_of\_applicants;i++)

{

big = marks\_array[i];

pos=i;

for(int j=i+1;j<no\_of\_applicants;j++)

{

if(marks\_array[j]>big)

{ big = marks\_array[j];

pos=j;

}

}

if(i!=pos)

{ temp = marks\_array[i];

marks\_array[i] = marks\_array[pos];

marks\_array[pos] = temp;

}

}

t\_app\_file.clear();

t\_app\_file.seekg(0);

while (!t\_app\_file.eof()) {

pos3=t\_app\_file.tellg();

t\_app\_file.read((char\*)&t\_app\_obj, sizeof(t\_app\_obj));

for(i = no\_of\_applicants; i>-1;i--)

{

if(t\_app\_obj.get\_marks()==-500)

{

t\_app\_obj.set\_rank(-1);

break;

}

else if(t\_app\_obj.get\_marks()==marks\_array[i])

{

t\_app\_obj.set\_rank(i+1);

break;

}

}

t\_app\_file.seekg(pos3);

t\_app\_file.write((char\*)&t\_app\_obj, sizeof(t\_app\_obj));

}

t\_app\_file.close();

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// FUNCTION TO INITIALISE SETTINGS FROM CONFIGURATION FILE

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void configure(){

State config\_obj;

fstream config\_file;

config\_file.open("config.dat",ios::in|ios::binary);

cout<<config\_file;

if(config\_file){

config\_file.read((char\*)&config\_obj,sizeof(config\_obj));

strcpy(password, config\_obj.get\_password());

positive\_mark=config\_obj.get\_positive\_mark();

negative\_mark=config\_obj.get\_negative\_mark();

no\_of\_ques\_to\_ask=config\_obj.get\_no\_of\_ques\_to\_ask();

strcpy(app\_file\_name,config\_obj.get\_app\_file\_name());

strcpy(ques\_file\_name,config\_obj.get\_ques\_file\_name());

}

config\_file.close();

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// FUNCTION EDIT SETTINGS

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void settings(){

clrscr();

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<" ::SETTINGS:: "<<endl<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<"CURRENT SETTINGS:"<<endl;

cout<<"ADMIN PASSWORD:"<<password<<endl;

cout<<"APPLICANT DATABASE FILENAME:"<<app\_file\_name<<endl;

cout<<"QUESTIONS DATABASE FILENAME:"<<ques\_file\_name<<endl;

cout<<"MARKS AWARDED FOR RIGHT ANSWER:"<<positive\_mark<<endl;

cout<<"MARKS DEDUCTED FOR WRONG ANSWER:"<<negative\_mark<<endl;

cout<<"NUMBER OF QUESTIONS TO BE ASKED IN THE TEST:"<<no\_of\_ques\_to\_ask<<endl;

cout<<"DO YOU WANT TO CHANGE THE SETTINGS?: y/n";

char ch;

cin>>ch;

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

{

State config\_obj;

ofstream config\_file;

config\_file.open("config.dat",ios::out|ios::trunc|ios::binary);

cout<<"ENTER NEW ADMIN PASSWORD OR ENTER X TO DO NOT CHANGE:";

gets(new\_password);

if((strcmp(new\_password,"x")!=0)&&(strcmp(new\_password,"X")!=0))

{

config\_obj.set\_password(new\_password);

}

else

{

config\_obj.set\_password(password);

}

cout<<"ENTER NEW FILENAME FOR APPLICANT DATABASE OR ENTER X TO DO NOT CHANGE:";

gets(new\_app\_file\_name);

if((strcmp(new\_app\_file\_name,"x")!=0)&&(strcmp(new\_app\_file\_name,"X")!=0))

{

config\_obj.set\_app\_file\_name(new\_app\_file\_name);

}

else

{

config\_obj.set\_app\_file\_name(app\_file\_name);

}

cout<<"ENTER NEW FILENAME FOR QUESTION DATABASE OR ENTER X TO DO NOT CHANGE:";

gets(new\_ques\_file\_name);

if((strcmp(new\_ques\_file\_name,"x")!=0)&&(strcmp(new\_ques\_file\_name,"X")!=0))

{

config\_obj.set\_ques\_file\_name(new\_ques\_file\_name);

}

else

{

config\_obj.set\_ques\_file\_name(ques\_file\_name);

}

cout<<"ENTER MARKS TO BE AWARDED FOR RIGHT ANSWER OR ENTER -1(MINUS ONE) TO DO NOT CHANGE:";

cin>>new\_positive\_mark;

if(new\_positive\_mark!=-1)

{

config\_obj.set\_positive\_mark(new\_positive\_mark);

}

else

{

config\_obj.set\_positive\_mark(positive\_mark);

}

cout<<"ENTER MARKS TO BE DEDUCTED FOR WRONG ANSWER OR ENTER -1(MINUS ONE) TO DO NOT CHANGE:";

cin>>new\_negative\_mark;

if(new\_negative\_mark!=-1)

{

config\_obj.set\_negative\_mark(new\_negative\_mark);

}

else

{

config\_obj.set\_negative\_mark(negative\_mark);

}

cout<<"ENTER NUMBER OF QUESTIONS TO BE ASKED IN TEST OR ENTER -1(MINUS ONE) TO DO NOT CHANGE:";

cin>>new\_no\_of\_ques\_to\_ask;

if(new\_no\_of\_ques\_to\_ask!=-1)

{

config\_obj.set\_no\_of\_ques\_to\_ask(new\_no\_of\_ques\_to\_ask);

}

else

{

config\_obj.set\_no\_of\_ques\_to\_ask(no\_of\_ques\_to\_ask);

}

config\_file.write((char\*)&config\_obj,sizeof(config\_obj));

config\_file.close();

configure();

}

admin\_login();

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// THE MAIN FUNCTION OF THE PROGRAM

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void main(){

clrscr();

configure();

clrscr();

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<" ::WELCOME TO THE TEST TAKING APPLICATION:: "<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<"1. APPLY NOW"<<endl;

cout<<"2. GIVE TEST"<<endl;

cout<<"3. GET RESULT"<<endl;

cout<<"4. ADMIN LOGIN"<<endl;

cout<<"5. EXIT"<<endl<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<"ENTER YOUR CHOICE (1, 2, 3, 4):"<<endl;

cin>>main\_menu\_choice;

switch(main\_menu\_choice)

{

case 1 :

apply\_now();

break;

case 2 : give\_test();

break;

case 3 : get\_result();

break;

case 4 : admin\_login();

break;

case 5 : exit(0);

}

}

BIBLIOGRAPHY

Computer Science with C++ -Sumita Arora

StackOverflow.com

GitHub.com

Floobits.com