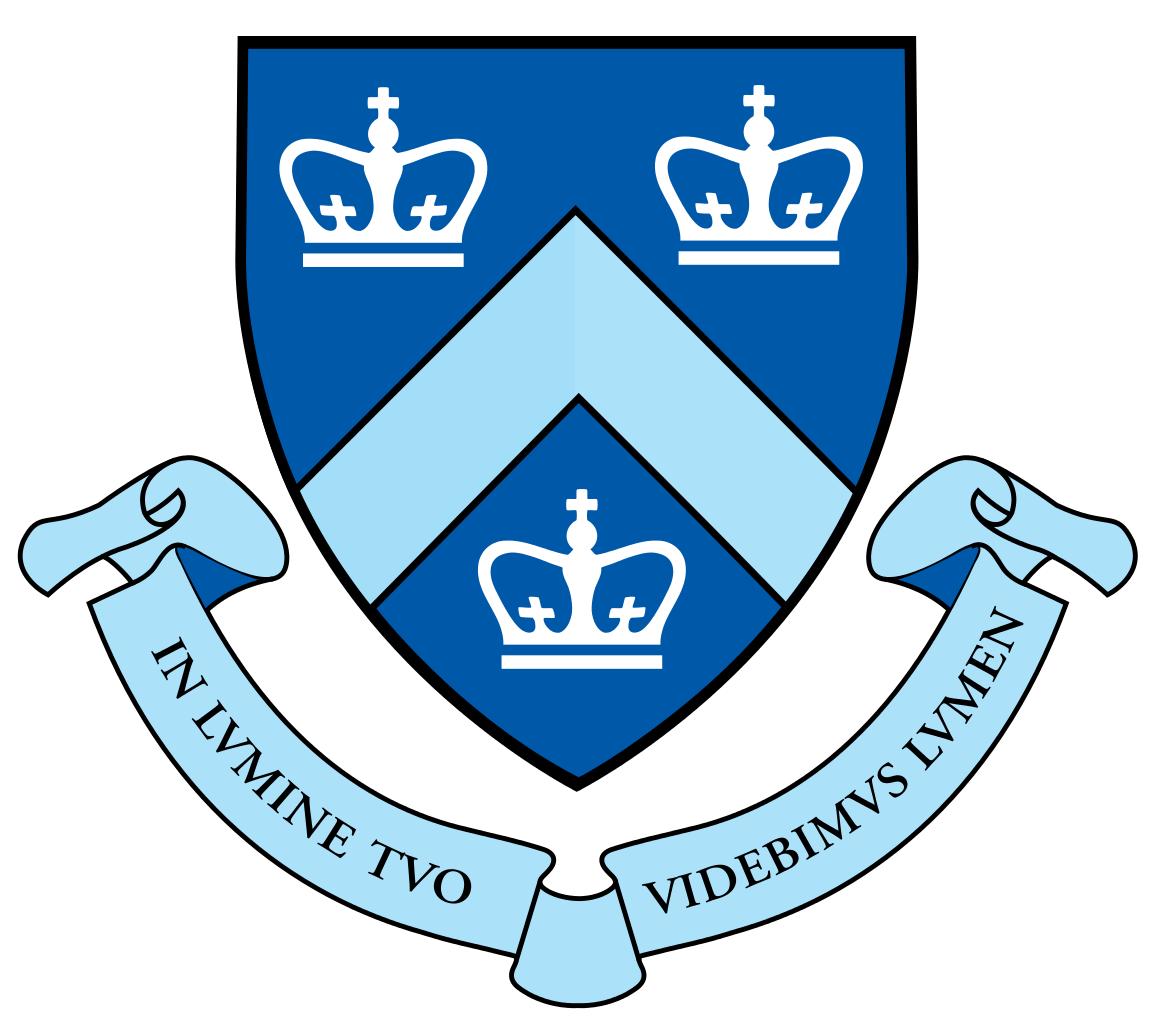


CollegeSimplify  
A tool to determine your compatibility score and to apply to colleges in the US



Certificate

This is to certify that this computer project has been completed by **Akshat Singh** of class **XII B** in partial fulfillment of the curriculum of the Central Board of Secondary Education leading to the award of the Senior Secondary Certificate for AISSCE Examination for the year 2018-2019

External Examiner Internal Examiner

Date: Date:

School Stamp Principal

Acknowledgement

I wish to express my deep gratitude and sincere thanks to the Principal Mrs. Avnita Bir, R.N.Podar for her encouragement and facilities that she provided for this project. I extend my hearty thanks to my Computer Science teacher, Mr Sameer Mohammad, who guided me to the successful completion of this project. I take this opportunity to express my deep sense of gratitude for his invaluable guidance, constant encouragement, constructive criticism, and immense motivation which has sustained my efforts at all stages of the project. Last but not the least; I extend my thanks to my family and friends for their constant support throughout the project

INDEX

1. Hardware and Software Specifications
2. Project Objective
3. Program Outline
4. Program Flowchart
5. Program Specifications
6. Source Code
7. Output
8. Limitations and Precautions
9. Bibliography

Hardware Specifications

This project was made on a system with the following specifications: -

1. Processor: Intel i5-6600T (2.70 GHz) i5-6th generation processor
2. RAM: 8 GB RAM
3. Operating System: Windows 10 (64 bit)

Software Specifications

The following software was used to create this program: -

1. Borland C++ 5.02

Project Objective

In the recent years, US Embassies have witnessed a surge in the number of Indian students applying for an F-1 Visa to pursue undergraduate studies in the United States. The process of getting admitted to a university in the United States, however, is difficult. Often, students waste their application fee in applying to the college not suited for their needs. This calls for a need to have an efficient system which will let the student know of his own capability in the form of a score. This score can be compared with the average student rating of a college to assist a student in finding the most suitable institute.









Program Outline

The program has been designed to function in a way similar to the CoalitionApplication portal and the popular website Prepscholar. A user can create his account, calculate his profile score, compare it with those of the universities, and can apply to the universities. The program also displays a receipt at the end of the code.





Program Flowchart

Program Specifications

1. List of Header Files used in the project
2. iostream.h – for cin and cout
3. fstream.h – for file handling
4. stdio.h – for gets and puts
5. conio.h – for the getch function
6. windows.h – for the Sleep function
7. List of Files used in the project
8. Data.txt
9. University\_List.txt
10. List of Classes used in the project
11. Data

class Data

{

private:

char username[30];

long accountID;

char name[30];

char highschool [30];

long phno;

public:

void Create\_Account ();

long return\_accountID ();

char\* return\_Name ();

char\* return\_highschool ();

};

1. student

class student

{

private:

int math,phy,eng,elec,chem;

char name[20];

char minsub[8];

int min;

int extracurr[5];

int noext;

int sat;

float gpa;

int ap[5];

int achiev[5];

int noap;

int noac;

char opt;

float score;

float achievscore;

float apscore;

float extrascore;

float satscore;

public:

student ();

void enterdetails ();

void calgpa ();

float scorecal();

};

1. List of independent member functions used in the program
2. void LoadScreen ()
3. void write\_Account ()
4. bool Login ()
5. void Apply ()
6. void main ()

Source Code

#include<iostream.h>

#include<stdio.h>

#include<fstream.h>

#include<conio.h>

#include<windows.h>

void LoadScreen()

{

cout<<"Loading";

for (int i=0;i<4;i++)

{

Sleep (500);

cout<<".";

}

cout<<endl;

}

class Data

{

public:

char username[30];

long accountID;

char name[30];

char highschool [30];

long phno;

void Create\_Account()

{

long veri;

cout<<"Let us start with your basic details \n";

cout<<"Name: ";

gets (name);

cout<<"Desired Username: ";

gets (username);

cout<<"Desired AccountID (NUMBERS ONLY): ";

cin>>accountID;

cout<<"Re-enter AccountID for verification: ";

cin>>veri;

while (accountID!=veri)

{

cout<<"IDs do not match!\n";

cout<<"Re-enter AccountID for verification: ";

cin>>veri;

}

cout<<"Current Highschool: ";

gets (highschool);

cout<<"Phone Number: ";

cin>>phno;

cout<<"Account Created!";

}

long return\_accountID()

{

return accountID;

}

char\* return\_Name()

{

return name;

}

char\* return\_highschool ()

{

return highschool;

}

};

void write\_Account ()

{

Data dat;

fstream file;

file.open ("Data.txt", ios::app);

dat.Create\_Account();

file.write ((char\*)&dat, sizeof(Data));

file.close();

}

bool Login(int x)

{

Data dat;

fstream File;

bool log=false;

File.open ("Data.txt", ios::in|ios::out);

while(!File.eof())

{

File.read((char\*)&dat, sizeof(Data));

if (dat.return\_accountID()==x)

{

cout<<"Login Successful\n\n";

log=true;

getch();

LoadScreen();

clrscr();

cout<<"Welcome! ";

puts(dat.return\_Name());

puts(dat.return\_highschool());

getch();

break;

}

}

return log;

}

void Apply()

{

int n, cost=0; char unames[100][100], course[100][100], email[30], gcemail[30], gcname[30], rname[30], rdesig[20], remail[30];

cout<<"Number of Colleges you wish to apply to: ";

cin>>n;

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

{

cout<<"Name of University "<<(i+1)<<": ";

gets (unames[i]);

cout<<"Desired Course at this college: ";

gets (course[i]);

cost=cost+4900;

}

cout<<"Most colleges will send you a separate college application after you are done submitting where you can elaborate the details of your application\n";

getch();

cout<<"Enter your Email to get the college specific email: ";

gets(email);

cout<<"Colleges in the United States also require your Guidance Counsellor to submit School Reports\n";

getch();

cout<<"Enter your Guidance Counsellor's name: ";

gets(gcname);

cout<<"Enter Guidance Counsellor's Email: ";

gets(gcemail);

LoadScreen();

cout<<"Invite Sent!";

cout<<"You're also required to add one compulsory recommender\n";

cout<<"Recommenders Name: ";

gets (rname);

cout<<"Recommenders Designation: ";

gets (rdesig);

cout<<"Recommenders Email: ";

gets (remail);

cout<<"Your basic application is done! Your invoice will be printed in the next dialogue box\n";

getch();

LoadScreen();

clrscr();

cout<<"S. No. Name Course Application Fee"<<endl;

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

{

cout<<(i+1)<<" "<<unames[i]<<" "<<course[i]<<" Rs. 4990"<<endl;;

}

cout<<" Total Fee: "<<cost;

cout<<endl;

cout<<endl;

system ("pause");

}

class student

{

private:

int math,phy,eng,elec,chem; //subject scores

char name[20];

char minsub[8]; // subject name of worst scores

int min; // min score to be excluded from gpa

int extracurr[5];

int noext;

int sat;

float gpa; // gpa calculated on the best 4 marks

int ap[5]; // grades of ap classes

int achiev[5]; // no of achievements

int noap; // no of ap classes

int noac; // no of extracurricular activities

char opt; // miscellaneous use for binary operations

float score; // comprehensive score based on gpa,achievements,and ap classes

float achievscore;

float apscore;

float extrascore;

float satscore;

public:

student()

{

min=math=phy=noap=noac=eng=elec=chem=0;

score=achievscore=satscore=apscore=extrascore=0.0;

opt='n';

for(int i=0;i<4;i++)

{

strcpy(minsub,"null");

}

for(int i=0;i<5;i++)

{

ap[i]=0;

achiev[i]=0;

extracurr[i]=0;

}

}

void enterdetails()

{

LoadScreen();

clrscr();

cout<<"Your score will be calculated on your extra-curriculars, Standardized Tests, and GPA\n";

cout<<"GPA will be calculated on your Senior Year Scores (Predicted or Final)\n";

cout<<"Mathematics (out of 100): ";

cin>>math;

cout<<"Physics (out of 100): ";

cin>>phy;

cout<<"Chemistry (out of 100): ";

cin>>chem;

cout<<"English (out of 100): ";

cin>>eng;

cout<<"Elective (out of 100): ";

cin>>elec;

cout<<"SAT score(out of 1600): ";

cin>>sat;

cout<<"Has the student participated in any extracurricular activity? Y for Yes/N for No: ";

cin>>opt;

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

{

cout<<"Enter the number of such activities: ";

cin>>noext;

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

{

cout<<"Enter 1 if Activity "<<(i+1)<<" is Organizational, 2 if it is volunteering, and 3 for others: ";

cin>>extracurr[i];

}

}

cout<<"Has the student taken any AP classes?Y for Yes/N for No: ";

cin>>opt;

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

{

cout<<"How many ap classes has the student taken (maximum 5): ";

cin>>noap;

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

{

cout<<"Enter marks(out of 5) for AP Test No:"<<(i+1)<<": ";

cin>>ap[i];

}

}

cout<<"Does the student possess any award which is at least state level? Y for Yes/ N for No: ";

cin>>opt;

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

{

cout<<"Enter number of achievements (maximum 5): ";

cin>>noac;

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

{

cout<<"Enter 1 if Activity "<<(i+1)<<" is International, Enter 2 if it is National level, and Enter 3 if it is State level: ";

cin>>achiev[i];

}

}

}

void calgpa()

{

if((math<=chem)&&(math<=eng)&&(math<=elec)&&(math<=phy))

{

strcpy(minsub,"Mathematics");

gpa=((math+chem+phy+eng)/4);

}

else if((chem<=math)&&(chem<=eng)&&(chem<=elec)&&(chem<=phy))

{

min=chem;

strcpy(minsub,"Chemistry");

gpa=((math+elec+phy+eng)/4);

}

else if((phy<=math)&&(phy<=eng)&&(phy<=elec)&&(phy<=chem))

{

min=phy;

strcpy(minsub,"Physics");

gpa=((math+chem+elec+eng)/4);

}

else if((eng<=math)&&(eng<=chem)&&(eng<=elec)&&(eng<=phy))

{

min=eng;

strcpy(minsub,"English");

gpa=((math+chem+phy+elec)/4);

}

else if((elec<=chem)&&(elec<=eng)&&(elec<=math)&&(elec<=phy))

{

min=elec;

strcpy(minsub,"Elective");

gpa=(math+chem+phy+eng)/6;

}

cout<<"Do you wish to see a GPA report y/n: ";

cin>>opt;

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

{

cout<<"\n Worst subject is ";

puts(minsub);

cout<<" in which marks obtained: "<<min<<endl;

cout<<"GPA calculated using top 4 subjects(out of 100): "<<gpa<<endl;

}

}

float scorecal()

{

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

{

if(achiev[i]==1)

achievscore+=20;

else if(achiev[i]==2)

achievscore+=10;

else

achievscore+=5;

}

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

{

if(extracurr[i]==1)

extrascore+=20;

else if(extracurr[i]==2)

extrascore+=10;

else

extrascore+=5;

}

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

{

apscore+=(ap[i]\*4);

}

satscore=(sat\*100/1600);

score=((gpa\*25/100)+(achievscore\*20/100)+(apscore\*10/100)+(satscore\*20/100)+(extrascore\*25/100));

cout<<"The student's comprehensive score is: "<<score<<endl;

return score;

}

};

void main()

{

fstream file, f, f1;

Data dat;

cout<<"Welcome to CollegeSimply India! The perfect guide for Indian students aspiring to study abroad"<<endl;

getch();

int c, cs1; bool flag = true;

while (flag==true)

{

cout<<"Press 1 if you're a new user\n";

cout<<"Press 2 if you're an existing user\n";

cout<<"Press 3 to exit\n";

cout<<"Enter your choice: ";

cin>>c;

switch (c)

{

case 1:

write\_Account();

break;

case 2:

bool che;

long acid;

cout<<"Account ID: ";

cin>>acid;

che = Login(acid);

if (che==false)

{

cout<<"Login Failed!\n";

}

while (che==true)

{

cout<<"\n\n1. Check your college compatibility score\n";

cout<<"2. View the entire college list\n";

cout<<"3. Submit your CollegeSimply Application to the \n";

cout<<"4. Sign Out";

cout<<"Enter your choice: ";

cin>>cs1;

clrscr();

switch (cs1)

{

case 1:

{

student obj;

float par;

obj.enterdetails();

obj.calgpa();

par = obj.scorecal();

cout<<"Your comprehensive score has been calculated. Press any key to go on and find the best universities for you...\n";

getch ();

cout<<"Displaying full list to help you pick your best fit";

getch();

LoadScreen();

cout<<"\n\n";

}

case 2:

char x[500];

file.open ("University\_List.txt", ios::in);

while (!file.eof())

{

file.getline (x, 400, '\n');

Sleep(100);

puts(x);

}

file.close();

break;

default:

cout<<"Invalid Input!";

break;

case 3:

LoadScreen();

clrscr();

cout<<"Welcome to the College Application Section of CollegeSimply (Powered by Prepscholar)\n";

getch();

Apply();

break;

case 4:

char conf;

cout<<"Are you sure you want to sign out of the application? Y for Yes, N for NO";

cin>>conf;

if (conf=='N'||conf=='n')

break;

else if (conf=='Y'||conf=='y')

che=false;

else

cout<<"Invalid Input!";

}

}

break;

case 3:

char e;

cout<<"Are you sure you want to quit? Y for Yes and N for No: ";

cin>>e;

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

{

cout<<"Thank you for using CollegeSimply";

Sleep (2000);

flag=false;

}

else if (e=='n'||e=='N')

break;

else

cout<<"Invalid Input!";

break;

default:

cout<<"Invalid Input!";

break;

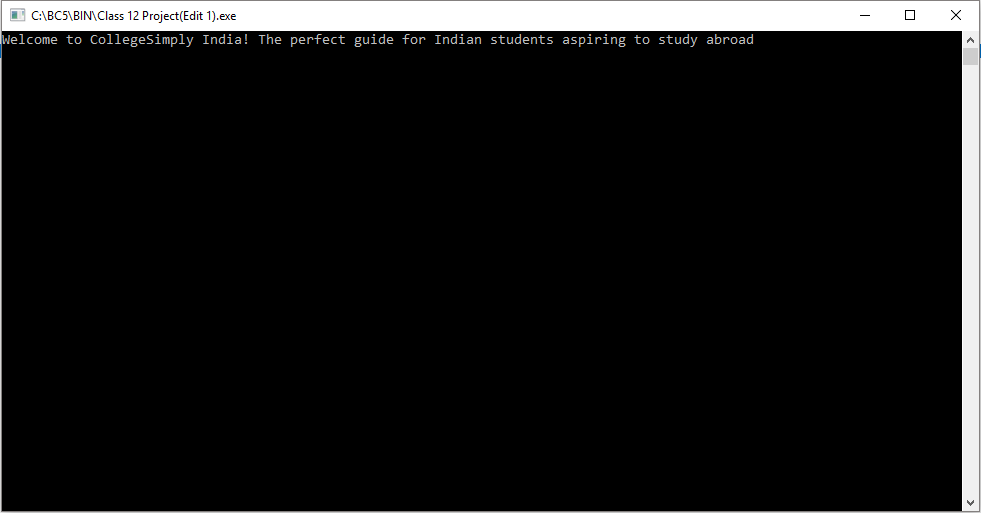
}

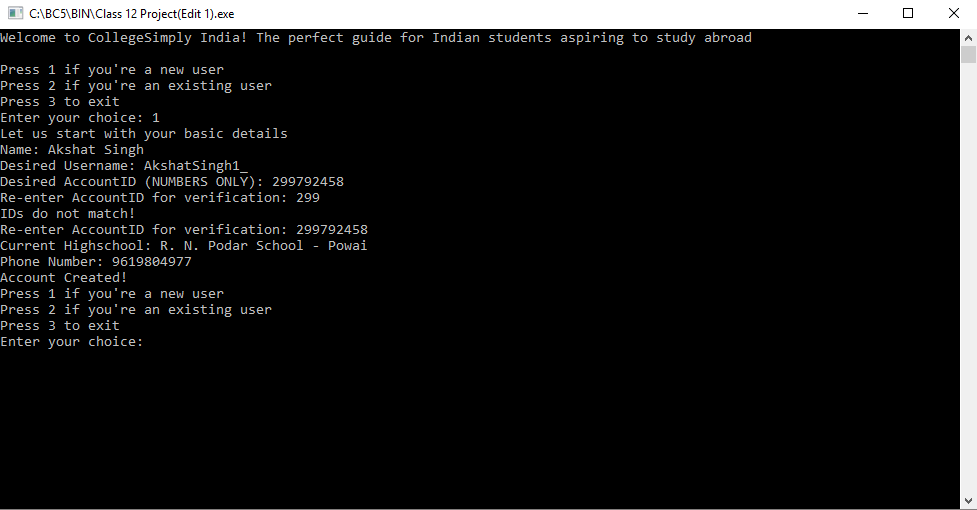
}

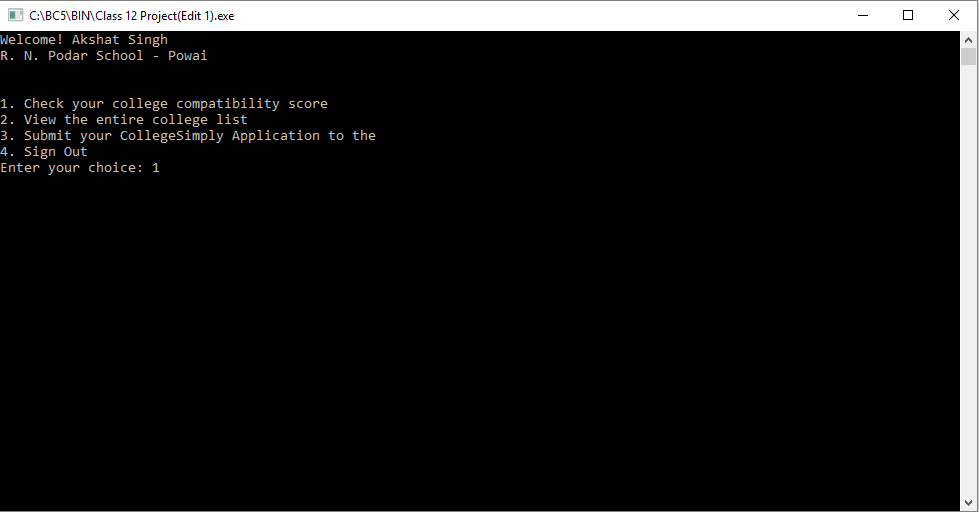
}

Output

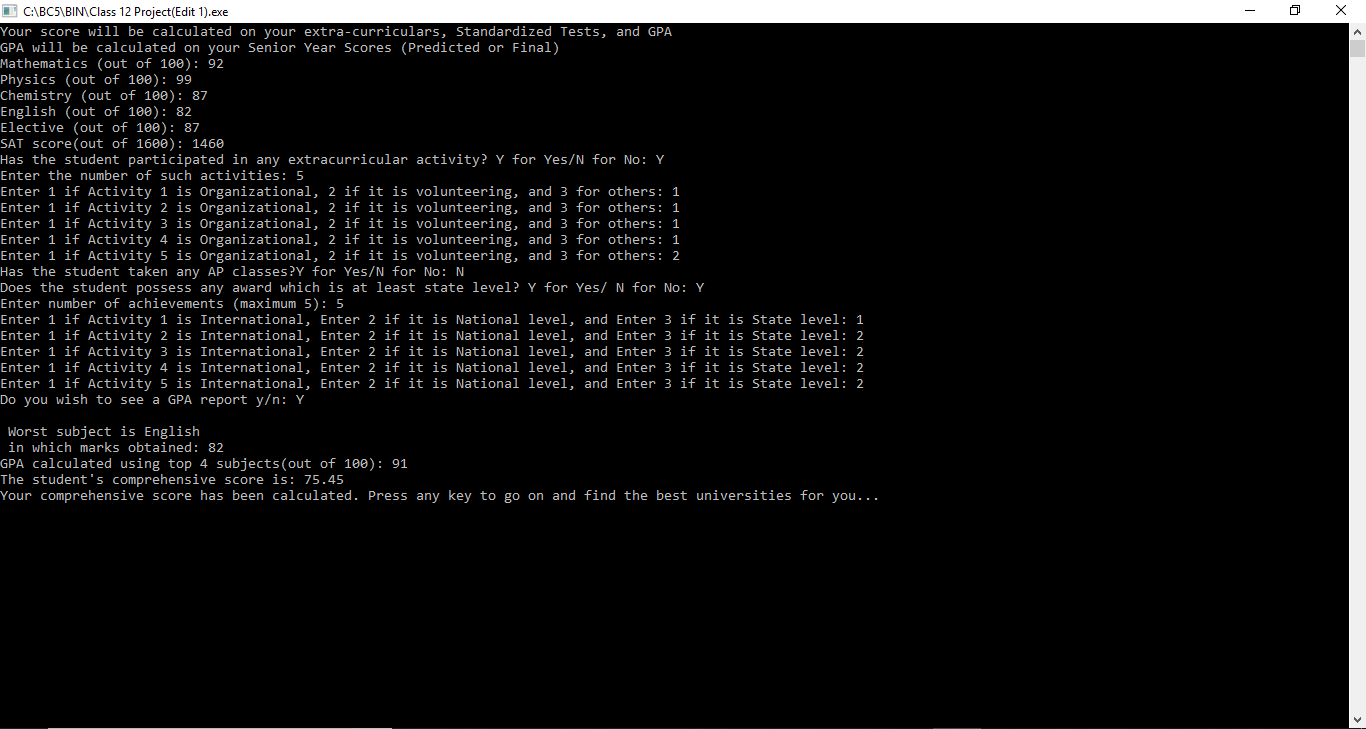
* The user can either register himself, or use an existing account as shown in the following steps



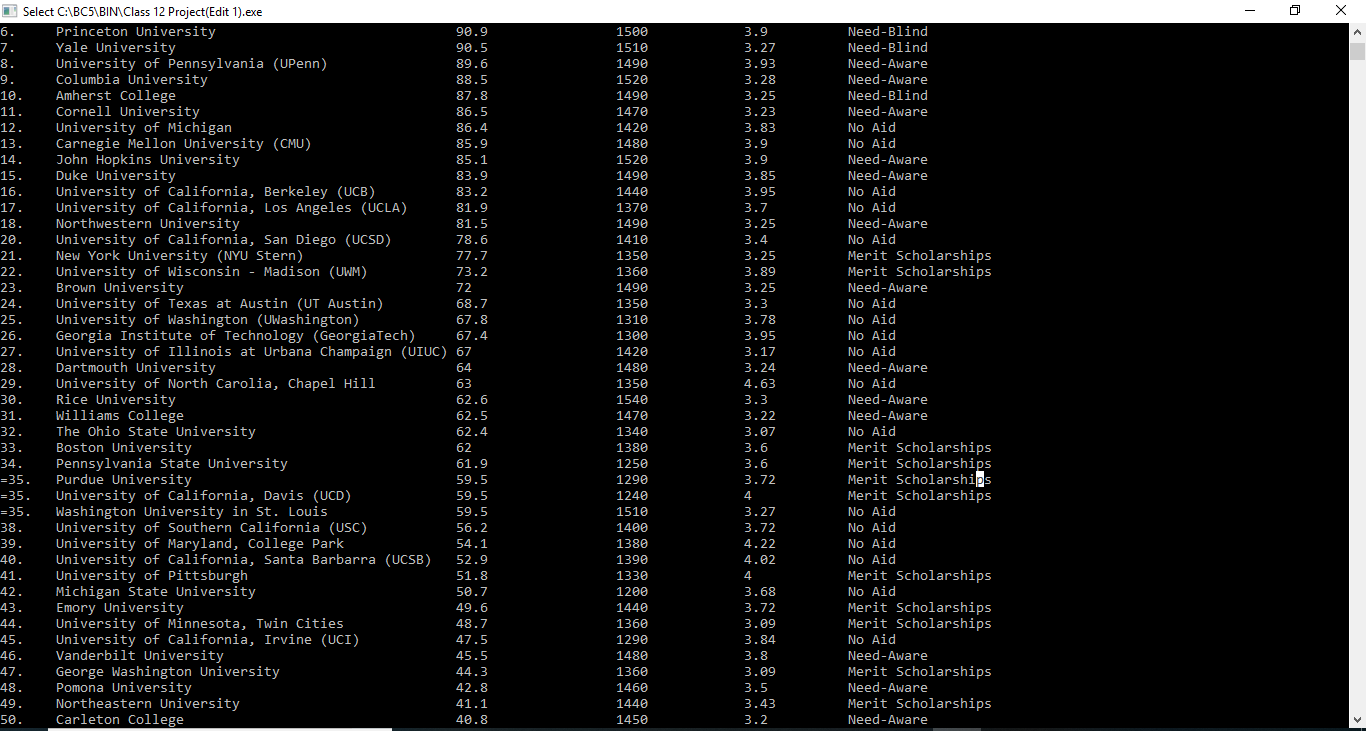
1. 



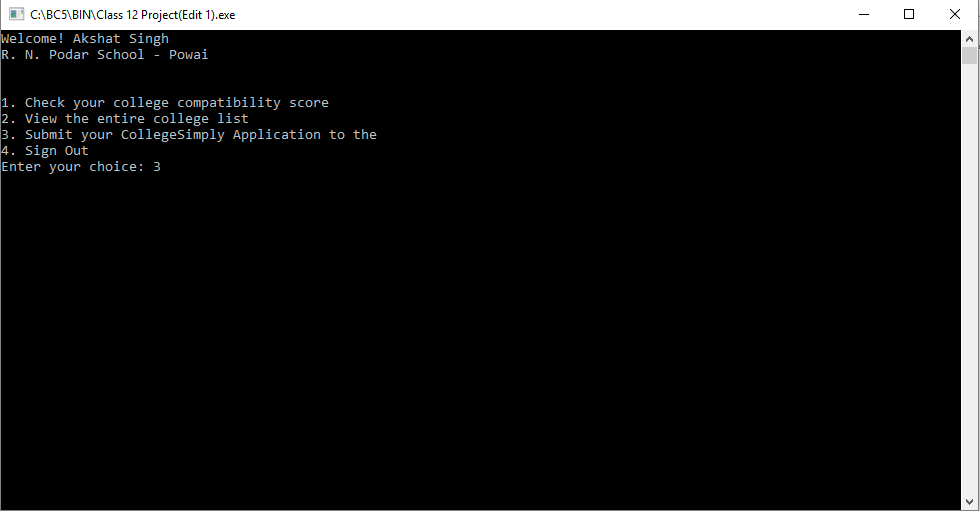
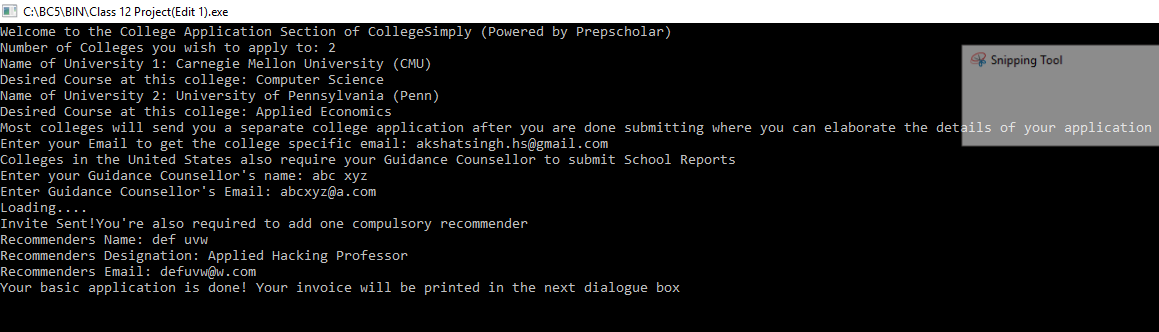
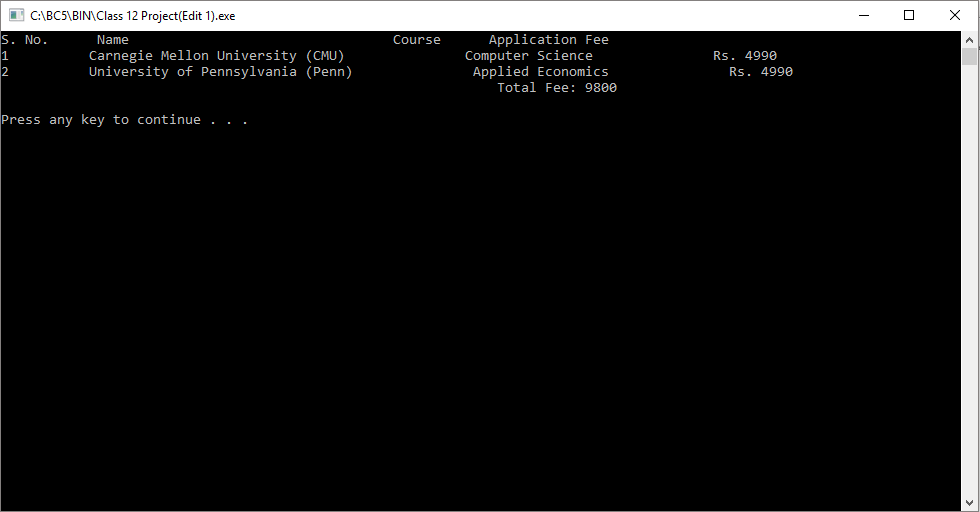
* In the following steps, the student’s profile score is calculated

1. 

* The list of all the partner institutions is displayed. A student may compare his score with the comprehensive average student score of his desired institute

1. 

* The user may also send an application directly

1. 
2. 
3. 

Limitations

1. The code only includes the 50 best American Universities
2. The application portal’s interface used here resembles the Canadian OUAC portal more closely than it resembles the American CommonApp or CoalitionApp.
3. The program allows the user to enter a maximum of 6 extra-curricular activities and 5 awards/honors. Most application portals allow users to enter upto 10 extra-curricular activities and awards.

Precautions

1. The university list is displayed coherently only when the terminal window is maximized.
2. On some C++ compilers, the large volume of the code may lead to a system crash.

Bibliography

1. [www.prepscholar.com](http://www.prepscholar.com)
2. [www.collegesimply.com](http://www.collegesimply.com)
3. [www.mycoalition.org](http://www.mycoalition.org)
4. [www.ivyleague.com](http://www.ivyleague.com)
5. [www.topuniversities.com](http://www.topuniversities.com)
6. [www.mit.edu](http://www.mit.edu)
7. [admission.enrollment.cmu.edu/pages/undergraduate-admission-statistics](https://admission.enrollment.cmu.edu/pages/undergraduate-admission-statistics)