COMPUTER SCIENCE

PROJECT

  ****

**SCHOOL MANAGEMENT**

**Harshul Agarwal**

**CLASS –XII A**

**ACKNOWLEDGEMENT**

**I HARSHUL AGARWAL WOULD LIKE TO GREET BY SAYING – ACKNOWLEDGING MY PARENTS WHO GAVE ME SUGGESTIONS FOR THE IMPROVEMENT OF ALL ASPECT IN THIS PROJECT .**

**IN PARTCULARLY ,I WOULD LIKE TO THANK MY COMPUTER SCIENCE TEACHER MS. DEEPALI FOR ALLOWING ME TO CARRY THE PROJECT WORK AND ASSISTED.**

**I ALSO OWE A GREAT DEAL OF THANKS TO MY PARENTS WHO ALSO BOOSTED UP MY MORALS AND ENCOURAGED ME DURING THE SUGGESTION PROJECT. I HAVE A GREAT PLEASURE IN THANKING THEM ALL FOR THERE GENERIOUS HELP IN THIS PROJECT.**

**CERTIFICATE**

**THIS IS TO CERTIFY THAT HARSHUL AGARWAL OF CLASS 12TH . ROLL NO HAS SUCCESSFULLY COMPLETED PROJECT UNDER MY SUPERVISION AND TO MY SATISFACTION.`**

**(Sigature)**

**INTODUCTION**

This is C++ Projects on School Management System, which provided a lot of facility to their user. The objective and scope of my Project School Management System is to record the details various activities of user. It will simplify the task and reduce the paper work.

The School Management System eliminates most of the limitations of the existing software. It has the following objectives:

* **Enhancement:**

The main objective of School Management System is to enhance and upgrade the existing system by increasing its efficiency and effectiveness. The software improves the working methods by replacing the existing manual system with the computer-based system.

* **Accuracy:**

The School Management System provides the uses a quick response with very accurate information regarding the users etc. Any details or system in an accurate manner, as and when required.

* **User-Friendly:**

The software School Management System has a very user-friendly interface. Thus the users will feel very easy to work on it. The software provides accuracy along with a pleasant interface. Make the present manual system more interactive, speedy and user friendly.

* **Availability:**

The transaction reports of the system can be retried as and when required. Thus, there is no delay in the availability of any information, whatever needed, can be captured very quickly and easily.

* **Maintenance Cost:**

Reduce the cost of maintenance.

**HEADER FILES**

#include<fstream.h>

#include<conio.h>

#include<stdio.h>

#include<string.h>

#include<process.h>

#include<stdlib.h>

**PROGRAM CODE**

ifstream fin;

ofstream fout;

int f=0,c,i=0;

class st

{ int an;

char name[20];

char g;

int m;

public:

void getdata(); /\* To enter details of students and

assign the grade \*/

void display(); // To display details of students

int getan() // To return the value of admission no

{

return an;

}

void enter(); // To enter and store the details in file

void search(); // To search a record in file

void del\_data(); // To delete the record from file

void modify(); // To modify the record stored in file

}s,s1;

void st::getdata()

{

cout<<"Enter name: ";

gets(name);

cout<<"\n Enter admission no: ";

cin>>an;

cout<<"\n Enter marks: ";

cin>>m;

if(m>=90)

g='A';

else if(m>=70&&m<90)

g='B';

else

g='C';

}

void st::display()

{

cout<<"\n Name: ";

puts(name);

cout<<" Admission no: "<<an;

cout<<"\n Marks: "<<m;

cout<<"\n Grade: "<<g;

}

void st::enter()

{ ofstream fout("ri.txt",ios::app);

s.getdata();

fout.write((char\*)&s,sizeof(st));

fout.close();

}

void st::search()

{ f=0;

cout<<"\n Enter admission no. you want to search: ";

int r; cin>>r;

fin.open("ri.txt");

fin.seekg(0);

while(fin.read((char\*)&s,sizeof(st)))

{

int g;

g=s.getan();

if(g==r)

{f=1;

s.display();

break;

}

}

if(f==0)

cout<<"\n Not found ";

fin.close();

}

void st::modify()

{ fstream f("ri.txt",ios::in|ios::out);

int a,pos;

cout<<"enter admission no u want to modify ";

cin>>a;

while(!f.eof())

{ pos=f.tellg();

f.read((char\*)&s,sizeof(s));

if(a==s.getan())

{

cout<<"enter new details \n";

s.getdata();

f.seekg(pos);

f.write((char\*)&s,sizeof(s)) ;

i=1 ;

break;

}

}

if(i==0)

cout<<"not found";

cout<<"\n new file";

f.seekg(0);

while(f.read((char\*)&s1,sizeof(s1)))

s1.display();

f.close();

}

void st::del\_data()

{ fin.open("ri.txt");

ofstream fout("temp.txt",ios::app);

cout<<"\n Enter admission no. you want to delete ";

int q; cin>>q;

f=0;

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

{

if(s.getan()==q)

{

f=1;

s.display();

cout<<"\n Are you sure you want to delete?(y/n): ";

char u; cin>>u;

if(u=='n')

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

}

else

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

}

if(f==0)

cout<<"\n Record not found";

fin.close();

fout.close();

remove("ri.txt");

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

fin.open("ri.txt");

cout<<"\n Record deleted!!";

cout<<endl<<"new record";

while(!fin.eof())

fin.read((char\*)&s1,sizeof(s1)) ;

s1.display();

fin.close();

}

void main()

{ clrscr();

int z=0;

char n,pa[20];

a: cout<<"enter passward to access: ";

gets(pa);

clrscr();

if(strcmp(pa,"apsvp")==0)

{ cout<<"\n \n \t \t WELCOME !!!!!"<<endl<<endl;

do

{ menu:

cout<<"\n 1. Want to insert data";

cout<<"\n 2. Want to search data";

cout<<"\n 3. Want to delete record";

cout<<"\n 4. Want to modify record";

cout<<"\n 5. Exit";

cout<<"\n Enter your choice: ";

cin>>c;

clrscr();

switch(c)

{ case 1:

{ s.enter();

break;

}

case 2:

{

s.search();

break;

}

case 3:

{ s.del\_data();

break;

}

case 4:

{s.modify();

break;

}

case 5:

{exit(0);

}

}

}while(c>=1&&c<=5);

}

else

{

while(z<4)

{ cout<<"\n wrong passward";

cout<<"\n want to try again(y/n)";

cin>>n;

z++;

if(n=='y')

goto a;

else

exit(0);

}

exit(0);

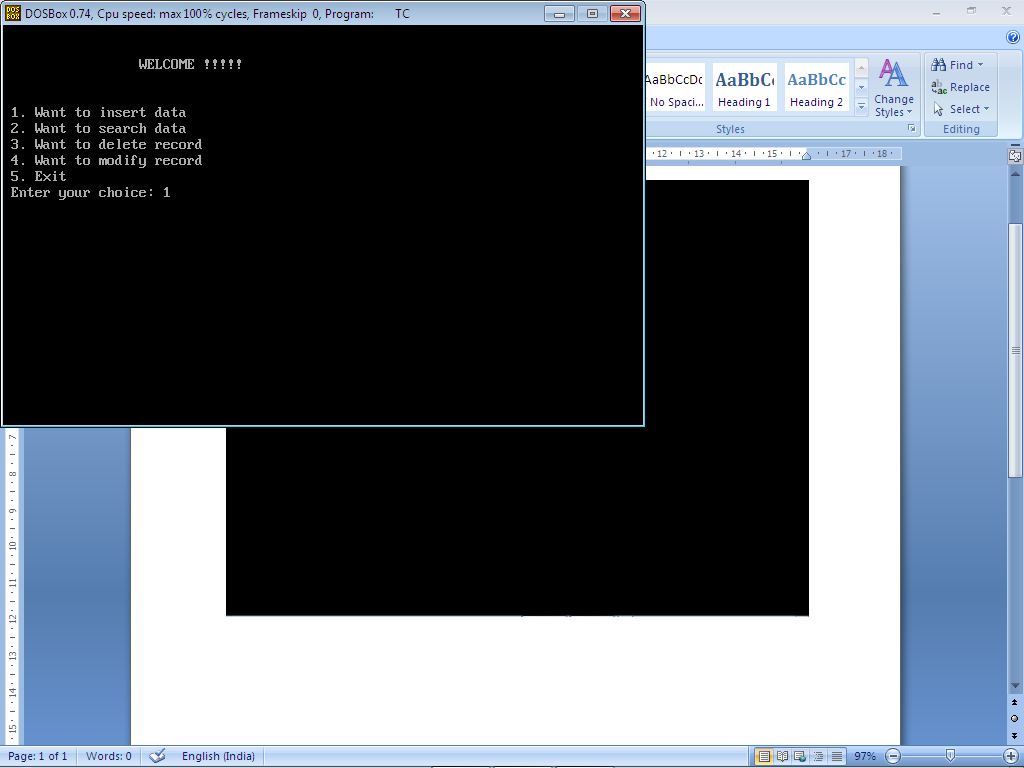
}

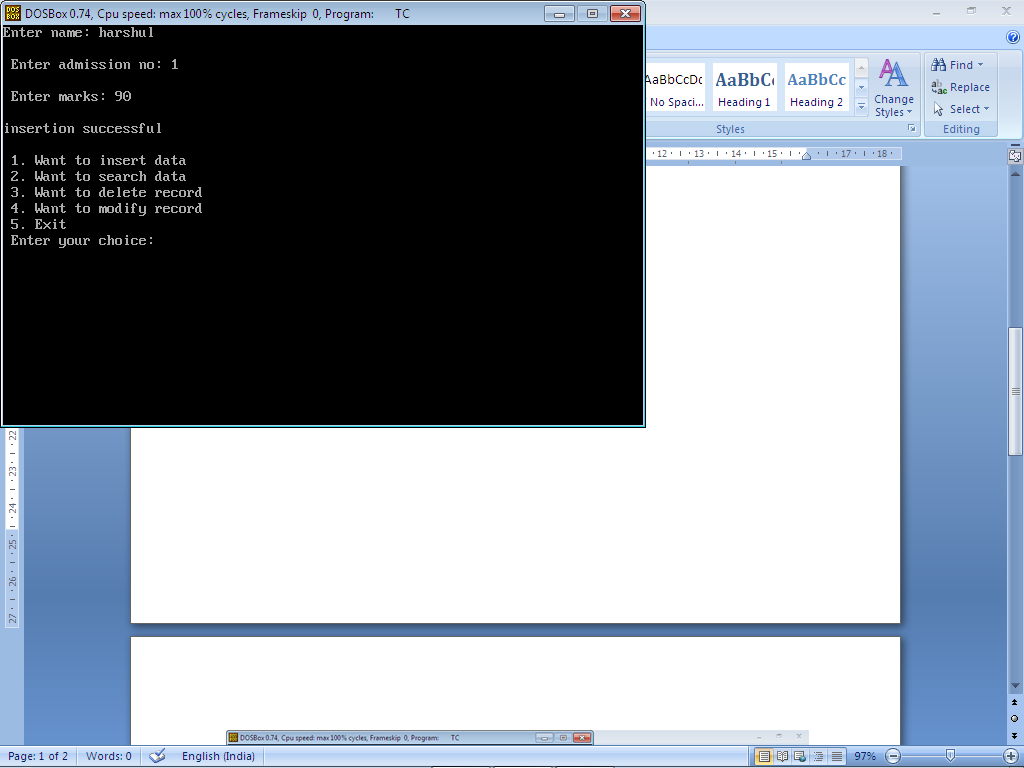
getch();

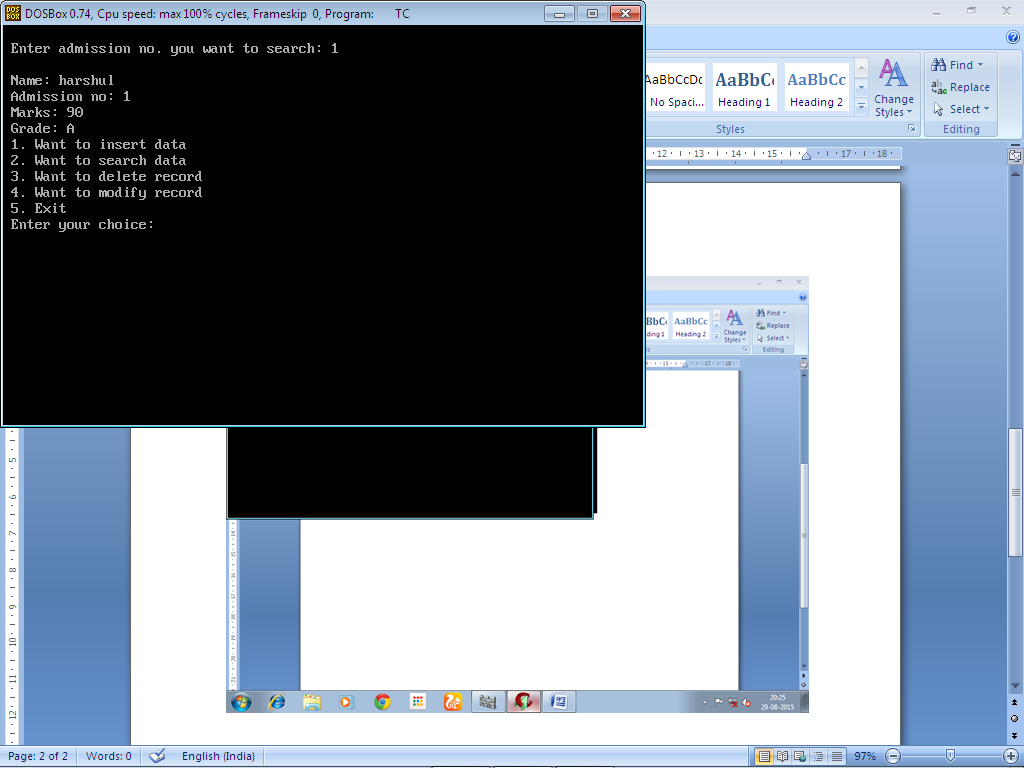
}

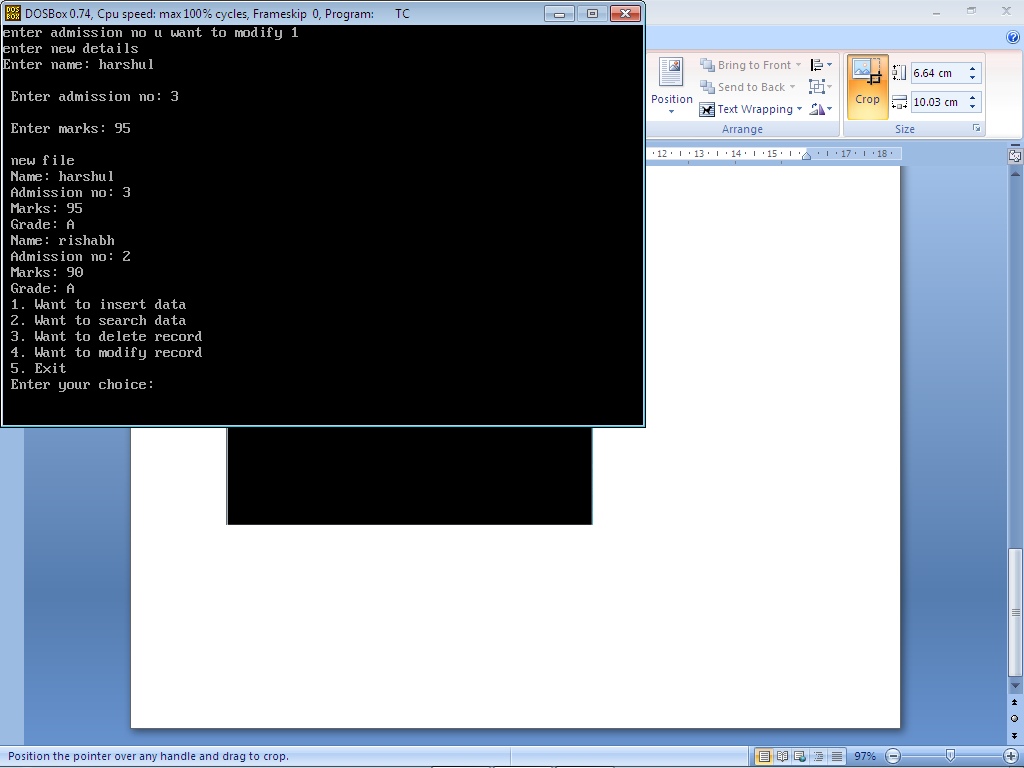
**OUTPUTS**

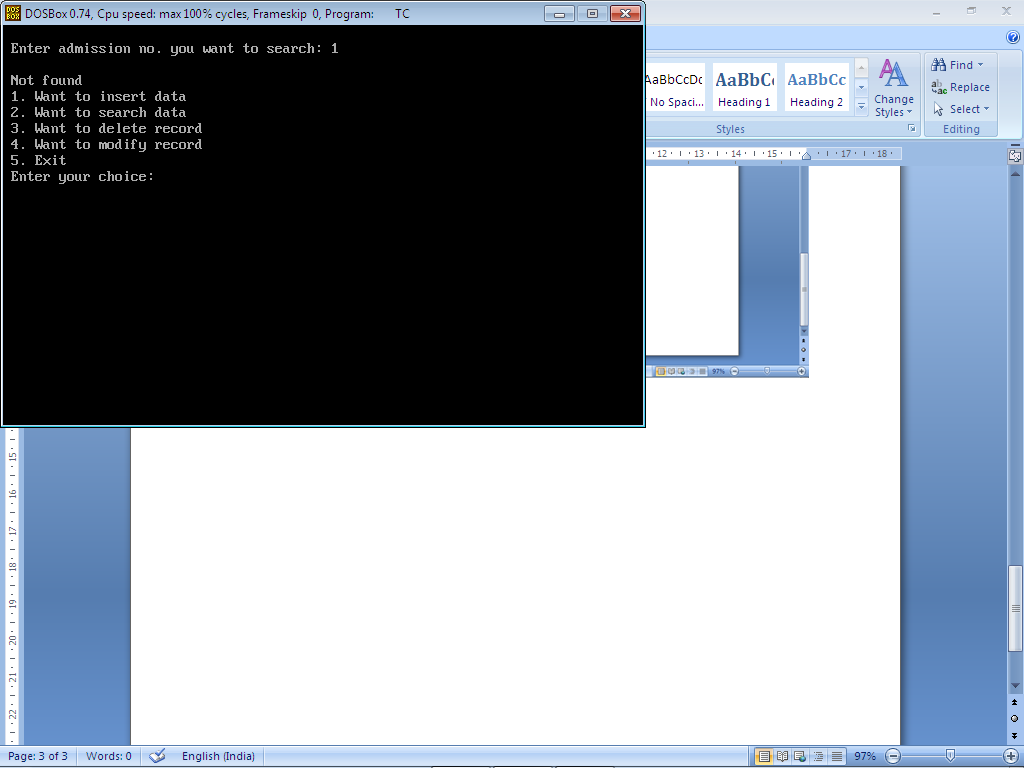


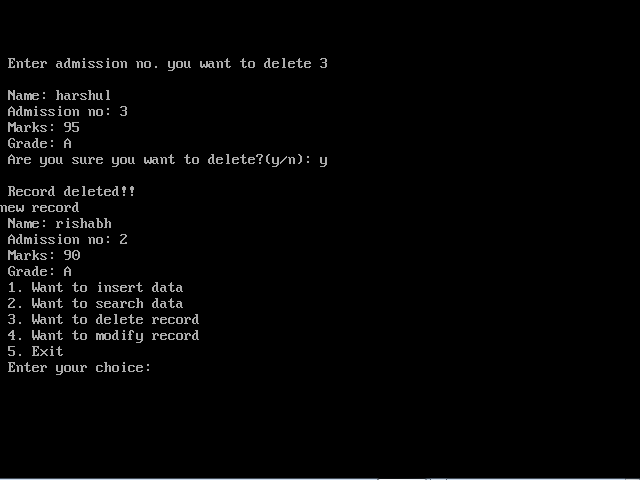


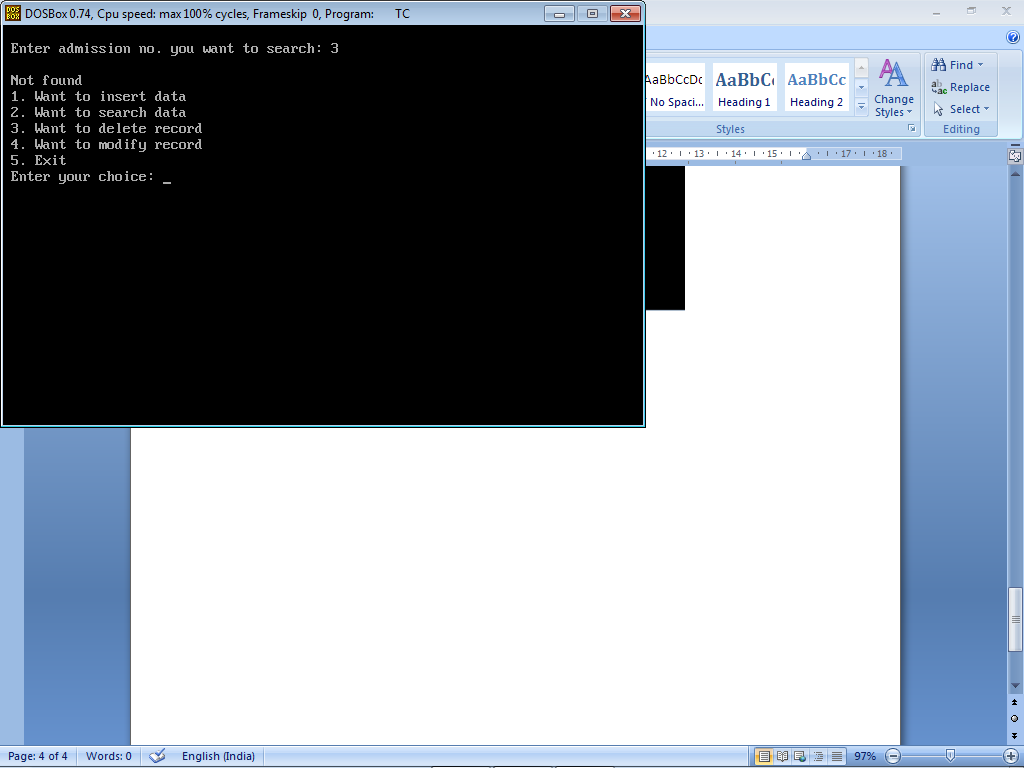












**BIBLIOGRAPHY**

* **SUMITA ARORA TEXTBOOK FOR CLASS XII**
* **WWW.ICBSE.COM**