COMPUTER SCIENCE PROJECT

CRIMINAL DATABASE

BY K.JEROME

NEGLIGENCE MAY LEAD TO LOSSES.

INDEX

- ABOUT THE PROJECT
- OOPS CONCEPTS
- HEADER FILES USED
- FUNCTIONS USED
- SOURCE CODE
- OUTPUT
- CONCLUSION
- BIBILIOGRAPHY

ABOUT THE PROJECT

A criminal record is simply information that's kept about a person's arrests and convictions. State, local and federal authorities store and use them for many purposes. They can be used for identification and to locate possible suspects in unsolved cases.

Courts can also use criminal records to determine sentences for crimes that are committed by that same person at a later time. Another use for criminal records, one that's becoming more frequent in recent years, comes in the form of background checks. Similar to the example that you just read, a background check is often used to determine if a person applying for a job has ever committed a crime.

A criminal record begins when a suspect is arrested for a crime. The person is fingerprinted and photographed. All of his or her personal information is recorded along with the information about the current arrest. If he or she is convicted of the crime, that information is also stored.

Criminal records began as handwritten or typed files that were kept at local police stations. Obviously, that type of system had its downside. Sharing of information contained within these files rarely occurred between police agencies. It was fairly easy for a person to escape his or her criminal past by simply moving to another city, state or even out of the country.

Criminal records are now stored in massive computer databases that are accessible throughout the world. As you can imagine, this ease of access makes it makes it simpler for law enforcement agencies to keep a detailed record a person's criminal activity.

OOPS CONCEPTS

The prime purpose of C++ programming was to add object orientation to the C programming language, which is in itself one of the most powerful programming languages.

The core of the pure object-oriented programming is to create an object, in code, that has certain properties and methods. While designing C++ modules, we try to see whole world in the form of objects. For example a car is an object which has certain properties such as color, number of doors, and the like. It also has certain methods such as accelerate, brake, and so on.

There are a few principle concepts that form the foundation of object-oriented programming:

Object

This is the basic unit of object oriented programming. That is both data and function that operate on data are bundled as a unit called as object.

Class

When you define a class, you define a blueprint for an object. This doesn't actually define any data, but it does define what the class name means, that is, what an object of the class will consist of and what operations can be performed on such an object.

Abstraction

Data abstraction refers to, providing only essential information to the outside world and hiding their background details, i.e., to represent the needed information in program without presenting the details.

For example, a database system hides certain details of how data is stored and created and maintained. Similar way, C++ classes provides different methods to the outside world without giving internal detail about those methods and data.

Encapsulation

Encapsulation is placing the data and the functions that work on that data in the same place. While working with procedural languages, it is not

always clear which functions work on which variables but object-oriented programming provides you framework to place the data and the relevant functions together in the same object.

Inheritance

One of the most useful aspects of object-oriented programming is code reusability. As the name suggests Inheritance is the process of forming a new class from an existing class that is from the existing class called as base class, new class is formed called as derived class.

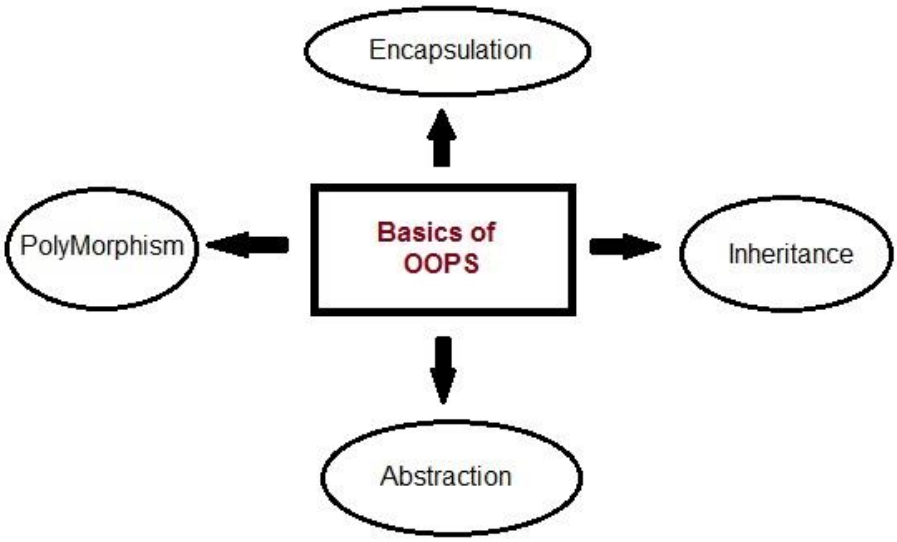
This is a very important concept of object-oriented programming since this feature helps to reduce the code size.

Polymorphism

The ability to use an operator or function in different ways in other words giving different meaning or functions to the operators or functions is called polymorphism. Poly refers to many. That is a single function or an operator functioning in many ways different upon the usage is called polymorphism.

Overloading

The concept of overloading is also a branch of polymorphism. When the exiting operator or function is made to operate on new data type, it is said to be overloaded.



HEADER FILES USED

Conio.h- conio.h header used in c programming contains functions for console input/output. Some of the most commonly used functions of conio.h are clrscr, getch, getche, kbhit etc. Functions of conio.h can be used to clear screen, change color of text and background, move text, check if a key is pressed or not and many more.

Stdio.h- Most of the C file input/output functions are defined in stdio.h (or in the C++ header cstdio, which contains the standard C functionality but in the std namespace).

Fstream.h- Contains function prototypes for functions that perform input from files on disk and output to files on disk.

String.h- Contains function prototypes for C-style string-processing functions.strcpy(),strlen(),strcat etc., are examples of string functions.

Stdlib.h- stdlib.h is the header of the general purpose standard library of C programming language which includes functions involving memory allocation, process control, conversions and others. The name "stdlib" stands for "standard library".

Iomanip.h- The header iomanip is part of the Input/output library of the C++ Standard Library. It defines the manipulator functions resetiosflags(), setiosflags(), setbase(), setfill(), setprecision(), and setw(). These functions may be conveniently used by C++ programs to affect the state of iostream objects.

Dos.h- dos.h header file of c language contains functions for handling interrupts, producing sound, date and time functions etc. It is borland specific and works in turbo c compiler.

FUNCTIONS USED

ncheck(char arr[])- FUNCTION TO CHECK THAT INPUT IS NOT NUMERIC.

s_check(char a[])- FUNCTION TO CHECK THAT THE INPUT FOR

SEX HAS ONLY 1 CHARACTER IN M OR F ONLY

input(void)-FUNCTION TO INPUT CRIMINAL RECORD

output(void)- FUNCTION TO DISPLAY CRIMINAL RECORD

delete_rec(char con_no[])-FUNCTION TO DELETE A CRIMINAL RECORD

dcheck(char n[])-FUNCTION TO CHECK WHETHER THE CONVICT NO IS PRESENT OR NOT AND RETURN THE RECORD'S POSITION

display()-FUNCTION TO DISPLAY REQUESTED CRIMINAL RECORD USING output(void) FUNCTION

remove()-FUNCTION TO CHECK FOR THE GIVEN CONVICT NO AND CALL delete_rec(char con_no[]) FUNCTION TO DELETE

menu()-FUNCTION TO DISPLAY MENU

modify()-FUNCTION TO MODIFY SPECIFIED CRIMINAL RECORD

list()-FUNCTION TO DISPLAY THE ENTIRE CRIMINAL RECORDS

empty()-FUNCTION TO DECLARE ALL VARIABLES AND GIVE NULL VALUE TO THEM

concheck(char b[])-FUNCTION TO CHECK CONVICT NO

SOURCE CODE

```
#include<iostream.h>
#include<conio.h>
#include<stdio.h>
#include<fstream.h>
#include<string.h>
#include<stdlib.h>
#include<iomanip.h>
#include<dos.h>
/*****************************************************************
**********************Structure for Date**************************
*****************************************************************/
struct dat
{
int d;
int m;
int y;
};

/*****************************************************************
*                Structure for Details of Crime of Criminal      *
*****************************************************************/
struct court
{
char  crime[20];
char  act[20];
char cour[20];
char state[20];
};

/*****************************************************************
*                        Structure for Physical Description of Criminal      *
*****************************************************************/
struct descp
{

char colour[20];
char face[20];
char imark[20];        //IDENTIFICATION MARKS(DEFORMITIES ETC.)
char  height[6];
```

```
};

/*****************************************************************
*                          Structure  for Details of F.I.R.      *
*****************************************************************/
struct police
{
char station[50];
char fir_no[10];
int d;
int m;
int y;
};

/*****************************************************************
*      Class ' CRIMINAL ' for getting all information about Convict   *
*****************************************************************/
class criminal
{

public:
char   name[20];
char   sex[3];
char    conno[10];
char    age[4];
char   address[50];          //ADDRESS OF CRIMINAL BEFORE ARREST.
court  COURT;
police POLICE;
descp  des;
int ncheck(char arr[]);  // FUNCTION TO CHECK THAT INPUT IS NOT
//    NUMERIC.

int  s_check(char  a[]);     // FUNCTION  TO  CHECK  THAT  THE  INPUT
FOR

//  SEX HAS ONLY 1 CHARACTER IN M OR F ONLY
void input(void);
void output(void);
void delete_rec(char con_no[]);
int dcheck(char n[]);
void display();
void remove();
int  menu();
```

```
void modify();
void list();
void empty();
int concheck(char b[]);
};

//          ====================>          End          of          Class
Criminal<============================

/*****************************************************************
            Function for Checking Weither Input is Correct
*****************************************************************/
int criminal::ncheck (char arr[])
{
int i=0;
while (arr[i]!='\0')
{
if (((arr[i]>64)&&(arr[i]<92))||((arr[i]>95)&&(arr[i]<124))
||(arr[i]==' '))
{

i++;
continue;
}
else
{
cout<<"\nINCORRECT INPUT. TRY AGAIN !!\n ";
return 0;
}
}
return 1;
}


/*****************************************************************
*           Function to Check whether Sex is Given Correctly                *
*****************************************************************/
int criminal::s_check(char a[])
{
clrscr();
if(( (a[0]=='m')||(a[0]=='f')||(a[0]=='M')||(a[0]=='F'))&&(a[1]=='\0') )
{
return(1);
```

```
}
else
{
cout<<"\n INCORRECT INPUT ,TRY AGAIN ";
return(0);
}

}


/*****************************************************************
*           Function for getting Convicts Record                 *
*****************************************************************/
void criminal::input(void)
{
clrscr();
empty();
criminal x;
int c = 0 ;
fstream filin;
filin.open("jail.dat",ios::app|ios::binary);
while(!c)
{
cout<<"\n ENTER THE CONVICT CODE : ";
gets(conno);
c=x.concheck(conno);
}
c=0;
while(!c)
{
cout<<"\n ENTER NAME OF CONVICT : ";
gets(name);
c= ncheck(name);
}
c=0;
while(!c)
{
cout<<"\n ENTER SEX : ";
gets(sex);
c =s_check(sex);
}
cout<<"\n ENTER ADDRESS : ";
gets(address);
```

```
cout<<"\n ENTER AGE : ";
gets(age);
clrscr();
cout<<"\n ENTER DESCRIPTION : \n\n\t ";
c=0;
while(!c)
{
cout<<"\nENTER COLOUR: ";
gets(des.colour);
c=ncheck(des.colour);
}
c=0;
while(!c)
{
cout<<"\n FACE DESCRIPTION:  ";
gets(des.face);
c=ncheck(des.face);
}
cout<<"\n ENTER HEIGHT(in cm's) : ";
gets(des.height);
cout<<"\nENTER IDENTIFICATION MARKS(if no distinguishing
marks,enter NONE):";
gets(des.imark);
clrscr();
cout<<"ENTER  : \n\n\t\t ";
c=0;
while(!c)
{
cout<<"COURT (from which the prisonor convicted) : ";
gets(COURT.cour);
c= ncheck(COURT.cour);
}
c=0;
while(!c)
{
cout<<"\n\t\t STATE : ";
gets(COURT.state);
c= ncheck(COURT.state);
}
c=0;
while(!c)
{
cout<<"\n\t\t CRIME : ";
```

```
gets(COURT.crime);
c= ncheck(COURT.crime);
}
cout<<"\n\t\t ACT (under which convicted) : ";
gets(COURT.act);
clrscr();

cout<<"\n\t\t ENTER COMPLETE ADDRESS OF POLICE STATION : ";
gets(POLICE.station);
cout<<"\n\t\t ENTER FIR NO. (under which convict was arressted): ";
gets(POLICE.fir_no);
cout<<"\n\t\t ENTER DATE OF ARREST(dd/mm/yy) :";
c=0;
while(!c)
{
cout<<"\n\t\t\t\t\t day : ";
cin>>POLICE.d;
if((POLICE.d>31)||(POLICE.d<1))
{
cout<<"\n\n THIS DATE DOES NOT EXIST,TRY AGAIN!!  ";
c=0;
}
else
c=1;
}
c=0;
while(!c)
{
cout<<"\n\t\t\t\t\t month : ";
cin>>POLICE.m;
if((POLICE.m>12)||(POLICE.m<1))
{
cout<<"\n\n THIS MONTH DOES NOT EXIST,TRY AGAIN!!  ";
c=0;continue;
}
else
c=1;
if((POLICE.d==31)      &&((POLICE.m==2)      ||      (POLICE.m==4)      ||(
POLICE.m==6) ||( POLICE.m==9) || (POLICE.m==11)))
{
cout<<"\n THIS MONTH DOES NOT HAVE 31 DAYS \n TRY AGAIN !!
";
c=0;
```

```
}
else
c=1;
}
c=0;
while(!c)
{
cout<<"\n\t\t\t\t\t year(in 4 digits) : ";
cin>>POLICE.y;
if((POLICE.y <= 1930)||(POLICE.y>2020) )
{
cout<<"\n INCORRECT INPUT \n TRY AGAIN!!\n" ;
c=0;
}
else
c=1;
}
char ch;
do
{
clrscr();
cout<<"\n DO YOU WANT TO SAVE THIS INFORMATION(enter y or n)
";
cin>>ch;
if(ch=='y'||ch=='Y')
{
cout<<"\n\n\n\t Saving your Record";
for(int p=0;p<5;p++)
{
cout<<" "<<(char)219;
cout<<"\a";
delay(100);
}
filin.write((char*)this,sizeof(criminal));
}
else
{
if((ch!='n')&&(ch!='N'))
{
cout<<"\n ENTER CORRECTLY \n TRY AGAIN !! ";
}
}
} while((ch!='y')&&(ch!='Y')&&(ch!='n')&&(ch!='N'));
```

```
filin.close();

}//End of input function


/*************************************************************
*              Function for showing Convict's Record                *
*************************************************************/
void criminal::output()
{
clrscr();
textcolor(GREEN);
textbackground(LIGHTGRAY);

cout<<"@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@PERSO
NAL                RECORD                OF                CONVICT
NO@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@";
cout<<"\n\nCONVICT NO          :\t";
puts(conno);
cout<<"\nNAME               :\t";
puts(name);
cout<<"\nSEX              :\t"<<sex;
cout<<"\n\nADDRESS            :\t";
puts(address);
cout<<"\nAGE              :\t";
puts(age);
cout<<"\n\nDESCRIPTION\n";
cout<<"\n1. COLOUR           :\t";
puts(des.colour);
cout<<"\n2. FACE            :\t";
puts(des.face);
cout<<"\n3. IDENTIFICATION MARK :\t";
puts(des.imark);
cout<<"\n4. HEIGHT           :\t";
puts(des.height);
cout<<"\n\n\n\n\t\t\t\t\tpress a key to continue....";
getche();
clrscr();
cout<<"@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
CRIMINAL                                                RECORD
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@";
cout<<"\n\nCOURT INFORMATION\n\n ";
```

```
cout<<"\n1. COURT FROM WHICH CONVICTED :\t";
puts(COURT.cour);
cout<<"\n2. STATE                   :\t";
puts(COURT.state);
cout<<"\n3. CRIME                   :\t";
puts(COURT.crime);
cout<<"\n4. ACT UNDER WHICH CONVICTED  :\t";
puts(COURT.act);
cout<<"\n\n\n\n\n\n\t\t\t\t\t press a key to continue....";
getche();
clrscr();
cout<<"@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
CRIMINAL                                                  RECORD
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@";
cout<<"\n\n\nPOLICE INFORMATION \n\n";
cout<<"\n1. FIR NO.                        :\t";
puts(POLICE.fir_no);
cout<<"\n2. DATE OF ARREST                  :\t"<<POLICE.d<<"/"
<<POLICE.m<<"/"<<POLICE.y;
cout<<"\n\n3. POLICE STATION(where fir was lodged) :\t";
puts(POLICE.station);
cout<<"\n\n\n\n\n\t\t\t\t\t press a key to continue....";
getche();
clrscr();
}//end of output function


/**************************************************************
*           Function for Deleting The Record OF A Convict                *
**************************************************************/

void criminal::delete_rec(char con_no[])
{
fstream filin;
filin.open("jail.dat",ios::in|ios::ate);
int q=filin.tellg();
int c=q/sizeof(criminal);
fstream temp;
temp.open("temp.dat",ios::out);
filin.seekg(0,ios::beg);
for(int i=0;i<c;i++)
{
```

```
filin.read((char*)this,sizeof(criminal));
if(strcmp(con_no,conno)!=0)
temp.write((char*)this,sizeof(criminal));
}
filin.close();
temp.close();
filin.open("jail.dat",ios::out);
temp.open("temp.dat",ios::in|ios::ate);
int a=temp.tellg();
int size=a/sizeof(criminal);
temp.seekg(0,ios::beg);
for(int k=0;k<size;k++)
{
temp.read((char*)this,sizeof(criminal));
filin.write((char*)this,sizeof(criminal));
}
filin.close();
temp.close();
cout<<"\n\n\n\n\t\tDeleting Your Record";
for(int p=0;p<5;p++)
{
cout<<" "<<(char)219;
cout<<"\a";
delay(100);
}
}
//---------------------------------------------------------------------------
int criminal::dcheck(char n[])
{
clrscr();
fstream file;
file.open("jail.dat",ios::in|ios::ate|ios::binary);
int x=0;
int count=0,c;
int q=file.tellg();
c=q/sizeof(criminal);
file.close();
file.open("jail.dat",ios::in|ios::binary);
for(int i=0;i<c;i++)
{
file.read((char*)this,sizeof(criminal));
count++;
if(strcmp(n,conno)==0)
```

```
{
x++;
break;
}
}
if(x==0)
{
cout<<"\n CONVICT NOT FOUND!! ";
return 0;
}
else
{
return count;
}

}
//---------------------------------------------------------------------------
void criminal::display()
{
clrscr();
char N[10];
int rec;
int loc;
 textcolor(BLUE);
  textbackground(LIGHTGRAY);
cout<<"\n ENTER THE CONVICT CODE OF CONVICT ";
cout<<" WHOSE INFORMATION YOU WANT  :";
gets(N);
fstream file;
rec= dcheck(N);
file.open("jail.dat",ios::in|ios::binary);
if(rec!=0)
{
loc=(rec-1)*sizeof(criminal);
file.seekg(loc);
file.read((char*)this,sizeof(criminal));
output();
}

file.close();

}
```

```
//--------------------------------------------------------------------------------------
-----------------------------
void criminal::remove()
{
clrscr();
char no[10];int s;
cout<<"\n ENTER THE CONVICT NO. OF THE CONVICT WHOSE
RECORD YOU WISH TO ";
cout<<"DELETE :" ;
gets(no);
s= dcheck(no);
if(s!=0)
delete_rec(no);
}
//--------------------------------------------------------------------------------------
-------------------------------
int criminal::menu()
{ int ch;
do
{
clrscr();
cout<<"\n
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ MAIN
MENU
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@@@";
cout<<"\n\t\t\tÉÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍ»";
cout<<"\n\t\t\tº1. TO ADD RECORD/RECORDS   º";
cout<<"\n\t\t\tº2. TO MODIFY RECORD/RECORDSº";
cout<<"\n\t\t\tº3. TO DISPLAY RECORD       º";
cout<<"\n\t\t\tº4. TO DELETE RECORD        º";
cout<<"\n\t\t\tº5. TO LIST                 º";
cout<<"\n\t\t\tº6. TO QUIT                 º";
cout<<"\n\t\t\tÈÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍ¼";
cout<<"\nEnter the choice: ";
cin>>ch;
}
while((ch!=1)&&(ch!=2)&&(ch!=3)&&(ch!=4)&&(ch!=5)&&(ch!=6));
return(ch);
}
//--------------------------------------------------------------------------------------
void criminal::modify()
{
```

```
clrscr();
fstream file ;
char npdr[10],ncrime[10];
char nact[10],nstate[10],ncourt[10],nstat[10],nfir[10];
char str[10],nprd[15],cno[10];
int check,loc;
criminal x;
dat D;
cout<<"\n MODIFICATION FUNCTION   ";
cout<<"\nENTER THE CONVICT NUMBER OF THE CONVICT ,TO BE
MODIFIED: ";
gets(str);
check=dcheck(str);
if(check!=0)
{
file.open("jail.dat",ios::out|ios::ate);
cout<<"\n ENTER THE MODIFIED COURT RECORD:  ";
cout<<"\n ACT  :";
gets(nact);
strcpy(COURT.act,nact);
cout<<"\nSTATE : ";
gets(nstate);
strcpy(COURT.state,nstate);
cout<<"\nCOURT : ";
gets(ncourt);
strcpy(COURT.cour,ncourt);
cout<<"\n ENTER THE  MODIFIED POLICE RECORD  ";
cout<<"\n STATION : ";
gets(nstat);
strcpy(POLICE.station,nstat);
cout<<"\nFIR NO   : ";
gets(nfir);
strcpy(POLICE.fir_no,nfir);
cout<<"\n DATE OF ARREST(dd,mm,yyyy)  ";
int c=0;
while(!c)
{
cout<<"\n\t\t\t\t\t day : ";
cin>>D.d;
if((D.d>31)||(D.d<1))
{
cout<<"\n\n THIS DATE DOES NOT EXIST,TRY AGAIN!!  ";
c=0;
```

```
}
else
c=1;
}
c=0;
while(!c)
{
cout<<"\n\t\t\t\t\t month : ";
cin>>D.m;
if((D.m>12)||(D.m<1))
{
cout<<"\n\n THIS MONTH DOES NOT EXIST,TRY AGAIN!!  ";
c=0;
continue;
}
else
c=1;
if((D.d==31)  &&((D.m==2)  ||  (D.m==4)  ||(  D.m==6)  ||(  D.m==9)  ||
(D.m==11)))
{
cout<<"\n THIS MONTH DOES NOT HAVE 31 DAYS \n TRY AGAIN !!
";
c=0;
}
else
c=1;
}
c=0;
while(!c)
{
cout<<"\n\t\t\t\t\t year(in 4 digits) : ";
cin>>D.y;
if((D.y <= 1930)||(D.y>2050) )
{
cout<<"\n INCORRECT INPUT \n TRY AGAIN!!\n" ;
c=0;
}
else
c=1;
}
POLICE.d=D.d;
POLICE.m=D.m;
POLICE.y=D.y;
```

```
loc=(check-1)*sizeof(criminal);
file.seekp(loc);
file.write((char*)this,sizeof(criminal));
file.close();
clrscr();
cout<<"Modifying and Saving Your Record";
for(int p=0;p<4;p++)
{
delay(300);
cout<<" "<<(char)219;
cout<<"\a";
}
}
}
//-------------------------------------------------------------------------------------
---
void criminal::list()
{
clrscr();
int j ;
fstream file;
file.open("jail.dat",ios::in|ios::ate|ios::binary);
int q=file.tellg();
int c=q/sizeof(criminal);
file.seekg(0);
cout<<"                        "<<endl;
cout<<"                                  "<<endl;
cout<<"                           CONVICT  LIST    "<<endl;
cout<<"                                   "<<endl;
cout<<"                        "<<endl;
cout<<"\n=============================================
================================"<<endl;
cout<<"SNO\tNAME\t\t\tCONVICT NO.\t\tCRIME"<<endl;
cout<<"=============================================
================================"<<endl;
int i=0;
while(file.read((char*)this,sizeof(criminal)))
{
 i++;
cout<<i<<"       ";
for( j=0;j<strlen(name);j++)
cout<<name[j];
cout<<"\t\t    ";
```

```
for(j=0;j<strlen(conno);j++)
cout<<conno[j];
cout<<"\t       ";
for(j=0;j<strlen(COURT.crime);j++)
cout<<COURT.crime[j];
cout<<"\n--------------------------------------------------------------------------------
\n";
if((i%3)==0&&(i!=c))
{
cout<<"Press a key to continue.....";
getch();
clrscr();
cout<<"\n=============================================
================================"<<endl;
cout<<"SNO\tNAME\t\t\tCONVICT NO.\t\t\tCRIME"<<endl;
cout<<"=============================================
================================"<<endl;
}
}
file.close();
}
//------------------------------------------------------------------------------------
---------
void criminal::empty()
{
int i;
for(i=0;i<20;i++)
name[i]=' ';
for(i=0;i<10;i++)
conno[i]=' ';
for(i=0;i<3;i++)
sex[i]=' ';
for(i=0;i<4;i++)
age[i]=' ';
for(i=0;i<50;i++)
address[i]=' ';
for(i=0;i<20;i++)
COURT.cour[i]=' ';
for(i=0;i<20;i++)
COURT.crime[i]=' ';
for(i=0;i<20;i++)
COURT.act[i]=' ';
for(i=0;i<20;i++)
```

```
COURT.state[i]=' ';
for(i=0;i<20;i++)
des.colour[i]=' ';
for(i=0;i<20;i++)
des.imark[i]=' ';
for(i=0;i<20;i++)
des.face[i]=' ';
for(i=0;i<6;i++)
des.height[i]=' ';
for(i=0;i<50;i++)
POLICE.station[i]=' ';
for(i=0;i<10;i++)
POLICE.fir_no[i]=' ';
POLICE.m = 0;
POLICE.d = 0;
POLICE.y = 0;
}
//-------------------------------------------------------------------------------------------
---------------
int criminal::concheck(char b[])
{
fstream  file;char a[10];
int check=0;
file.open("jail.dat",ios::in|ios::ate);
int q=file.tellg();
int size=q/sizeof(criminal);
file.seekg(0,ios::beg);
for(int i=0;i<size;i++)
{
file.read((char*)this,sizeof(criminal));
strcpy(a,conno);
if(strcmp(b,a)==0)
{
check+=1;
break;
}
}
if(check==0)
{return(10);
}
else
{
cout<<"\n GIVEN CONVICT CODE ALREADY EXISTS!! "<<endl;
```

```
return(0);
}
}
void main()
{
clrscr();
textbackground(LIGHTGRAY);
textcolor(RED);
int p,q=800;
char ph[]="PROCESSING...PLEASE WAIT ";
gotoxy(20,10);
cout<<ph;
gotoxy(44,10);
for(p=0;p<15;p++)
{
delay(q);
cout<<(char)178;
cout<<"\a";
q-=10;
}
clrscr();
textbackground(WHITE);
textcolor(RED);
int i;
gotoxy(1,1);
for(i=1;i<=80;i++)//for upper border
{if(i%2==0)
{gotoxy(i,1);
cprintf("Ü");
delay(25);
}
else
{gotoxy(i,1);
cprintf("ß");
delay(25);
}
}

gotoxy(80,2);
for(i=2;i<=49;i++)//for right border
{if(i%2==0)
{gotoxy(80,i);
cprintf("Þ");
```

```
delay(25);
}
else
{gotoxy(80,i);
cprintf("Ý");
delay(25);
}
}
gotoxy(79,49);
for(i=79;i>=1;i--)//for lower border
{if(i%2==0)
{gotoxy(i,49);
cprintf("Ü");
delay(25);
}
else
{gotoxy(i,49);
cprintf("ß");
delay(25);
}
}
gotoxy(1,49);
for(i=49;i>=2;i--)//for left border
{if(i%2==0)
{gotoxy(1,i);
cprintf("Þ");
delay(25);

}
else
{gotoxy(1,i);
cprintf("Ý");
delay(25);


}
}

gotoxy(29,10);
char ch[]="WELCOME TO C.S.C PROJECT";
for(i=0;i<strlen(ch);i++)
{ textcolor(RED);
delay(100);
```

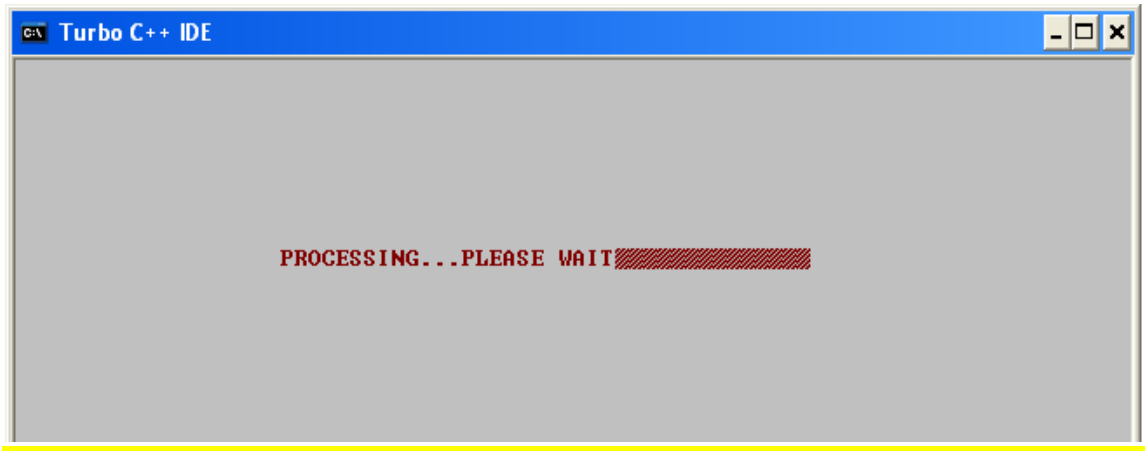
```
cout<<ch[i];
}
gotoxy(28,11);
cout<<"ÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍ";
char ch1[]="DONE BY:K.Jerome & K.Simpson";
gotoxy(27,13);
for(i=0;i<strlen(ch1);i++)
{
delay(100);
cout<<ch1[i];
}
cout<<endl;
char ch2[]="TOPIC:CRIMINAL RECORDS MANAGEMENT";
gotoxy(25,17);
for(i=0;i<strlen(ch2);i++)
{delay(100);
cout<<ch2[i];
}
textcolor(MAGENTA);
gotoxy(29,10);
delay(500);
cprintf("WELCOME TO C.S.C PROJECT");
gotoxy(27,13);
cprintf("DONE BY:K.Jerome & K.Simpson");
gotoxy(25,17);
cprintf("TOPIC:CRIMINAL RECORDS MANAGEMENT");
delay(4000);
clrscr();
int x;
for( i=0;i<3;i++)
{
clrscr();

  textcolor(WHITE);
  textbackground(LIGHTBLUE);
cprintf("############################  &&&&&&&&&&&&&&&&&&
##################################\r\n");
cprintf("############################     &CRIMINAL     RECORD&
##################################\r\n");
cprintf("############################  &&&&&&&&&&&&&&&&&&
##################################\r\n");
cout<<"\n\t\t\t\t.....Done by:K.Jerome ";
cout<<"\t\n\nENTER PASSWORD..";
```

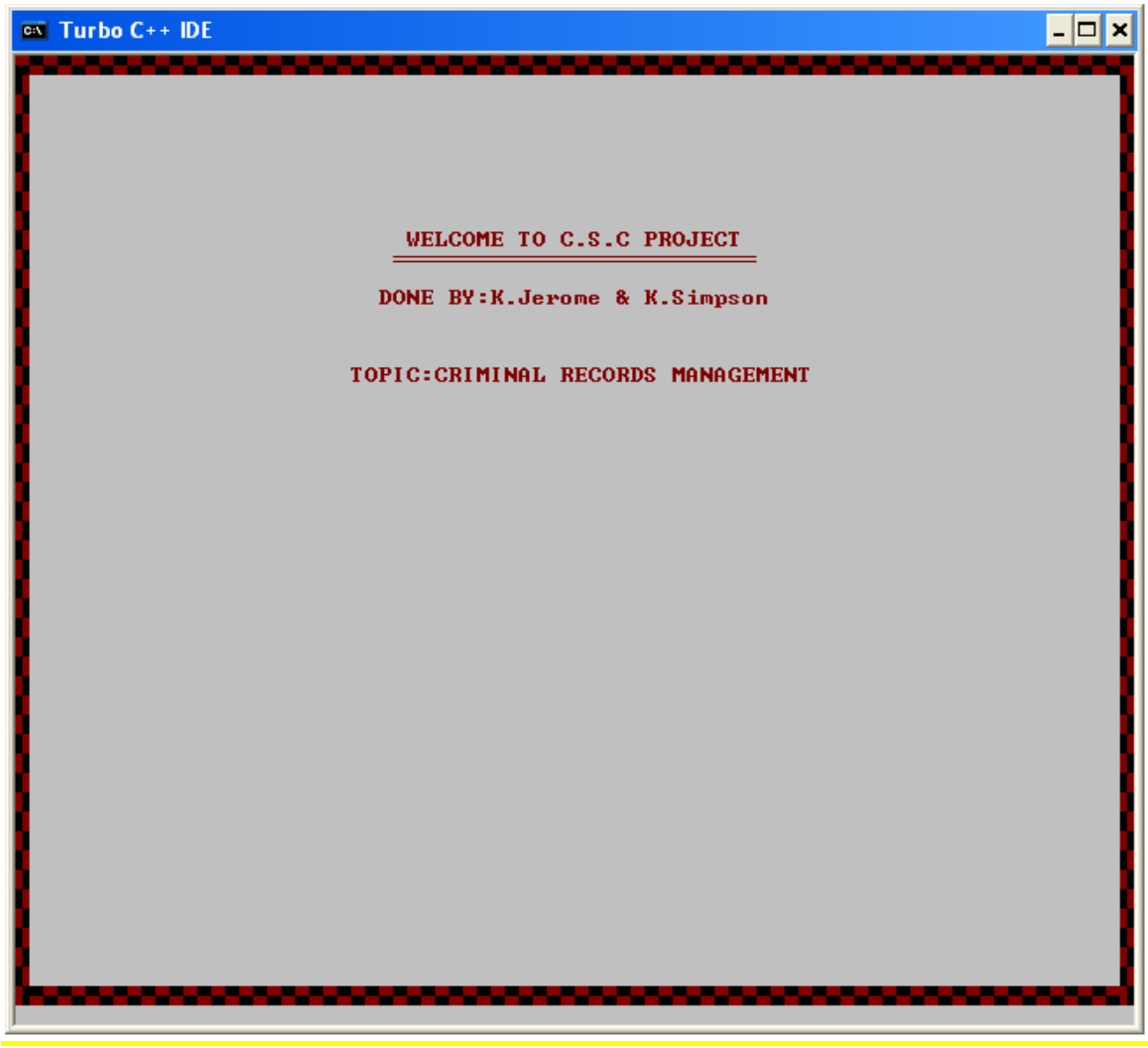
```
 textcolor(WHITE);
 textbackground(LIGHTRED);
char ch[20];
gotoxy(10,15);
cout<<endl;
cprintf("      |                                  | ");
cout<<endl;
cprintf("      |          ==================          | ");
cout<<endl;
cprintf("      |          |||||||||           | ");
cout<<endl;
cprintf("      |          ||||||||<|          | ");
cout<<endl;
cprintf("      |       ()  |||||||||  ()       | ");
cout<<endl;
cprintf("      |       ::  |||||||||  ::       | ");
cout<<endl;
cprintf("      |       ::  |||||||||  ::       | ");
cout<<endl;
cprintf("      |      [~~] |||||||||| [~~]      | ");
cout<<endl;
cprintf("      |       ++  |||||||||  ++       | ");
cout<<endl;
cprintf("      |          +++++++++++++++++|          | ");
cout<<endl;
cprintf("                 |{}    JAIL     <|          | ");
cout<<endl;
cprintf("      |          +++++++++++++++++|          | ");
cout<<endl;
cprintf("      |          |||||||||           | ");
cout<<endl;
cprintf("      |          |||||||||           | ");
cout<<endl;
cprintf("      |          |||||||||           | ");
cout<<endl;
cprintf("      |          |||||||||           | ");
cout<<endl;
cprintf("      |          |||||||||           | ");
cout<<endl;
cprintf("      |          ||||||||<|          | ");
cout<<endl;
cprintf("      | ___________|_|_|_|_|_|_|_|_|_|_____________| ");
cout<<endl;
```

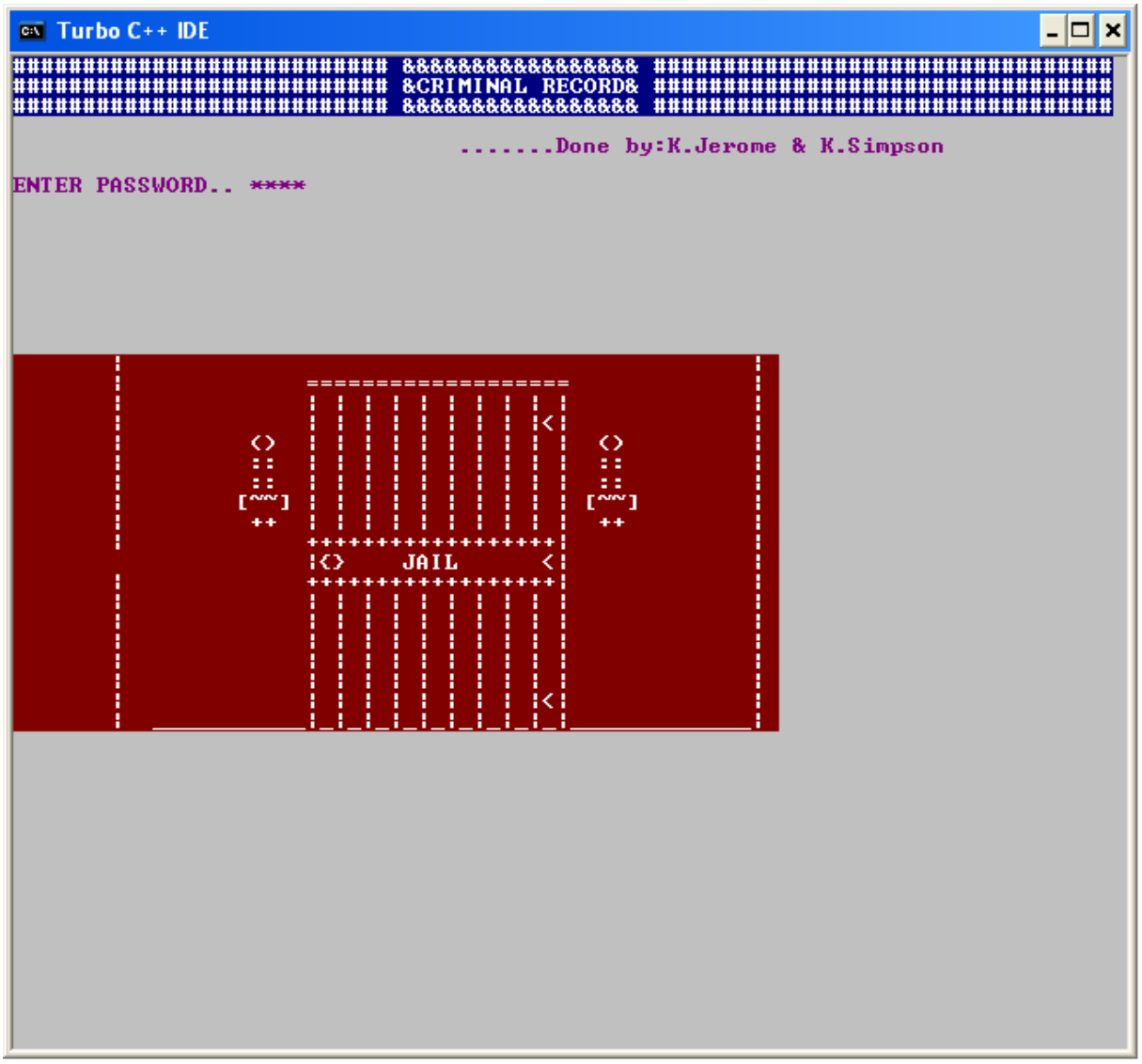
```
 textcolor(RED);
 textbackground(LIGHTGRAY);

int j=0;
int a;
a=0;
while(1)
{ch[j]=getch();
if(ch[j]==13)
break;
gotoxy(a+18,7);
cout<<'*';
j++;
a++;
}
ch[j]='\0';
x=strcmpi(ch,"jsjs");
if(x==0)
{
cout<<'*';
break;
}
else
{
cout<<"\a";
}
}

if(x==0)
{
int choice,ans='y';
criminal l;
while((ans=='y'|| ans=='Y'))
{
choice= l.menu();
switch(choice)
{
case 1:
l.input();
break;
case 2 :
l.modify();
break;
```

```
case 3:
l.display();
break;
case 4:
l.remove() ;
break;
case 5:
l.list();
break;
case 6:
exit(0);

}
cout<<"\n\n\nDO YOU WANT TO CONTINUE(press y to continue)..";
ans=getch();
if(ans=='y'||ans=='Y')
continue;
}
}
}
```

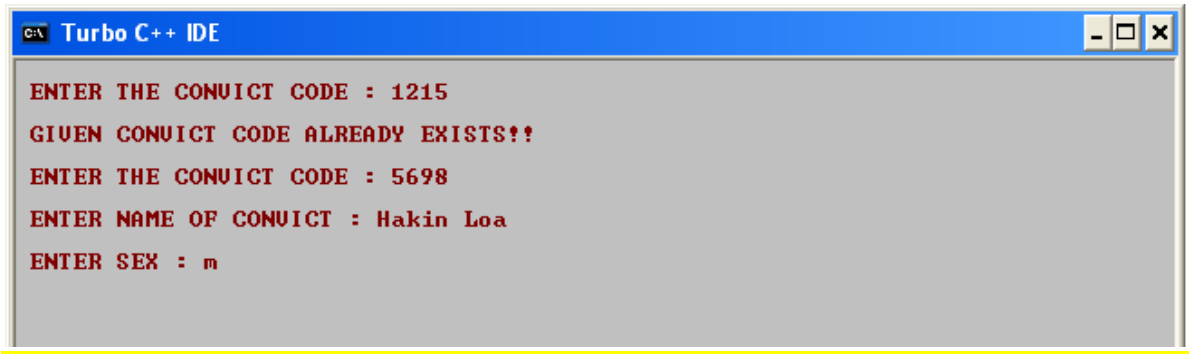
OUTPUT

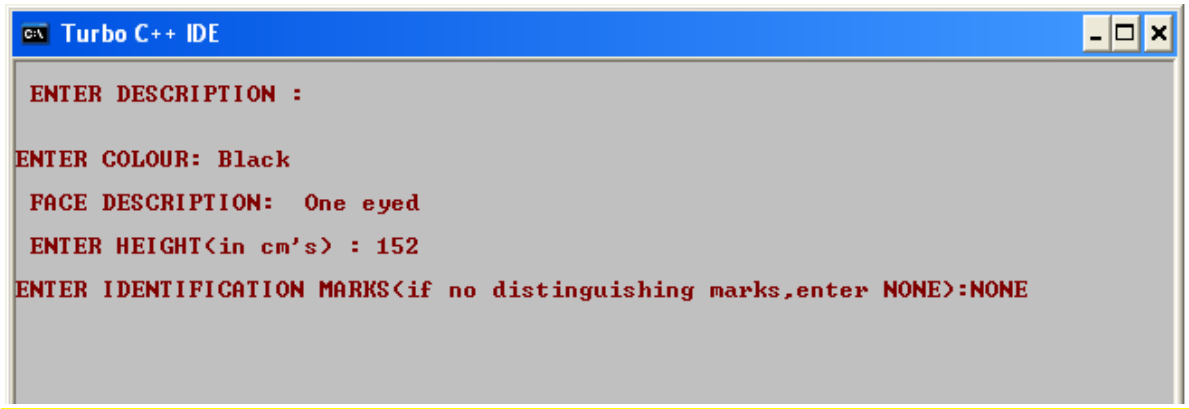

Turbo C++ IDE
PROCESSING...PLEASE WAIT

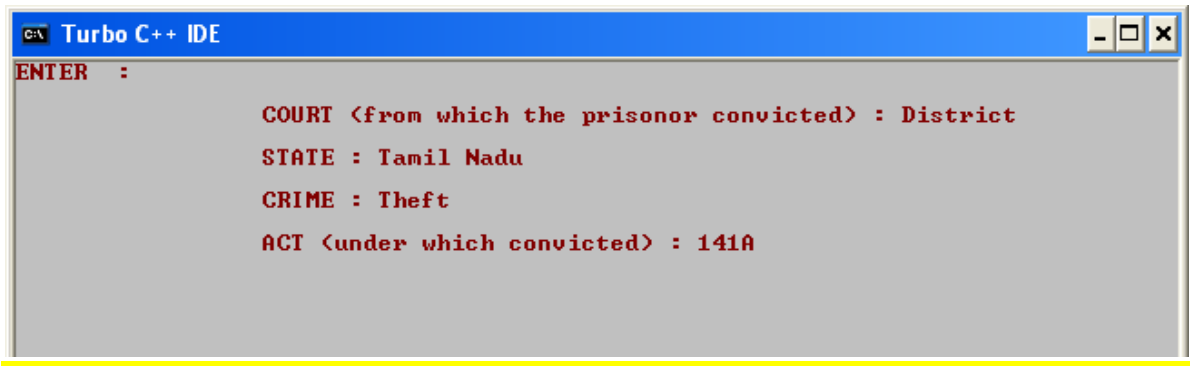

Turbo C++ IDE
WELCOME TO C.S.C PROJECT
DONE BY:K.Jerome & K.Simpson
TOPIC:CRIMINAL RECORDS MANAGEMENT

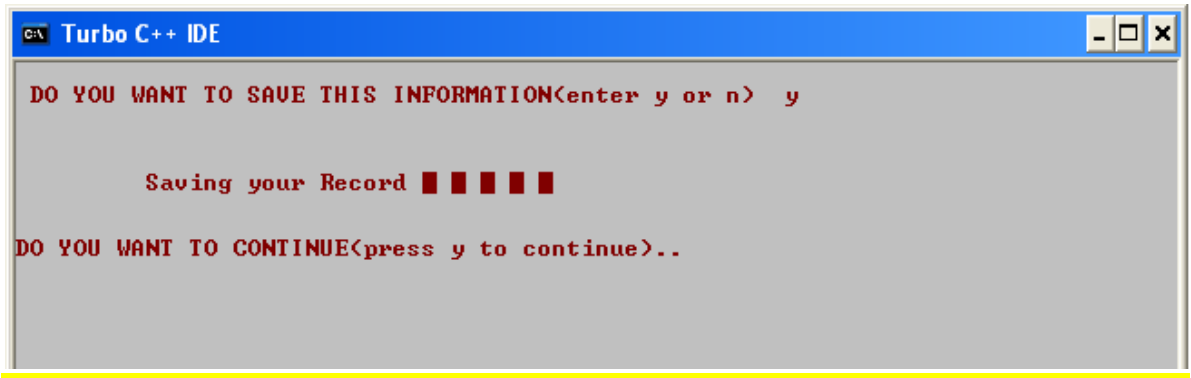

Turb C++ IDE
&CRIMINAL RECORD&
.....Done by:K.Jerome & K.Simpson
ENTER PASSWORD.. ****
JAIL


Turbo C++ IDE
MAIN MENU
1. TO ADD RECORD/RECORDS
2. TO MODIFY RECORD/RECORDS
3. TO DISPLAY RECORD
4. TO DELETE RECORD
5. TO LIST
6. TO QUIT
Enter the choice: 1


Turbo C++ IDE
ENTER THE CONVICT CODE : 1215
GIVEN CONVICT CODE ALREADY EXISTS!!
ENTER THE CONVICT CODE : 5698
ENTER NAME OF CONVICT : Hakin Loa
ENTER SEX : m


Turbo C++ IDE
ENTER DESCRIPTION :
ENTER COLOUR: Black
FACE DESCRIPTION: One eyed
ENTER HEIGHT(in cm's) : 152
ENTER IDENTIFICATION MARKS(if no distinguishing marks,enter NONE):NONE


Turbo C++ IDE
ENTER :
COURT (from which the prisonor convicted) : District
STATE : Tamil Nadu
CRIME : Theft
ACT (under which convicted) : 141A

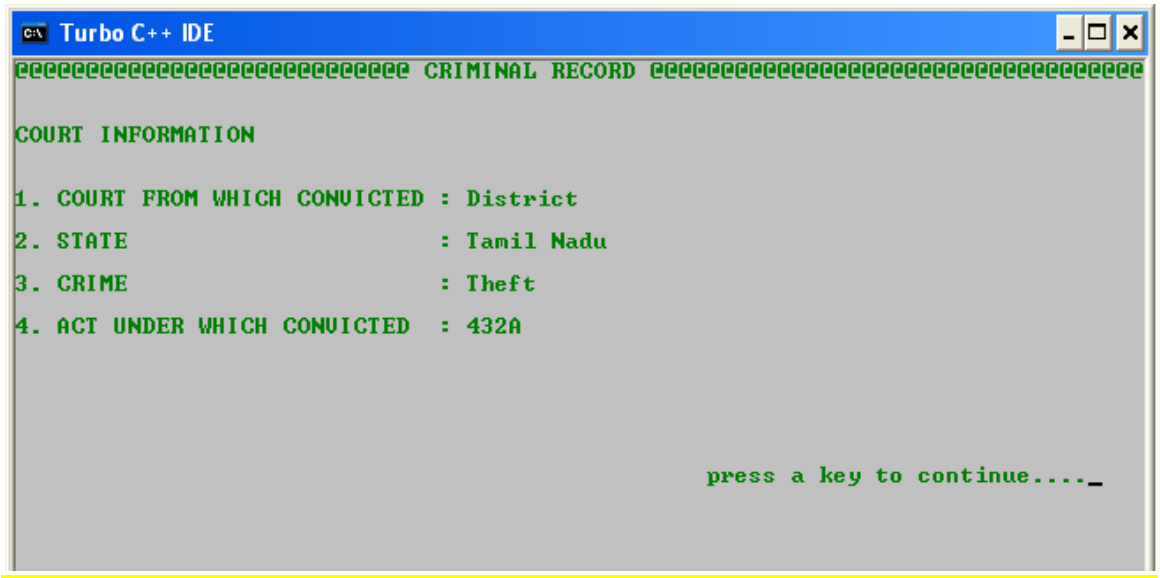

Turbo C++ IDE
DO YOU WANT TO SAVE THIS INFORMATION(enter y or n) y
Saving your Record
DO YOU WANT TO CONTINUE(press y to continue)..

Turbo C++ IDE

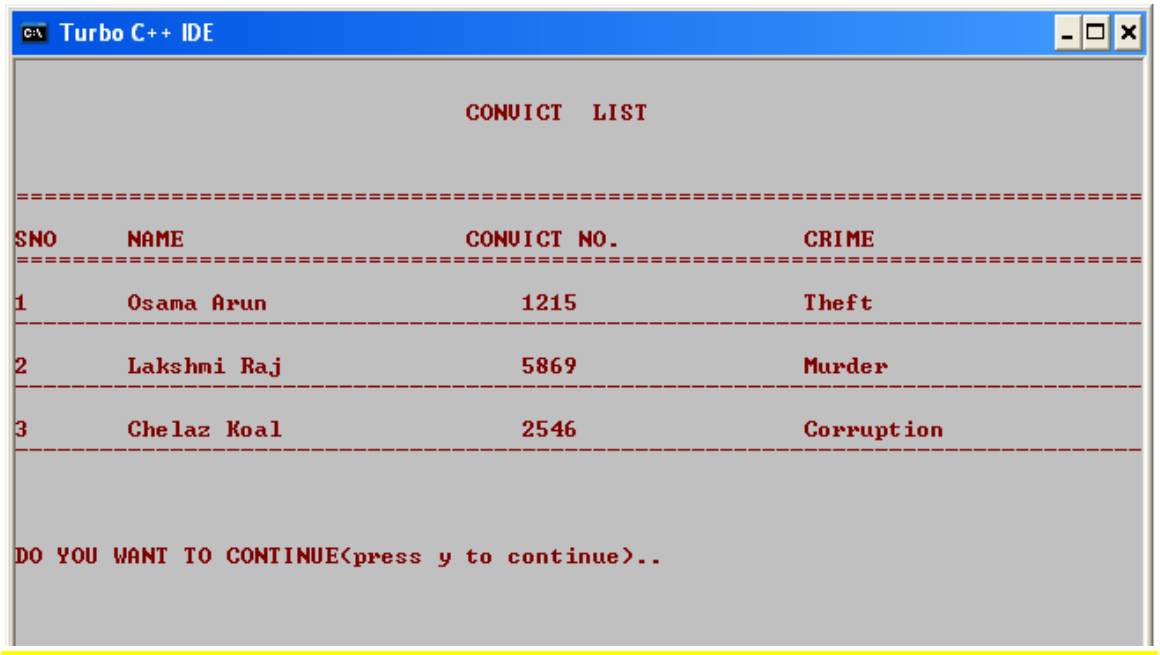
```
 ENTER THE MODIFIED COURT RECORD:
 ACT  :565

STATE : Kerala

COURT : High Court

 ENTER THE  MODIFIED POLICE RECORD
 STATION : K.K.Nagar

FIR NO    : 652

 DATE OF ARREST(dd,mm,yyyy)
                                        day : 31

                                        month : 2

 THIS MONTH DOES NOT HAVE 31 DAYS
 TRY AGAIN !!
                                        month : 8

                                        year(in 4 digits) : 2015_
```

Turbo C++ IDE

```
@@@@@@@@@@@@@@@@@@@@@@@@@@PERSONAL RECORD OF CONVICT NO@@@@@@@@@@@@@@@@@@@@@@@@@@

CONVICT NO               :        1215

NAME                     :        Osama Arun

SEX                      :        m

ADDRESS                  :        No:99/76, North East road, T.Nagar, Chennai-04

AGE                      :        37

DESCRIPTION

1. COLOUR                :        Black

2. FACE                  :        One eyed

3. IDENTIFICATION MARK   :        NONE

4. HEIGHT                :        152


                                        press a key to continue....
```


Turbo C++ IDE
CRIMINAL RECORD
COURT INFORMATION
1. COURT FROM WHICH CONVICTED : District
2. STATE : Tamil Nadu
3. CRIME : Theft
4. ACT UNDER WHICH CONVICTED : 432A
press a key to continue....


Turbo C++ IDE
CRIMINAL RECORD
POLICE INFORMATION
1. FIR NO. : 134
2. DATE OF ARREST : 5/5/2009
3. POLICE STATION(where fir was lodged) : T.Nagar
press a key to continue....


Turbo C++ IDE
CONVICT LIST
SNO NAME CONVICT NO. CRIME
1 Osama Arun 1215 Theft
2 Lakshmi Raj 5869 Murder
3 Chelaz Koal 2546 Corruption
DO YOU WANT TO CONTINUE(press y to continue)..

CONCLUSION

There is still scope of improvement in our project. It can be improvised to better standards by providing user login which can also help employers to know about the history of their labourers, and in politics as it can be used by common man to know about the electoral candidate. This way of maintaining records of the criminals in electronic form not only reduces maintenance but also is time saving.

While entering the criminal's crime, the application should automatically give the ACT (under which convicted).Only limited number of criminal records could be added, which can be improved. The criminal records should be easily accessible by each and every police station throughout the country. If the criminal is a foreigner, additional details like passport details, country origin must be included in the database.

BIBLIOGRAPHY

- http://www.tutorialspoint.com/cplusplus/cpp_object_oriented.htm
- http://www.studytonight.com/cpp/cpp-and-oops-concepts.php
- http://people.howstuffworks.com/criminal-record.htm
- http://www.programmingsimplified.com
- http://stdcxx.apache.org/

BOOKS USED

COMPUTER SCIENCE WITH C++ BY SUMITA ARORA