#include<fstream.h>

#include<stdlib.h>

#include<stdio.h>

#include<conio.h>

#include<graphics.h>

#include<dos.h>

#include<string.h>

#include<iomanip.h>

ofstream r;

ifstream t;

void credits();

void menu1();

void menu2();

void menu3();

void terms();

class xii

{

public:

int roll;

char na[20];

int m1,m2,m3,m4,m5;

char g[10];

int t;

int p;

void input()

{

gotoxy(1,5);

cout<<"\tENTER ROLL NUMBER OF THE STUDENT :";

cin>>roll;

cout<<"\tENTER NAME OF THE STUDENT :";

gets(na);

cout<<"\n\n\tENTER MARKS OF THE 5 SUBJECTS OUT OF 100";

cout<<"\n";

cout<<"\tENTER MARKS IN ENGLISH:";

cin>>m1;

cout<<"\n";

cout<<"\tENTER MARKS IN MATHS:";

cin>>m2;

cout<<"\n";

cout<<"\tENTER MARKS IN PHYSICS:";

cin>>m3;

cout<<"\n";

cout<<"\tENTER MARKS IN CHEMISTRY:";

cin>>m4;

cout<<"\n";

cout<<"\tENTER MARKS IN COMPUTER:";

cin>>m5;

cout<<"\n";

cleardevice();

t=m1+m2+m3+m4+m5;

p=t/5;

gotoxy(1,1);

}

void output()

{

cout<<"\n";

cout<<" "<<roll<<"\t"<<" "<<na<54<setw(10)<<"\t";

cout<<" "<<m1<<","<<m2<<","<<m3<<","<<m4<<","<<m5<<"\t";

cout<<" "<<p<<"%"<<"\t"<<"\t"<<" "<<g<<"\n";

cout<<"\t";

}

}s[30];

int i,n;

void writes()

{

clrscr();

cleardevice();

cout<<"\n";

cout<<"\tENTER THE NUMBER OF RECORDS TO BE CREATED :";

cin>>n;

cleardevice();

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

{

s[i].input();

if(s[i].p>=95)

strcpy(s[i].g,"A1");

if(s[i].p>=90&&s[i].p<95)

strcpy(s[i].g,"A2");

if(s[i].p>=80&&s[i].p<90)

strcpy(s[i].g,"B1");

if(s[i].p>=70&&s[i].p<80)

strcpy(s[i].g,"B2");

if(s[i].p>=60&&s[i].p<70)

strcpy(s[i].g,"C1");

if(s[i].p>=50&&s[i].p<60)

strcpy(s[i].g,"C2");

if(s[i].p>=40&&s[i].p<50)

strcpy(s[i].g,"D1");

if(s[i].p<40)

strcpy(s[i].g,"D2");

r.write((char\*)&s[i],sizeof(s[i]));

}

}

void append()

{

clrscr();

cleardevice();

cout<<"\n";

int k;

cout<<"\t\t\t\*-----------\*"<<"\n";

cout<<"\t\t\t| APPENDING |"<<"\n";

cout<<"\t\t\t\*-----------\*"<<"\n"<<"\n";

cout<<"\tENTER THE NUMBER OF RECORDS TO BE APPENDED :";

cin>>k;

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

{

s[i].input();

if(s[i].p>=95)

strcpy(s[i].g,"A1");

if(s[i].p>=90&&s[i].p<95)

strcpy(s[i].g,"A2");

if(s[i].p>=80&&s[i].p<90)

strcpy(s[i].g,"B1");

if(s[i].p>=70&&s[i].p<80)

strcpy(s[i].g,"B2");

if(s[i].p>=60&&s[i].p<70)

strcpy(s[i].g,"C1");

if(s[i].p>=50&&s[i].p<60)

strcpy(s[i].g,"C2");

if(s[i].p>=40&&s[i].p<50)

strcpy(s[i].g,"D1");

if(s[i].p<40)

strcpy(s[i].g,"D2");

r.write((char\*)&s[i],sizeof(s[i]));

}

getch();

n+=k;

}

void search()

{

clrscr();

cleardevice();

char name[20];

int flag=0;

cout<<"\n";

cout<<"\t\t\t\*-----------\*"<<"\n";

cout<<"\t\t\t| SEARCHING |"<<"\n";

cout<<"\t\t\t\*-----------\*"<<"\n"<<"\n";

step:

cout<<"\n\tENTER THE NAME TO BE SEARCHED :";

gets(name);

cout<<"\n";

cout<<"ROLL NAME MARKS PERCENTAGE GRADE"<<"\n";

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

{

t.read((char \*) &s[i],sizeof(s[i]));

if(strcmp(name,s[i].na)==0)

{

flag=1;

s[i].output();

break;

}

}

if(flag==0)

{

cleardevice();

cout<<"\tSORRY\n";

cout<<"\tTHE NAME DOES NOT EXIST.\n";

}

getch();

}

void update()

{

clrscr();

cleardevice();

cout<<"\t\t\t\*--------------\*"<<"\n";

cout<<"\t\t\t| UPDATED LIST |"<<"\n";

cout<<"\t\t\t\*--------------\*"<<"\n"<<"\n";

cout<<"ROLL NAME MARKS PERCENTAGE GRADE"<<"\n";

cout<<"-----------------------------------------------------------"<<"\n";

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

{

if(s[i].p<=40.0)

s[i].t=s[i].t+(s[i].t/10);

s[i].p=s[i].t/5.0;

if(s[i].p>=50.0&&s[i].p<60.0)

strcpy(s[i].g,"C2");

if(s[i].p>=40.0&&s[i].p<50.0)

strcpy(s[i].g,"D1");

if(s[i].p<40.0)

strcpy(s[i].g,"Failed");

r.write((char\*)&s[i],sizeof(s[i]));

s[i].output();

}

getch();

}

void sort()

{

clrscr();

cleardevice();

xii tp;

cout<<"\t\t\t\*-------------\*"<<"\n";

cout<<"\t\t\t| SORTED LIST |"<<"\n";

cout<<"\t\t\t\*-------------\*"<<"\n"<<"\n";

cout<<"ROLL NAME MARKS PERCENTAGE GRADE"<<"\n";

cout<<"---------------------------------------------------------"<<"\n";

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

{

t.read((char \*) &s[i],sizeof(s[i]));

for(i=0;i<n-1;i++)

{ for(int j=i+1;j<n;j++)

{ if(strcmp(s[i].na,s[j].na)>0)

{ tp=s[i];

s[i]=s[j];

s[j]=tp;

}

}

}

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

s[i].output();

}

getch();

}

void failed()

{

cleardevice();

cout<<"\t\*\*----------------------------------------------------------\*\*"<<"\n";

cout<<"\t|| THE LIST OF STUDENTS WHO FAILED IN MORE THAN ONE SUBJECT ||"<<"\n";

cout<<"\t\*\*----------------------------------------------------------\*\*\n\n";

cout<<"ROLL NAME MARKS PERCENTAGE GRADE"<<"\n";

cout<<"-------------------------------------------------------"<<"\n";

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

{

int c=0;

t.read((char \*) &s[i],sizeof(s[i]));

if (s[i].m1<40)

c++;

if (s[i].m2<40)

c++;

if (s[i].m3<40)

c++;

if (s[i].m4<40)

c++;

if (s[i].m5<40)

c++;

if(c>=2)

{

s[i].output();

}

}

getch();

}

void reads()

{

clrscr();

cleardevice();

cout<<"\n";

cout<<"\t\*\*------------------------------------------------\*\*"<<"\n";

cout<<"\t|| THE REPORT CARD OF STUDENTS OF CLASS XII-B2 ||"<<"\n";

cout<<"\t\*\*------------------------------------------------\*\*"<<"\n"<<"\n";

cout<<"ROLL NAME MARKS PERCENTAGE GRADE"<<"\n";

cout<<"--------------------------------------------------------"<<"\n";

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

{

t.read((char \*) &s[i],sizeof(s[i]));

s[i].output();

}

getch();

}

void terms()

{

int gdriver = DETECT, gmode, errorcode;

initgraph(&gdriver,&gmode,"c:bgi");

errorcode = graphresult();

if (errorcode !=grOk) /\* an error code occurred \*/

{

printf("Graphics error: %s\n", grapherrormsg(errorcode));

printf("Press any key to halt:");

getch;

exit(1); /\* terminate with an error code \*/

}

char chc;

rectangle(1,1,getmaxx(),getmaxy());

settextstyle(GOTHIC\_FONT,VERT\_DIR,5);

settextjustify(LEFT\_TEXT,CENTER\_TEXT); //Was actually- settextjustify(LEFT\_TEXT,CENTER\_TEXT,5);

outtextxy(50,240,"REPORT CARD");

settextstyle(SANS\_SERIF\_FONT,HORIZ\_DIR,5);

outtextxy(240,50,"SELECT A TERM");

outtextxy(240,90,"OF YOUR CHOICE.");

outtextxy(240,105," ......................................");

settextstyle(SANS\_SERIF\_FONT,HORIZ\_DIR,4);

setbkcolor(BLACK);

outtextxy(300,170,"1.Ist TERM");

outtextxy(300,220,"2.IInd TERM");

outtextxy(300,270,"3.IIIrd TERM");

outtextxy(300,320,"4.EXIT");

settextstyle(SANS\_SERIF\_FONT,HORIZ\_DIR,3);

outtextxy(400,400,"ENTER CHOICE");

chc=getch();

switch(chc)

{

case '1':cleardevice();

menu1();

break;

case '2':cleardevice();

menu2();

break;

case '3':cleardevice();

menu3();

break;

case '4':cleardevice();

credits();

exit(0);

default :clrscr();

cleardevice();

settextstyle(TRIPLEX\_FONT,HORIZ\_DIR,5);

outtextxy(220,210,"WRONG CHOICE !");

outtextxy(170,270,"PLEASE TRY AGAIN.");

sleep(2);

cleardevice();

terms();

}

closegraph();

}

void credits()

{

cleardevice();

setbkcolor(4);

gotoxy(33,6);

cout<<"-:Created By:-";

for(int i=70;i>28;i--)

{

gotoxy(i,8);

cout<<"Navi Arora";

delay(70);

}

for(i=70;i>28;i--)

{

gotoxy(i,10);

cout<<"www.NaviArora.com";

delay(70);

}

gotoxy(28,25);

cout<<"Press any key to exit...";

getch();

exit(0);

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*menu1\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void menu1()

{

cleardevice();

char ch;

back:

settextstyle(GOTHIC\_FONT,VERT\_DIR,5);

settextjustify(LEFT\_TEXT,CENTER\_TEXT);

outtextxy(50,240,"REPORT CARD");

settextstyle(SANS\_SERIF\_FONT,VERT\_DIR,4);

setbkcolor(BLACK);

settextjustify(LEFT\_TEXT,CENTER\_TEXT);

outtextxy(110,240,"Ist TERM");

settextstyle(SANS\_SERIF\_FONT,HORIZ\_DIR,4);

outtextxy(290,50,"1.MAKE REPORT CARD");

outtextxy(290,90,"2.APPEND NAMES");

outtextxy(290,130,"3.SORT NAMES");

outtextxy(290,170,"4.SEARCH A NAME");

outtextxy(290,210,"5.UPDATE GRADE");

outtextxy(290,250,"6.VIEW REPORT CARD");

outtextxy(290,290,"7.LIST OF FAILURES");

outtextxy(290,330,"8.MAIN MENU");

settextstyle(SANS\_SERIF\_FONT,HORIZ\_DIR,3);

outtextxy(400,420,"ENTER CHOICE");

ch=getch();

switch(ch)

{

case '1':

r.open("report\_1");

writes();

r.close();

cleardevice();

goto back;

case '2':

r.open("report\_1",ios::app|ios::out);

append();

r.close();

cleardevice();

goto back;

case '3':

t.open("report\_1");

sort();

t.close();

cleardevice();

goto back;

case '4':

t.open("report\_1");

search();

t.close();

cleardevice();

goto back;

case '5':

r.open("report\_1",ios::end);

update();

r.close();

cleardevice();

goto back;

case '6':

t.open("report\_1");

reads();

t.close();

cleardevice();

goto back;

case '7':

t.open("report\_1");

failed();

t.close();

cleardevice();

goto back;

case '8':

closegraph();

terms();

default :clrscr();

cleardevice();

settextstyle(TRIPLEX\_FONT,HORIZ\_DIR,5);

outtextxy(220,240,"WRONG CHOICE !");

outtextxy(190,280,"PLEASE TRY AGAIN.");

sleep(2);

cleardevice();

goto back;

}

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*menu2\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void menu2()

{

cleardevice();

char ch;

back:

settextstyle(GOTHIC\_FONT,VERT\_DIR,5);

settextjustify(LEFT\_TEXT,CENTER\_TEXT);

outtextxy(50,240,"REPORT CARD");

settextstyle(SANS\_SERIF\_FONT,VERT\_DIR,4);

setbkcolor(BLACK);

settextjustify(LEFT\_TEXT,CENTER\_TEXT);

outtextxy(110,240,"IInd TERM");

settextstyle(SANS\_SERIF\_FONT,HORIZ\_DIR,4);

outtextxy(290,50,"1.MAKE REPORT CARD");

outtextxy(290,90,"2.APPEND NAMES");

outtextxy(290,130,"3.SORT NAMES");

outtextxy(290,170,"4.SEARCH A NAME");

outtextxy(290,210,"5.UPDATE GRADE");

outtextxy(290,250,"6.VIEW REPORT CARD");

outtextxy(290,290,"7.LIST OF FAILURES");

outtextxy(290,330,"8.MAIN MENU");

settextstyle(SANS\_SERIF\_FONT,HORIZ\_DIR,3);

outtextxy(400,420,"ENTER CHOICE");

ch=getch();

switch(ch)

{

case '1':

r.open("report\_2");

writes();

r.close();

cleardevice();

goto back;

case '2':

r.open("report\_2",ios::app|ios::out);

append();

r.close();

cleardevice();

goto back;

case '3':

t.open("report\_2");

sort();

t.close();

cleardevice();

goto back;

case '4':

t.open("report\_2");

search();

t.close();

cleardevice();

goto back;

case '5':

r.open("report\_2",ios::end);

update();

r.close();

cleardevice();

goto back;

case '6':

t.open("report\_2");

reads();

t.close();

cleardevice();

goto back;

case '7':

t.open("report\_2");

failed();

t.close();

cleardevice();

goto back;

case '8':

closegraph();

terms();

default :clrscr();

cleardevice();

settextstyle(TRIPLEX\_FONT,HORIZ\_DIR,5);

outtextxy(220,240,"WRONG CHOICE !");

outtextxy(190,280,"PLEASE TRY AGAIN.");

sleep(2);

cleardevice();

goto back;

}

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*menu3\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void menu3()

{

cleardevice();

char ch;

back:

settextstyle(GOTHIC\_FONT,VERT\_DIR,5);

settextjustify(LEFT\_TEXT,CENTER\_TEXT);

outtextxy(50,240,"REPORT CARD");

settextstyle(SANS\_SERIF\_FONT,VERT\_DIR,4);

setbkcolor(BLACK);

settextjustify(LEFT\_TEXT,CENTER\_TEXT);

outtextxy(110,240,"IIIrd TERM");

settextstyle(SANS\_SERIF\_FONT,0,4);

outtextxy(290,50,"1.MAKE REPORT CARD");

outtextxy(290,90,"2.APPEND NAMES");

outtextxy(290,130,"3.SORT NAMES");

outtextxy(290,170,"4.SEARCH A NAME");

outtextxy(290,210,"5.UPDATE GRADE");

outtextxy(290,250,"6.VIEW REPORT CARD");

outtextxy(290,290,"7.LIST OF FAILURES");

outtextxy(290,330,"8.MAIN MENU");

settextstyle(SANS\_SERIF\_FONT,0,3);

outtextxy(400,420,"ENTER CHOICE");

ch=getch();

switch(ch)

{

case '1':

r.open("report\_3");

writes();

r.close();

cleardevice();

goto back;

case '2':

r.open("report\_3",ios::app|ios::out);

append();

r.close();

cleardevice();

goto back;

case '3':

t.open("report\_3");

sort();

t.close();

cleardevice();

goto back;

case '4':

t.open("report\_3");

search();

t.close();

cleardevice();

goto back;

case '5':

r.open("report\_3",ios::end);

update();

r.close();

cleardevice();

goto back;

case '6':

t.open("report\_3");

reads();

t.close();

cleardevice();

goto back;

case '7':

t.open("report\_3");

failed();

t.close();

cleardevice();

goto back;

case '8':

closegraph();

terms();

default :clrscr();

cleardevice();

settextstyle(TRIPLEX\_FONT,HORIZ\_DIR,5);

outtextxy(220,240,"WRONG CHOICE !");

outtextxy(190,280,"PLEASE TRY AGAIN.");

sleep(2);

cleardevice();

goto back;

}

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*main\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void main()

{

int gdriver = DETECT, gmode, errorcode;

initgraph(&gdriver,&gmode,"c:\\turboc3\\bgi");

errorcode = graphresult();

if (errorcode != grOk) /\*an error occurred\*/

{

printf("Graphics error: %s\n", grapherrormsg(errorcode));

printf("Press any key to halt:");

getch;

exit(1); /\* terminate with an error code \*/

}

settextstyle(SANS\_SERIF\_FONT,HORIZ\_DIR,5);

rectangle(1,1,getmaxx(),getmaxy());

setbkcolor(9);

outtextxy(300,100,"A");

delay(2000);

outtextxy(200,170,"PROJECT ON");

delay(2000);

outtextxy(95,240,"REPORT CARD MAKING");

delay(2000);

for(int i=0;i<360;i++)

{

ellipse(320,240,0,i,300,200);

delay(12);

}

closegraph();

terms();

}