// Ashish Waghmode

// roll no : 2174

#include <algorithm>

#include <iostream>

#include <vector>

#include <string>

using namespace std;

class student

{

public:

int rollno;

string name;

char dob[15];

bool operator== (const student & student1)

{

return (rollno == student1.rollno);

}

bool operator< (const student & student1)

{

return (rollno < student1.rollno);

}

friend ostream & operator << (ostream & out, const student &);

friend istream & operator >> (istream & in, student & k);

};

ostream & operator << (ostream & out, const student & k)

{

out << "\n\t\t" << k.rollno << "\t\t" << k.name << "\t\t" << k.dob;

return out;

}

istream & operator >> (istream & in, student & k)

{

cout << "\nEnter Roll No : ";

in >> k.rollno;

cout << "\nEnter Name : ";

in >> k.name;

cout << "\nEnter DOB : ";

in >> k.dob;

return in;

}

/\* Function Defination to Accept Record \*/

vector < student > read ()

{

int n;

student k;

vector < student > st;

cout << "\nEnter total no. of students : ";

cin >> n;

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

{

cin >> k;

st.push\_back (k);

}

return st;

}

void

printfunction (const student & k)

{

cout << k;

}

void

print (const vector < student > &st)

{

cout << "\n\t\tROLL NO\t\tNAME\t\tDATE OF BIRTH";

for\_each (st.begin (), st.end (), printfunction);

}

/\* Function Defination For Inserting New Record \*/

void

insert (vector < student > &st)

{

student k;

cin >> k;

st.push\_back (k);

}

//delete record

void delet(vector<student>&st)

{

student k;

cout<<"\nEnter Student Roll No To Delete : ";

cin>>k.rollno;

vector<student>::iterator p;

p=find(st.begin(),st.end(),k);

if(p!=st.end())

st.erase(p);

else

cout<<"\nNot found ";

}

//search record

void search( vector<student>&st)

{

student k;

cout<<"\nEnter Student Roll No To Search : ";

cin>>k.rollno;

cout<<"\n\n\t\tROLL NO\t\tNAME\t\tDATE OF BIRTH";

vector<student>::iterator p;

p=find(st.begin(),st.end(),k);

if(p!=st.end())

cout<<\*p;

else

cout<<"\nNot found ";

}

//sort records

bool myfunc(const student &k, const student &i2)

{

return(k.rollno<i2.rollno);

}

void sort( vector<student> &st)

{

sort(st.begin(),st.end(),myfunc);

}

int

main ()

{

vector < student > st;

int op;

do

{

cout << "\n\n\t\t-------<< MENU >>---------";

cout << "\n\t\t1.Create ";

cout << "\n\t\t2.Display ";

cout << "\n\t\t3.Insert ";

cout << "\n\t\t4.Delete ";

cout << "\n\t\t5.Search ";

cout << "\n\t\t6.Sort";

cout << "\n\t\t7.Quit";

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

cout << "\n\t\tEnter your choice : ";

cin >> op;

switch (op)

{

case 1:

st = read ();

break;

case 2:

print (st);

break;

case 3:

insert (st);

break;

case 4:

delet(st);

break;

case 5:

search(st);

break;

case 6:

sort(st);

print(st);

break;

}

}

while (op != 7);

}

/\* -------<< MENU >>---------

1.Create

2.Display

3.Insert

4.Delete

5.Search

6.Sort

7.Quit

--------------------------

Enter your choice : 1

Enter total no. of students : 2

Enter Roll No : 2110

Enter Name : AAAAA

Enter DOB : 01/01/2000

Enter Roll No : 2122

Enter Name : BBBB

-------<< MENU >>---------

1.Create

2.Display

3.Insert

4.Delete

5.Search

6.Sort

7.Quit

--------------------------

Enter your choice : 2

ROLL NO NAME DATE OF BIRTH

2110 AAAAA 01/01/2000

2122 BBBB 02/01/2001

-------<< MENU >>---------

1.Create

2.Display

3.Insert

4.Delete

5.Search

6.Sort

7.Quit

--------------------------

Enter your choice : 5

Enter Student Roll No To Search : 2110

ROLL NO NAME DATE OF BIRTH

2110 AAAAA 01/01/2000

-------<< MENU >>---------

1.Create

2.Display

3.Insert

4.Delete

5.Search

6.Sort

7.Quit

--------------------------

Enter your choice : 7

\*/