

```

/*
Write C++ Program using STL for sorting and searching user defined
records such as item records using vector container.
*/
#include <iostream> //standard input output stream header file
#include <algorithm> //The STL algorithms are generic because they can
operate on a variety of data structures
#include <vector> //The header file for the STL vector library is vector.
using namespace std;
class Item // creating class Item
{
public:
char name[10];
int quantity;
int cost;
int code;
bool operator==(const Item& i1) //Boolean operators allow you to create
more complex conditional statements
{
if(code==i1.code) //operator will return 1 if the comparison is true, or
0 if the comparison is false
return 1;
return 0;
}
bool operator<(const Item& i1)
{
if(code<i1.code) //operator will return 1 if the comparison is true, or
0 if the comparison is false
return 1;
return 0;
}
};
vector<Item> o1;
void print(Item &i1);
void display();
void insert();
void search();
void dlt();
bool compare(const Item &i1, const Item &i2)
{
//if (i1.name != i2.name) return i1.cost < i2.cost;
return i1.cost < i2.cost;
}
int main()
{
int ch;
do
{
cout<<"\n* * * * * Menu * * * * *";
cout<<"\n1.Insert";
cout<<"\n2.Display";
cout<<"\n3.Search";
cout<<"\n4.Sort";
cout<<"\n5.Delete";
cout<<"\n6.Exit";
cout<<"\nEnter your choice : ";
cin>>ch;

```

```

switch(ch)
{
case 1:
insert();
break;

case 2:
display();
break;

case 3:
search();
break;

case 4:
sort(ol.begin(),ol.end(),compare);
cout<<"\n\n Sorted on Cost : ";
display();
break;

case 5:
dlt();
break;

case 6:
exit(0);
}

}while(ch!=7);
return 0;
}
void insert()
{
Item il;
cout<<"\nEnter Item Name : ";
cin>>il.name;
cout<<"\nEnter Item Quantity : ";
cin>>il.quantity;
cout<<"\nEnter Item Cost : ";
cin>>il.cost;
cout<<"\nEnter Item Code : ";
cin>>il.code;
ol.push_back(il);
}
void display()
{
for_each(ol.begin(),ol.end(),print);
}
void print(Item &il)
{
cout<<"\n";
cout<<"\nItem Name : "<<il.name;
cout<<"\nItem Quantity : "<<il.quantity;
cout<<"\nItem Cost : "<<il.cost;
cout<<"\nItem Code : "<<il.code;
cout<<"\n\n";
}
void search()

```

```

{
    vector<Item>::iterator p;
    Item i1;
    cout<<"\nEnter Item Code to search : ";
    cin>>i1.code;
    p=find(o1.begin(),o1.end(),i1);
    if(p==o1.end())
    {
        cout<<"\nNot found!!!";
    }
    else
    {
        cout<<"\nFound!!!";
    }
}

void dlt()
{
    vector<Item>::iterator p;
    Item i1;
    cout<<"\nEnter Item Code to delete : ";
    cin>>i1.code;
    p=find(o1.begin(),o1.end(),i1);
    if(p==o1.end())
    {
        cout<<"\nNot found!!!";
    }
    else
    {
        o1.erase(p);
        cout<<"\nDeleted!!!";
    }
}

```