



OOP

Name: Mahesh Jagtap

Modern College of Engineering

Shivajinagar, Pune 5. Roll no: 21027

Assignment no. 5

Title: Personal information system using sorting & searching for STL & vector container.

Problem statement:

Write a C++ program using STL for sorting and searching user defined records such as person's (name, DOB, mobile no. etc). using vector container.

Prerequisites: oop basics, STL, vector container.

Objectives: To learn the concept of STL, searching, sorting & vector container.

Theory:

The standard Template Library (STL) is a set of C++ template classes to provide common programming data structures & functions such as lists, stacks, arrays, etc.

It is a library of container classes, algorithms, and iterators.

It is a generalized library & so, its components are parameterized.

A working knowledge of template classes is a prerequisite for working with STL.



STL has four components

- Algorithms
- containers
- functions
- Iterators

① Algorithms:

- searching
- sorting
- Important STL Algorithms
- useful array algorithms
- partition operations
- Numeric

② Containers

- containers or container classes store objects of data. There are in total seven standard "first class" container classes & three container adaptor classes & only seven header files that provide access to this container or container adaptors.

* sequence containers:

implement data structures which can be accessed in a sequential manner.

- vector
- list
- deque
- arrays
- forward list (introduced in C++11)



3

* Container adaptors !
provide a different interface for sequential containers.

- queue
- priority - queue
- stack

* Associative containers :-

implement sorted data structures that can be quickly searched ($O(\log n)$ complexity).

- set
- multiset
- map
- multimap.

* unordered Associative Containers :-

implement unordered data structures that can be quickly searched.

- unordered set
- unordered_multiset
- unordered_map
- unordered_multimap

③ functions :

- The STL includes classes that overload the function call operator. Instances of such classes are called functions objects or function objects. functions allow the working of associated function to be customized with the help of parameters to be



passed.

④ Iterators:

- As name suggests, iterators are used for working upon a sequence of values. They are the major feature that allow generality in STL.

* Utility Library:

- Defined in header `<utility>`
- pair

Sorting:

It is one of the most basic functions applied to data. It means arranging the data in a particular fashion, which can be increasing or decreasing. There is a built in function in C++ STL by the name of `sort()`.

This function internally uses Intro Sort. In more details it is implemented using hybrid of quicksort, heap sort & insertion sort.

By default, it uses Quicksort but if Quicksort is doing unfair partitioning & taking more than $n \log n$ time, it switches to heap sort & when the array size becomes really small, it switches to Insertion sort.

The prototype for sort is :

sort (start address, end address)

Searching:

start address = the address of 1st element of array

end address: address of next contiguous location of the last element of the array.

So actually sort() in the range of [start ad, end address]

// sorting:

```
# include <iostream>
```

```
# include <algorithm>
```

```
using namespace std;
```

```
void show (int a[])
```

```
{ for (int i=0; i<10; i++)
```

```
    cout << a[i];
```

```
}
```

```
int main()
```

```
{ int a[10] = {1, 5, 8, 9, 6, 7, 3, 4, 2, 0};
```

```
    cout << "The array before sorting : " ;
```

```
    show(a);
```

```
    cout << sort(a, a+10);
```

```
    cout << "The array after sorting is " ;
```

```
    show(a);
```

```
}
```



6

Qp:-

The array before sorting: 1 5 8 9 6 7 3 4 2 0

The array after sorting: 0 1 2 3 4 5 6 7 8 9

Searching (binary)

It is a widely used algorithm for searching that requires array to be sorted before search is applied.

The main idea behind this algo is to keep dividing the array in half until element is found or all elements are examined.

The prototype for binary search is:
binary search (startad, endad, value to find)

where,

startad = the address of first element of array.

endad = the address of last element of array.

Value to find: the target value which we have to search for.



7

Algorithm:

1. start
2. Give a header file to use 'vector'
3. create vector 'the item record'
4. Initialize variables to store item code, item name, quantity & cost.
5. Using iterators store as many records you want to store using predefined functions push-back().
6. Using predefined function sort(), sort the data stored according to user requirements.
7. Using predefined functions, search() the element from the vector the user wants to check.
8. display ~~the~~ & call the functions using a menu.
9. end.

Input: personal info such as name, DOB, mobile no.

Output:

* menu *

1. Insert
2. Display
3. search
4. sort
5. Delete
6. Exit.

Output:

Enter your Choice : 1

Enter Item name : Bat

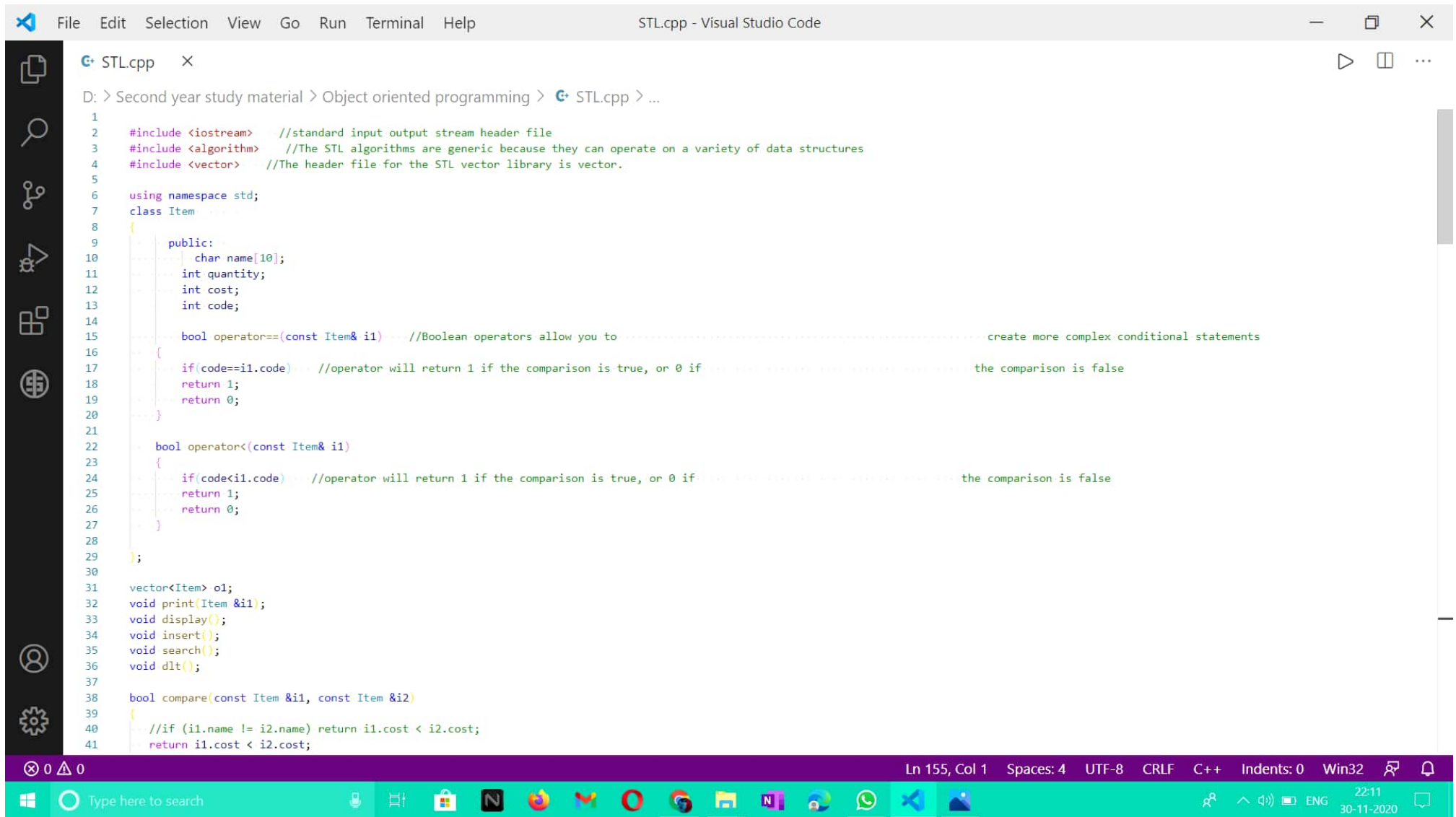
Enter Item Quantity : 2

Enter ~~Item~~ item cost : 50

Enter Item code : 1

Conclusion:-

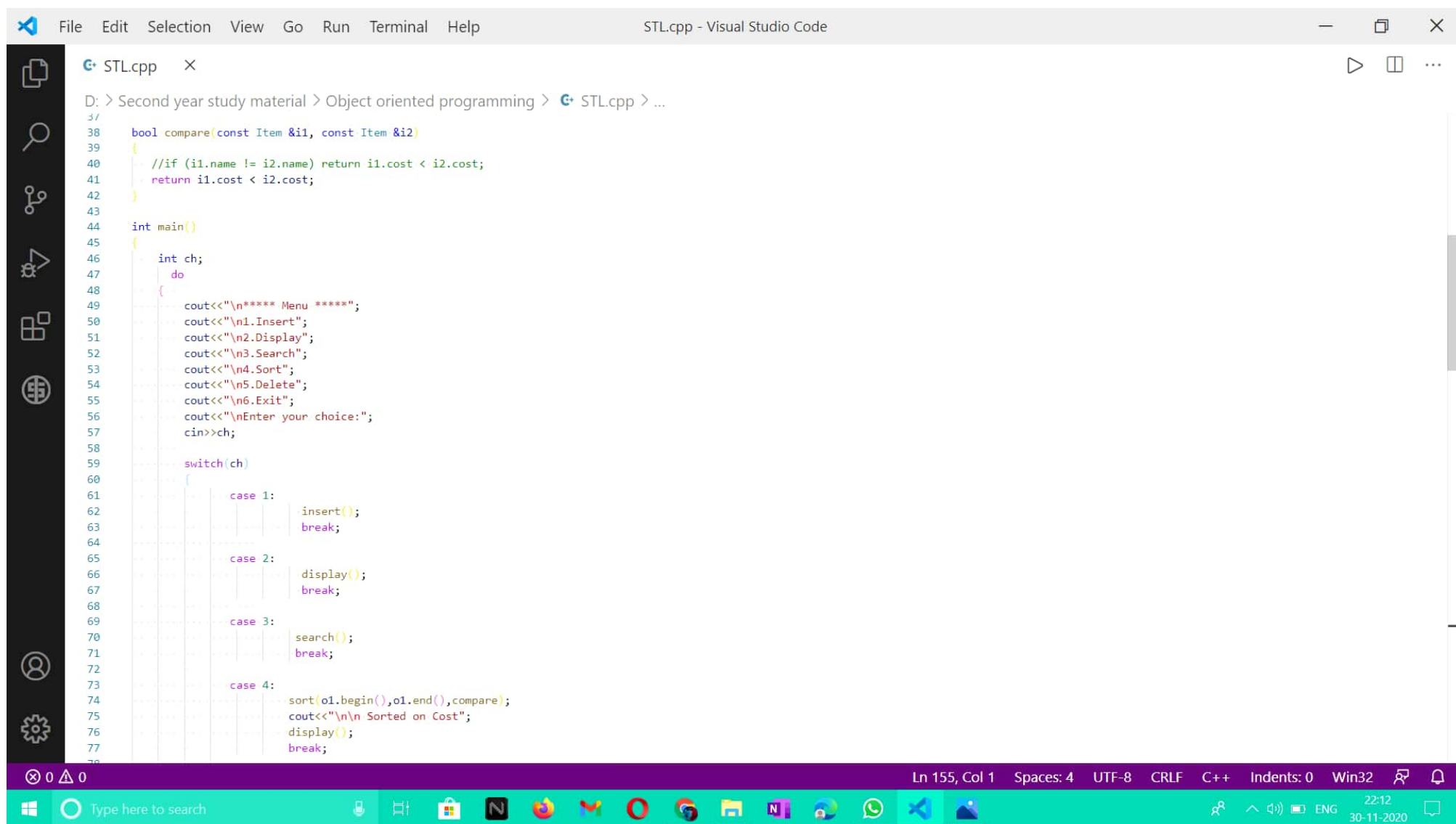
Hence, we have successfully studied the concept of STL (Standard template library) & how it makes many data structures easy. It brieks about the predefined functions of STL & their uses ~~are~~ such as a search() & sort().



The image shows a screenshot of the Visual Studio Code editor interface. The title bar at the top reads "STL.cpp - Visual Studio Code". The menu bar includes "File", "Edit", "Selection", "View", "Go", "Run", "Terminal", and "Help". The editor window displays a C++ file named "STL.cpp" with the following code:

```
1
2 #include <iostream> //standard input output stream header file
3 #include <algorithm> //The STL algorithms are generic because they can operate on a variety of data structures
4 #include <vector> //The header file for the STL vector library is vector.
5
6 using namespace std;
7 class Item
8 {
9     public:
10     char name[10];
11     int quantity;
12     int cost;
13     int code;
14
15     bool operator==(const Item& i1) //Boolean operators allow you to create more complex conditional statements
16     {
17         if(code==i1.code) //operator will return 1 if the comparison is true, or 0 if the comparison is false
18             return 1;
19         return 0;
20     }
21
22     bool operator<(const Item& i1)
23     {
24         if(code<i1.code) //operator will return 1 if the comparison is true, or 0 if the comparison is false
25             return 1;
26         return 0;
27     }
28
29 };
30
31 vector<Item> o1;
32 void print(Item &i1);
33 void display();
34 void insert();
35 void search();
36 void dlt();
37
38 bool compare(const Item &i1, const Item &i2)
39 {
40     //if (i1.name != i2.name) return i1.cost < i2.cost;
41     return i1.cost < i2.cost;
```

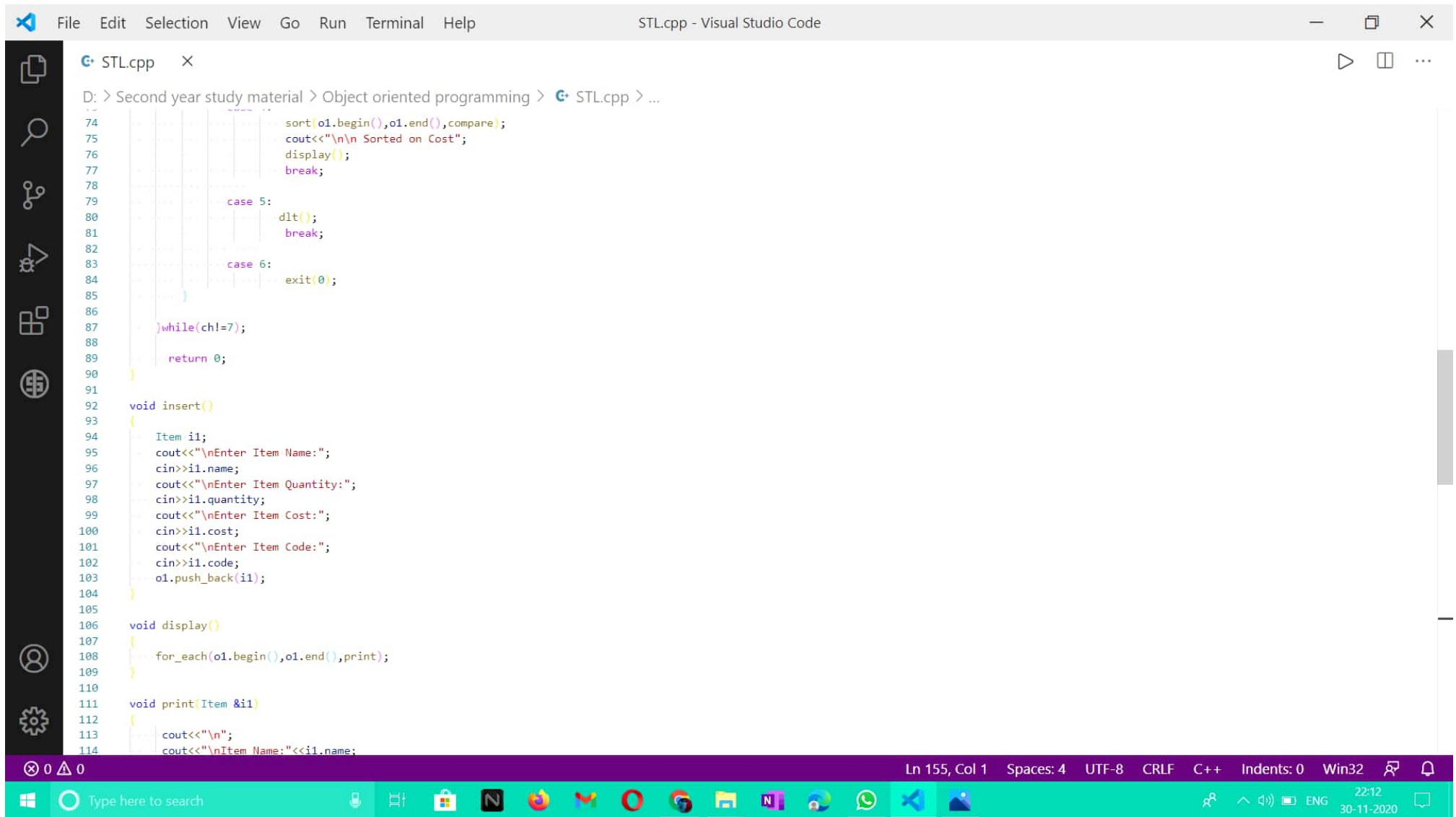
The status bar at the bottom shows "Ln 155, Col 1", "Spaces: 4", "UTF-8", "CRLF", "C++", "Indents: 0", "Win32", and the date "30-11-2020". The Windows taskbar is visible at the very bottom with various application icons.



The image shows a screenshot of the Visual Studio Code editor interface. The title bar at the top reads "STL.cpp - Visual Studio Code". The menu bar includes "File", "Edit", "Selection", "View", "Go", "Run", "Terminal", and "Help". The editor window displays a C++ file named "STL.cpp" with the following code:

```
37
38 bool compare(const Item &i1, const Item &i2)
39 {
40     //if (i1.name != i2.name) return i1.cost < i2.cost;
41     return i1.cost < i2.cost;
42 }
43
44 int main()
45 {
46     int ch;
47     do
48     {
49         cout<<"\n***** Menu *****";
50         cout<<"\n1.Insert";
51         cout<<"\n2.Display";
52         cout<<"\n3.Search";
53         cout<<"\n4.Sort";
54         cout<<"\n5.Delete";
55         cout<<"\n6.Exit";
56         cout<<"\nEnter your choice:";
57         cin>>ch;
58
59         switch(ch)
60         {
61             case 1:
62                 insert();
63                 break;
64
65             case 2:
66                 display();
67                 break;
68
69             case 3:
70                 search();
71                 break;
72
73             case 4:
74                 sort(o1.begin(),o1.end(),compare);
75                 cout<<"\n\n Sorted on Cost";
76                 display();
77                 break;
78         }
79     } while (ch != 6);
80 }
```

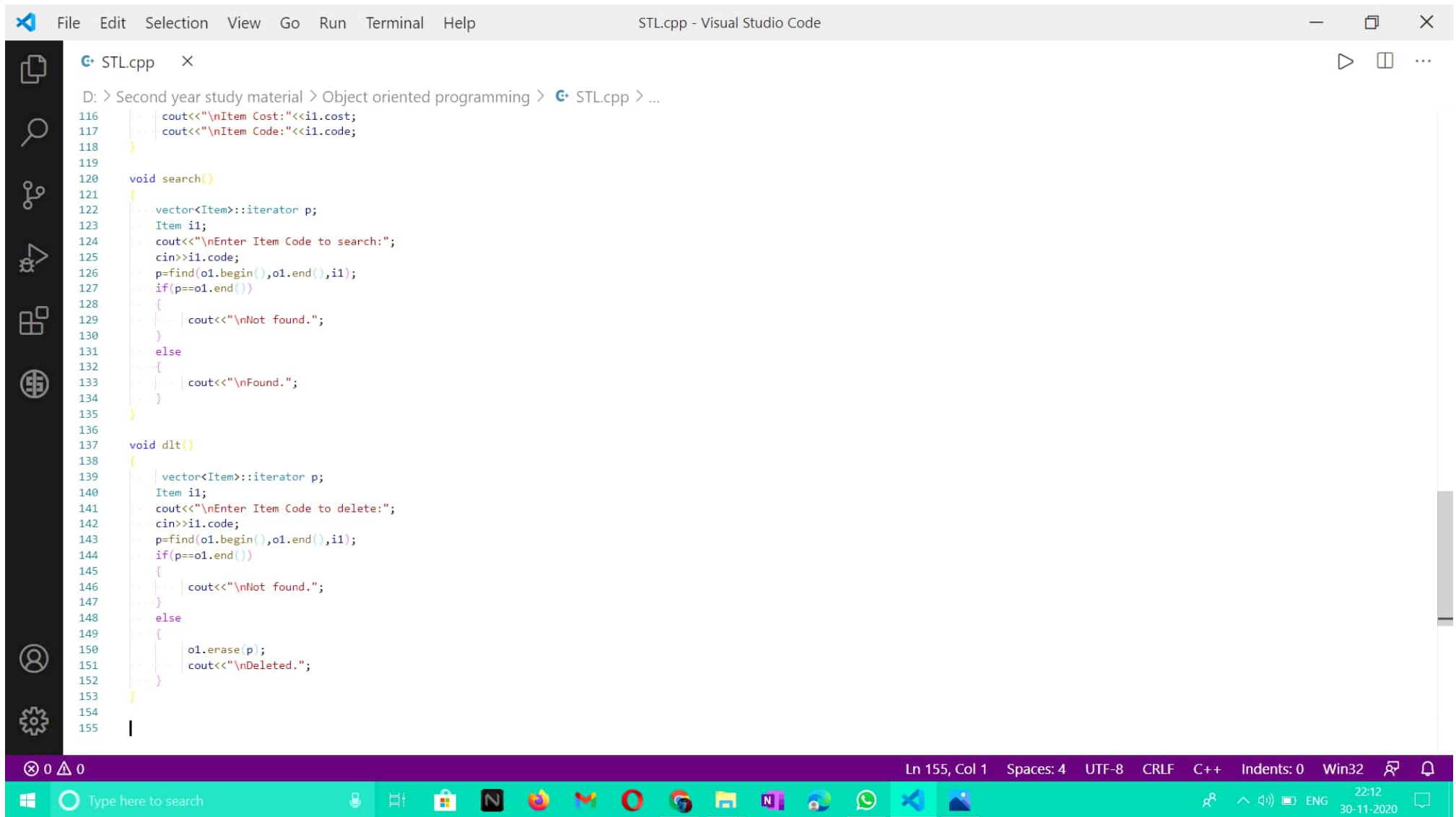
The status bar at the bottom indicates "Ln 155, Col 1", "Spaces: 4", "UTF-8", "CRLF", "C++", "Indents: 0", "Win32", and the date "22:12 30-11-2020". The Windows taskbar is visible at the very bottom with various application icons.



The image shows a screenshot of the Visual Studio Code editor interface. The title bar at the top reads "STL.cpp - Visual Studio Code". The menu bar includes "File", "Edit", "Selection", "View", "Go", "Run", "Terminal", and "Help". The editor window displays a C++ file named "STL.cpp" with the following code:

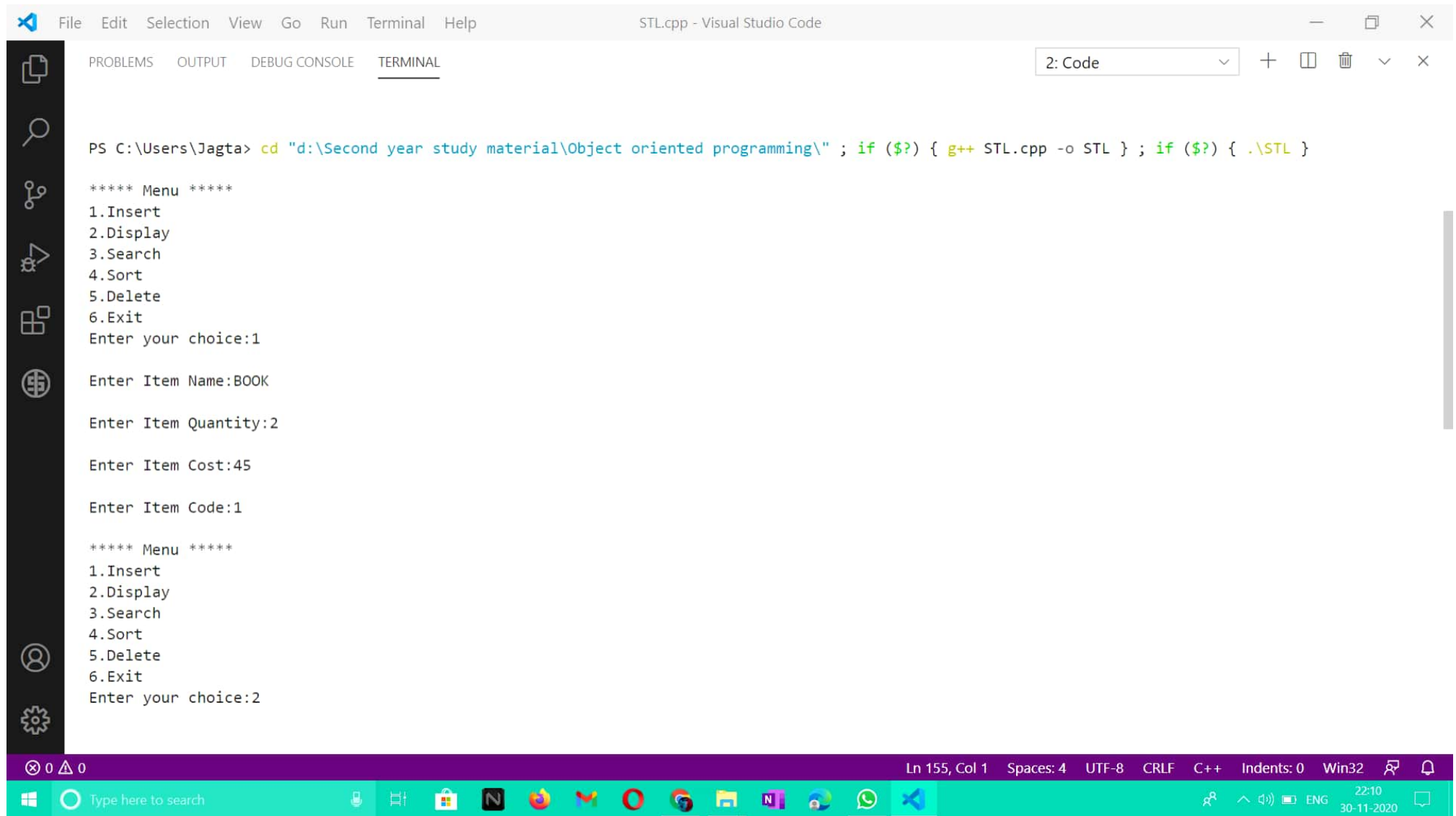
```
74 sort(o1.begin(),o1.end(),compare);
75 cout<<"\n\n Sorted on Cost";
76 display();
77 break;
78
79 case 5:
80     dlt();
81     break;
82
83 case 6:
84     exit(0);
85 }
86
87 }while(ch!=7);
88
89 return 0;
90 }
91
92 void insert()
93 {
94     Item i1;
95     cout<<"\nEnter Item Name:";
96     cin>>i1.name;
97     cout<<"\nEnter Item Quantity:";
98     cin>>i1.quantity;
99     cout<<"\nEnter Item Cost:";
100    cin>>i1.cost;
101    cout<<"\nEnter Item Code:";
102    cin>>i1.code;
103    o1.push_back(i1);
104 }
105
106 void display()
107 {
108     for_each(o1.begin(),o1.end(),print);
109 }
110
111 void print(Item &i1)
112 {
113     cout<<"\n";
114     cout<<"\nItem Name:"<<i1.name;
```

The status bar at the bottom shows "Ln 155, Col 1", "Spaces: 4", "UTF-8", "CRLF", "C++", "Indents: 0", "Win32", and a system clock showing "22:12 30-11-2020". The Windows taskbar is visible at the very bottom with various application icons.



```
File Edit Selection View Go Run Terminal Help STL.cpp - Visual Studio Code
STL.cpp x
D: > Second year study material > Object oriented programming > STL.cpp > ...
116     cout<<"\nItem Cost:"<<i1.cost;
117     cout<<"\nItem Code:"<<i1.code;
118 }
119
120 void search()
121 {
122     vector<Item>::iterator p;
123     Item i1;
124     cout<<"\nEnter Item Code to search:";
125     cin>>i1.code;
126     p=find(o1.begin(),o1.end(),i1);
127     if(p==o1.end())
128     {
129         cout<<"\nNot found.";
130     }
131     else
132     {
133         cout<<"\nFound.";
134     }
135 }
136
137 void dlt()
138 {
139     vector<Item>::iterator p;
140     Item i1;
141     cout<<"\nEnter Item Code to delete:";
142     cin>>i1.code;
143     p=find(o1.begin(),o1.end(),i1);
144     if(p==o1.end())
145     {
146         cout<<"\nNot found.";
147     }
148     else
149     {
150         o1.erase(p);
151         cout<<"\nDeleted.";
152     }
153 }
154
155 |
```

Ln 155, Col 1 Spaces: 4 UTF-8 CRLF C++ Indents: 0 Win32 22:12 30-11-2020



The image shows a Visual Studio Code window with the 'TERMINAL' tab active. The terminal displays the execution of a C++ program named STL.cpp. The program starts by changing the directory to 'd:\Second year study material\Object oriented programming\' and then runs a compilation command: `if ($?) { g++ STL.cpp -o STL } ; if ($?) { .\STL }`. The program then prints a menu with six options: 1.Insert, 2.Display, 3.Search, 4.Sort, 5.Delete, and 6.Exit. The user enters '1' as their choice. The program then prompts for 'Enter Item Name:BOOK', 'Enter Item Quantity:2', 'Enter Item Cost:45', and 'Enter Item Code:1'. It then prints the menu again. The user enters '2' as their choice. The status bar at the bottom shows the file is at line 155, column 1, with 4 spaces, UTF-8 encoding, CRLF line endings, C++ language, 0 indents, and Win32 architecture. The system tray at the bottom shows the date and time as 30-11-2020, 22:10.

```
PS C:\Users\Jagta> cd "d:\Second year study material\Object oriented programming\" ; if ($?) { g++ STL.cpp -o STL } ; if ($?) { .\STL }

***** Menu *****
1.Insert
2.Display
3.Search
4.Sort
5.Delete
6.Exit
Enter your choice:1

Enter Item Name:BOOK

Enter Item Quantity:2

Enter Item Cost:45

Enter Item Code:1

***** Menu *****
1.Insert
2.Display
3.Search
4.Sort
5.Delete
6.Exit
Enter your choice:2
```

```
STL.cpp - Visual Studio Code
2: Code

Enter your choice:2

Item Name:BOOK
Item Quantity:2
Item Cost:45
Item Code:1
***** Menu *****
1.Insert
2.Display
3.Search
4.Sort
5.Delete
6.Exit
Enter your choice:3

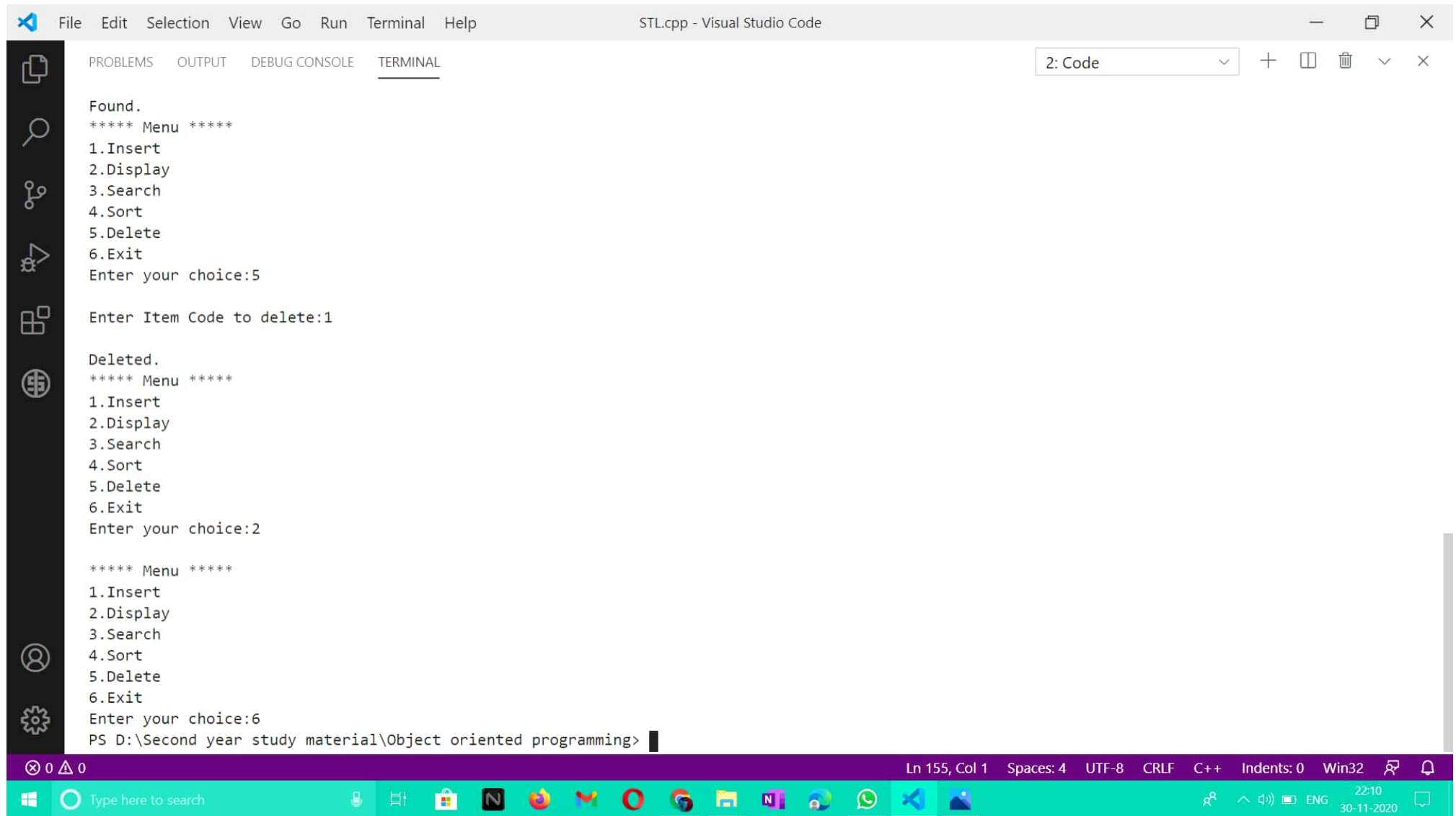
Enter Item Code to search:1

Found.
***** Menu *****
1.Insert
2.Display
3.Search
4.Sort
5.Delete
6.Exit
Enter your choice:5

Enter Item Code to delete:1

Deleted.
```

Ln 155, Col 1 Spaces: 4 UTF-8 CRLF C++ Indents: 0 Win32 22:10 30-11-2020



```
File Edit Selection View Go Run Terminal Help STL.cpp - Visual Studio Code
2: Code
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
Found.
***** Menu *****
1.Insert
2.Display
3.Search
4.Sort
5.Delete
6.Exit
Enter your choice:5

Enter Item Code to delete:1

Deleted.
***** Menu *****
1.Insert
2.Display
3.Search
4.Sort
5.Delete
6.Exit
Enter your choice:2

***** Menu *****
1.Insert
2.Display
3.Search
4.Sort
5.Delete
6.Exit
Enter your choice:6
PS D:\Second year study material\Object oriented programming>
```