



Oop

Name: Mahesh

Modern College of Engineering Jagtap

Shivajinagar, Pune 5. Roll no: 21027

Assignment no. 5

Title :- Function Template

Problem Statement :

Implement a function template selection sort. Write a program that inputs, sorts & outputs an ^{int} array and a float array.

Prerequisites :

Object oriented programming

• Templates in C++

Objectives :

To learn the concept of Template.

Theory :-

Templates :

Templates are the foundation of generic programming, which involves writing code in a way that is independent of any particular type.

A template is a blueprint or formula for creating a generic class or a function.

The library containers like iterators & algorithms are examples of generic programming.



& have been developed using template concept. There is a single definition of each container, such as vector, but we can define many different kinds of vectors for example, `vector<int>` or `vector<string>`.

You can use templates to define functions as well as classes, let us see how do they work:

Function Template:

The general form of a template function definition is shown here:

```
template <class type> rettype function name  
                                   (parameter list)  
{  
    // body of function  
}
```

Here, type is a placeholder name for a data type used by the function. This name can be used within the function definition. The following is the example that returns the max of two values:

3

Class template:

Just as we can define function templates, we can also define class templates. The general form of a generic class declaration is shown here:

```
template <class type> class-name.  
{  
    - - -  
}
```

Here, type is the placeholder type name, which will be specified when a class is instantiated. You can define more than one generic data type by using a comma-separated list.

Selection Sort:

Selection sort is a sorting algorithm, specifically an in-place comparison sort. It has $O(n^2)$ time complexity, making it inefficient on large lists, & generally performs worse than similar insertion sort.

Selection sort is noted for its simplicity, & it has performance advantages over more complicated algorithms in certain situations, particularly where auxiliary memory is limited.

④

How selection sort works?

example:

14	33	27	10	35	19	42	44
----	----	----	----	----	----	----	----

For the first position in the sorted list, the whole list is scanned sequentially. The first position where 14 is stored presently, we search the whole list & find that 10 is the lower value.

14	33	27	10	35	19	42	44
----	----	----	----	----	----	----	----

Do we replace 14 with 10. After one iteration, 10 which happens to be the minimum value in the list appears in the 1st position of sorted list.

10	33	27	14	35	19	42	44
----	----	----	----	----	----	----	----

for second position, where 33 is residing, we start scanning the rest of the list in linear manner.

10	33	27	14	35	19	42	44
----	----	----	----	----	----	----	----



Modern College of Engineering

Shivajinagar, Pune 5.

we find that 14 is the second lowest value in the list & it should appear at second place. we swap these values.

10	33	27	14	35	19	42	44
----	----	----	----	----	----	----	----

After two iterations, two least values positioned at the beginning in the sorted manner. The same process is applied on the rest of the items in the array.

10	14	27	33	35	19	42	44
----	----	----	----	----	----	----	----

10	14	19	33	35	27	42	44
----	----	----	----	----	----	----	----

10	14	19	33	35	27	42	44
----	----	----	----	----	----	----	----

10	14	19	27	35	33	42	44
----	----	----	----	----	----	----	----

10	14	19	27	35	33	42	44
----	----	----	----	----	----	----	----

10	14	19	27	35	33	42	44
----	----	----	----	----	----	----	----

10	14	19	27	33	35	42	44
----	----	----	----	----	----	----	----

10	14	19	27	33	35	42	44
----	----	----	----	----	----	----	----



Algo :-

1. start
2. Declare the template parameter T
3. Define template function for selection sort
4. In main() Define two arrays, one for integer & another for float & take a input for both arrays & call sorting function template to sort the no.
5. Stop.

Input :-

Enter how many elements you want :- 5

Enter integers :- 5 8 9 3 7

Enter float :- 9.4 3.8 5.5 2.2 6.7

O/p :-

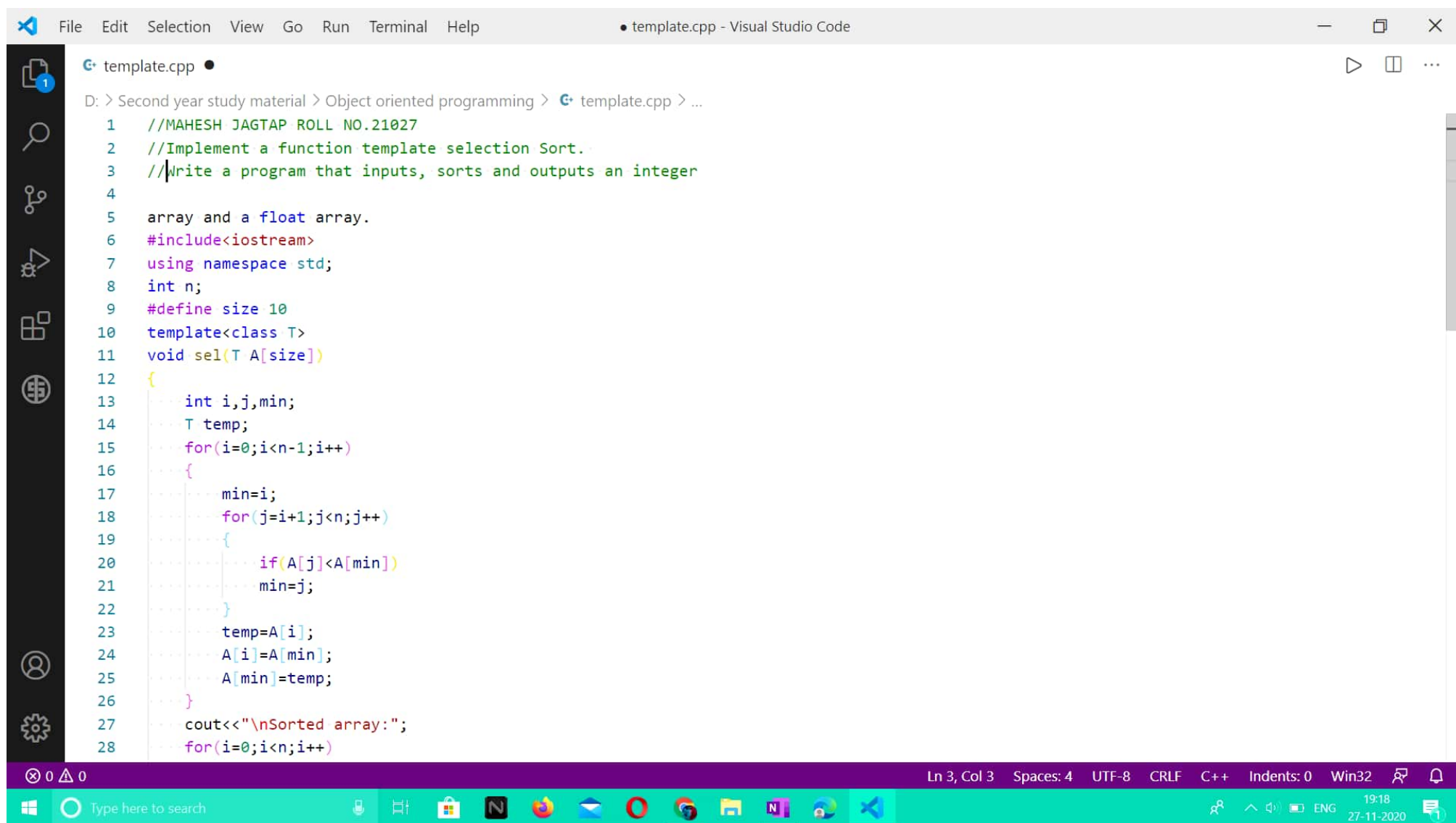
sorted list :-

3 5 7 8 9

2.2 3.8 5.5 6.7 9.4

Conclusion :-

In this assignment, we studied the concept of template & using it performed the program of selection sort on integers & float array.



The image shows a screenshot of the Visual Studio Code editor interface. The title bar at the top indicates the file is 'template.cpp' in Visual Studio Code. The menu bar includes File, Edit, Selection, View, Go, Run, Terminal, and Help. The left sidebar contains icons for Explorer, Search, Source Control, Run and Debug, Extensions, and Settings. The Explorer view shows the file 'template.cpp' under the path 'D: > Second year study material > Object oriented programming > template.cpp > ...'. The main editor area displays the following C++ code:

```
1 //MAHESH JAGTAP ROLL NO.21027
2 //Implement a function template selection Sort.
3 //Write a program that inputs, sorts and outputs an integer
4
5 array and a float array.
6 #include<iostream>
7 using namespace std;
8 int n;
9 #define size 10
10 template<class T>
11 void sel(T A[size])
12 {
13     int i,j,min;
14     T temp;
15     for(i=0;i<n-1;i++)
16     {
17         min=i;
18         for(j=i+1;j<n;j++)
19         {
20             if(A[j]<A[min])
21                 min=j;
22         }
23         temp=A[i];
24         A[i]=A[min];
25         A[min]=temp;
26     }
27     cout<<"\nSorted array:";
28     for(i=0;i<n;i++)
```

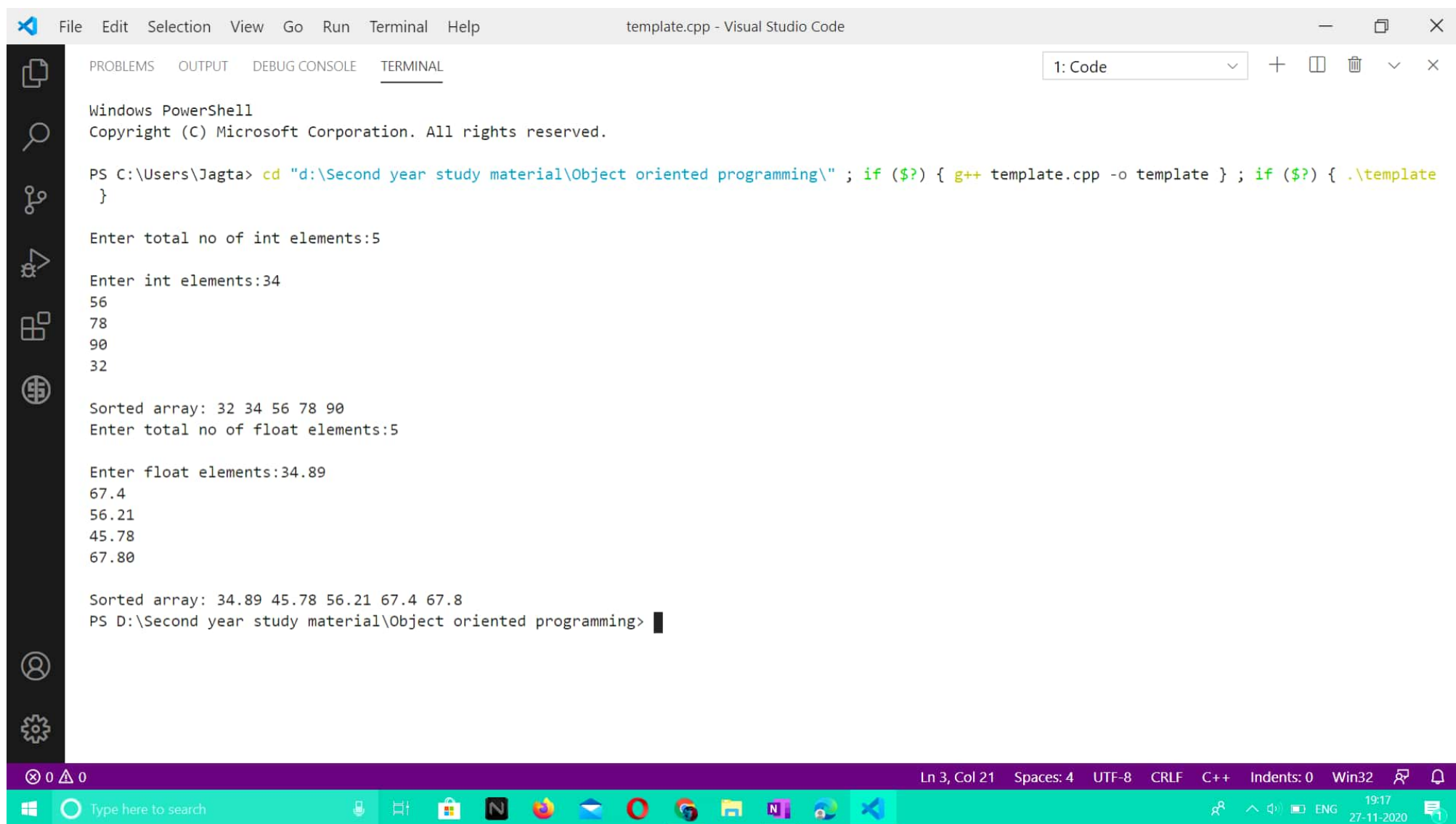
The status bar at the bottom shows 'Ln 3, Col 3', 'Spaces: 4', 'UTF-8', 'CRLF', 'C++', 'Indents: 0', 'Win32', and the system clock '19:18 27-11-2020'.

```
File Edit Selection View Go Run Terminal Help
• template.cpp - Visual Studio Code

D: > Second year study material > Object oriented programming > template.cpp > ...

30     cout<<" "<<A[i];
31 }
32 }
33
34 int main()
35 {
36     int A[size];
37     float B[size];
38     int i;
39
40     cout<<"\nEnter total no of int elements:";
41     cin>>n;
42     cout<<"\nEnter int elements:";
43     for(i=0;i<n;i++)
44     {
45         cin>>A[i];
46     }
47     sel(A);
48
49     cout<<"\nEnter total no of float elements:";
50     cin>>n;
51     cout<<"\nEnter float elements:";
52     for(i=0;i<n;i++)
53     {
54         cin>>B[i];
55     }
56     sel(B);
57 }
```

Ln 3, Col 3 Spaces: 4 UTF-8 CRLF C++ Indents: 0 Win32 19:18 27-11-2020



The image shows a Visual Studio Code window with a terminal open. The terminal is running a C++ program that sorts an array of integers and a set of floats. The code is written in a Windows PowerShell prompt. The program prompts the user to enter the total number of integer elements (5), then the integer elements themselves (34, 56, 78, 90, 32). It then prompts for the total number of float elements (5), followed by the float elements (34.89, 67.4, 56.21, 45.78, 67.80). Finally, it displays the sorted arrays: integers [32, 34, 56, 78, 90] and floats [34.89, 45.78, 56.21, 67.4, 67.8].

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

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

Enter total no of int elements:5

Enter int elements:34
56
78
90
32

Sorted array: 32 34 56 78 90
Enter total no of float elements:5

Enter float elements:34.89
67.4
56.21
45.78
67.80

Sorted array: 34.89 45.78 56.21 67.4 67.8
PS D:\Second year study material\Object oriented programming>
```