## **Experiment - 1.1**

Student Name: Pankaj Singh Kanyal UID: 20BCS6668

Branch: AIML Section/Group : AIML-4B

**Semester:** 5<sup>th</sup> **Date:** 24/08/2022

Subject Name: Advance Programming Lab Subject Code: 20CSV-334 20AML-4

#### 1.Aim/Overview of the Practical

Write a Program in C++ to find the K<sup>th</sup> smallest/largest element in an array using template and C++ STL.

## 2.Task to be done

**a.** Take input from the user and store the element in an array.

**b.** Write a program to find the  $k^{th}$  smallest element in the array.

## 3. Program Code:

```
#include<bits/stdc++.h>
using namespace std;

template<typename T>
T ksmallestelement(T arr[],int k)
{
    set<T>s;
    int size = sizeof(arr)/sizeof(arr[0]);

for(int i=0;i<size;i++)
{
    s.insert(arr[i]);
}</pre>
```



```
auto it = s.begin();
for(int i=0;i<k;i++)</pre>
it++;
return *it;
}
int main()
vector<float> f = {1.1,0.1,4.5,6.8};
vector<int> i = \{1,2,3,3,4,2,1,1,1,-1\};
float f1[] = \{1.1, 0.1, 4.5, 6.8\};
int i1[] = \{1,2,3,3,4,2,1,1,1,-1\};
int arr[] =
\{11,1,1,1,1,232,2,2,2,1,1,2,32,1,2,3,1,1231,1,2,3,1,2,41,3,3,
2,12,3,2,1,23,2,32};
//
cout<<"The Kth Smallest Element :</pre>
"<<ksmallestelement(arr,2)<<endl;
return 0;
}
```

## 4. Output:

## 5. Observation / Discussion

- Template is a blue print of a class or a function which takes data type as the parameter.
- The main idea of using a template is to make a data-type independent program.
- Templates are expanded at compiler time. Template is similar to macros.
- We have created one function named as kthsmalllest() which take a array and the number of smallest element to find.

#### 6. Result:

We have successfully implemented and find the smallest kth element in an array. Using set and vector in combination with the template function to make our program generic and independent of the data-typed used.

# 7. Evaluation Grid (To be created as per the SOP and Assessment guidelines by the faculty):

| Sno. | Parameters | Obtained Marks | Maximum marks |
|------|------------|----------------|---------------|
| 1.   |            |                |               |
| 2.   |            |                |               |
| 3.   |            |                |               |
| 4.   |            |                |               |







