Abstractions and Paradigms of Programming

Bachelor of Engineering

Computer Engineering

Set in STL

Sets are a type of associative containers in which each element has to be unique because the value of the element identifies it. The values are stored in a specific order.

- **Properties:**

1. The set stores the elements in **sorted** order.
2. All the elements in a set have **unique values**.
3. The value of the element cannot be modified once it is added to the set, though it is possible to remove and then add the modified value of that element. Thus, the values are **immutable**.
4. Sets follow the **Binary search tree** implementation.
5. The values in a set are **unindexed**.

Set in STL

```
Main(){
Set<string> s;
s.insert(“abc”);
s.insert(“pqr”);
s.insert(“xyz”);
Auto it = s.find(“abc”);
If(it=s.end()){
Cout<<“No such value”;
Else{
Cout<<(*it);
}
}
}
```

Printing the elements of a set

Range based loops

```
For(string value:s){
Cout<<value<<endl;
}
```

Auto Keyword

```
For(auto it=s.begin();it!=s.end();it++)
Cout<<(*it)<<endl;
```

Question on ordered sets

Given n strings, print unique strings in lexographical order. N<=10^5,|s|=10000.

```
Main(){
Set<string>s;
Int n; //number of strings
Cin>>n;
For(int i=0;i<n;i++){
String str;
Cin>>str;
s.insert(str);
}
For(auto value : s){
Cout<<value<<endl;
}}
```

Multiset

Multisets are a type of containers similar to the set, with the exception that multiple elements can have the same values.

```
Main(){
Multiset <String>m;
m.insert(“abc”);
m.Insert(“pqr”);
m.insert(“xyz”);
m.insert(“abc”);
For(string value:s){
Cout<<value<<endl;
}
}
```

Find and Erase in Multiset

```
Main(){
Multiset <String>m;
m.insert(“abc”);
m.Insert(“pqr”);
m.insert(“xyz”);
m.insert(“abc”);
Auto it = m.find(“abc”);
If(it!=m.end()){
Cout<<(*it);
}
Else{
Cout<<“No such value”;
}
```

Erase() in multiset

```
Main(){
Multiset <String>m;
m.insert(“abc”);
m.Insert(“pqr”);
m.insert(“xyz”);
m.insert(“abc”);
Auto it = m.find(“abc”);
If(it!=m.end()){
m.erase(it);
}
Else{
Cout<<“No such value”;
}
m.erase(“abc”); //deletes all the abcs.
```