STL

STANDART TEMPLATE LIBRARY

STL

#include <bits/stdc++.h>

Containers

- vector
- queue
- stack
- priority_queue
- set
- map

Algorithms

- sort
- search
- binary search
- count

Iterators

- .begin()
- .end()

vector

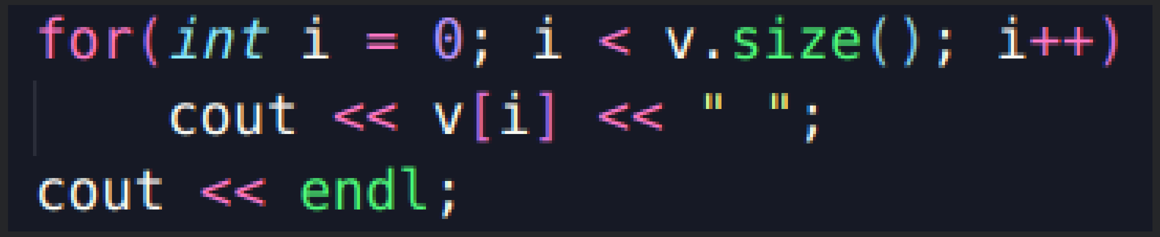
```
vector<tipo> nome;
vector<tipo> nome(tamanho);
vector<tipo> nome(tamanho, valor_inicial);
```

1. v.push_back(x)
2. v.resize(N)
3. v.assign(N, x)
4. v.clear()
5. v.size()

vector

for(int i=0; i<v.size(); i++)

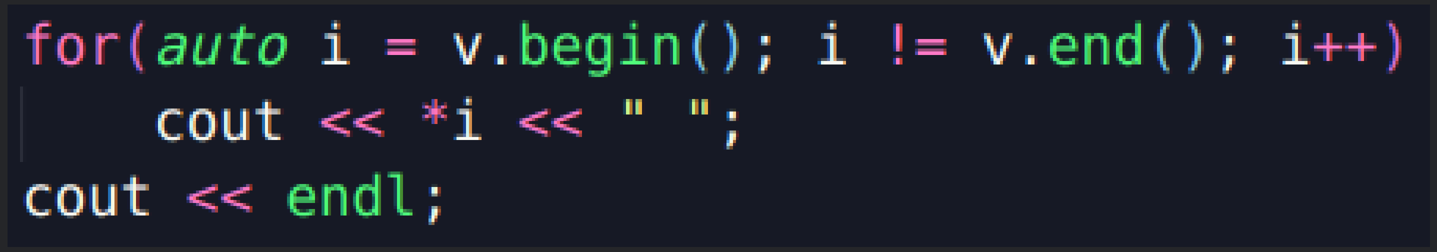
v[i]



```
for(int i = 0; i < v.size(); i++)
    cout << v[i] << " ";
cout << endl;
```

for(auto i=v.begin(); i !=v.end(); i++)

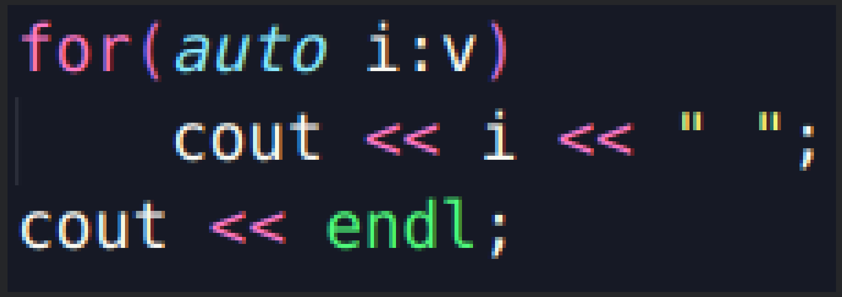
*i



```
for(auto i = v.begin(); i != v.end(); i++)
    cout << *i << " ";
cout << endl;
```

for(auto i : v)

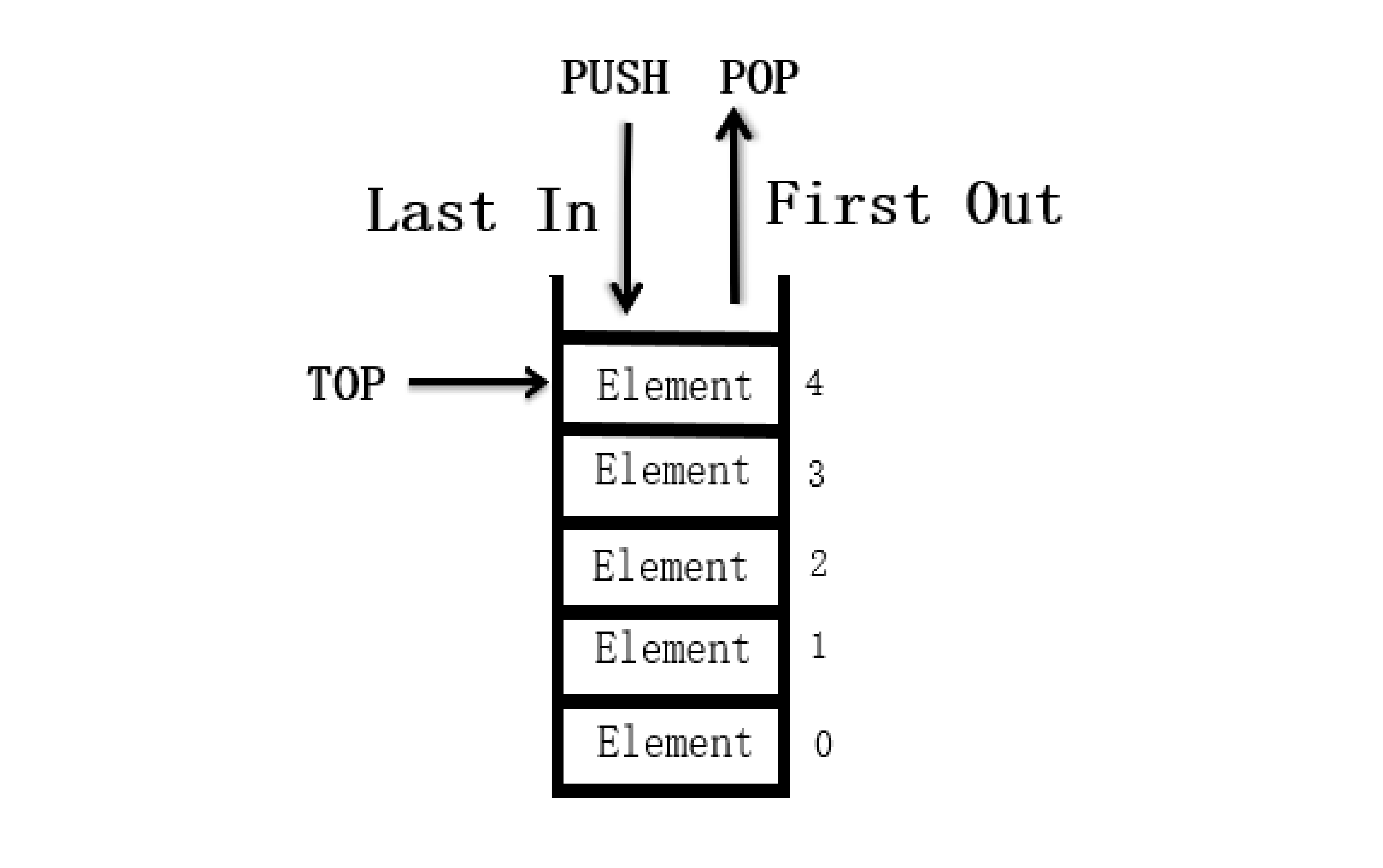
i



```
for(auto i:v)
    cout << i << " ";
cout << endl;
```

stack

stack<tipo> nome;



1 **s.push(x)**

2 **s.top()**

3 **s.pop()**

4 **s.size()**

5 **s.empty()**

queue

```
queue<tipo> nome;
```

1. q.push(x)
2. q.front()
3. q.pop()
4. q.size()
5. q.empty()

priority_queue

priority_queue<tipo> nome;

priority_queue<tipo, vector<tipo>, greater<tipo>> nome;

1. **pq.push(x)**
2. **pq.top()**
3. **pq.pop()**
4. **pq.size()**
5. **pq.empty()**

pair

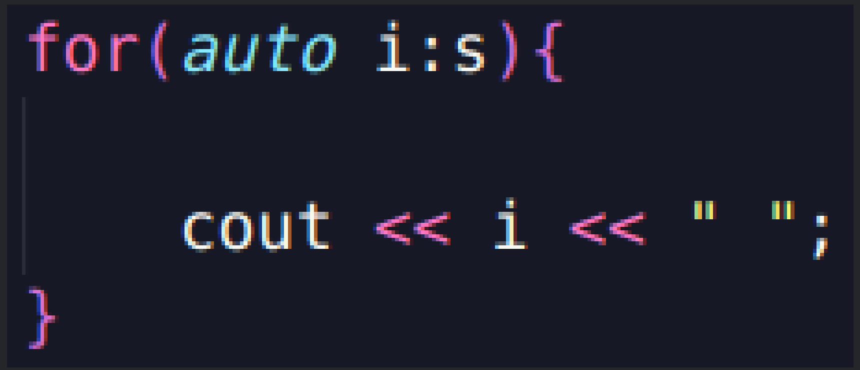
pair<tipo1, tipo2> nome;

pr.first;
pr.second;

set

set<tipo> nome;

st.insert(x);
st.count(x);
st.clear();
st.erase(x);



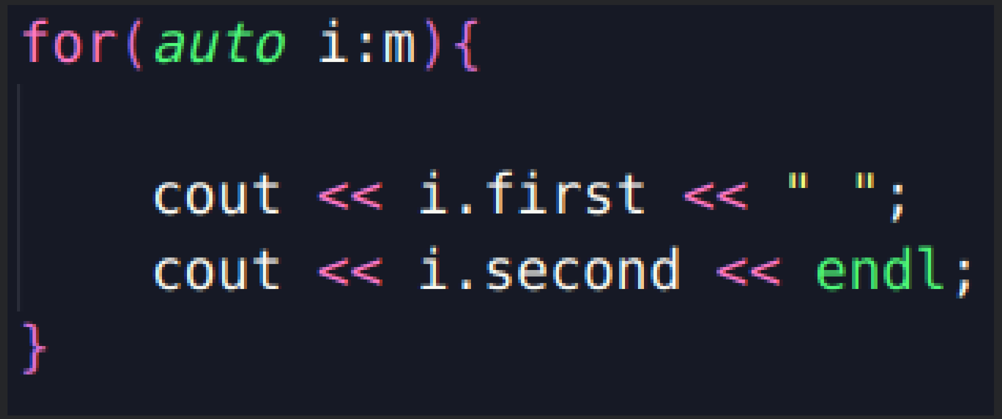
```
for(auto i:s){

    cout << i << " ";
}
```

map

map<tipo1, tipo2> nome;

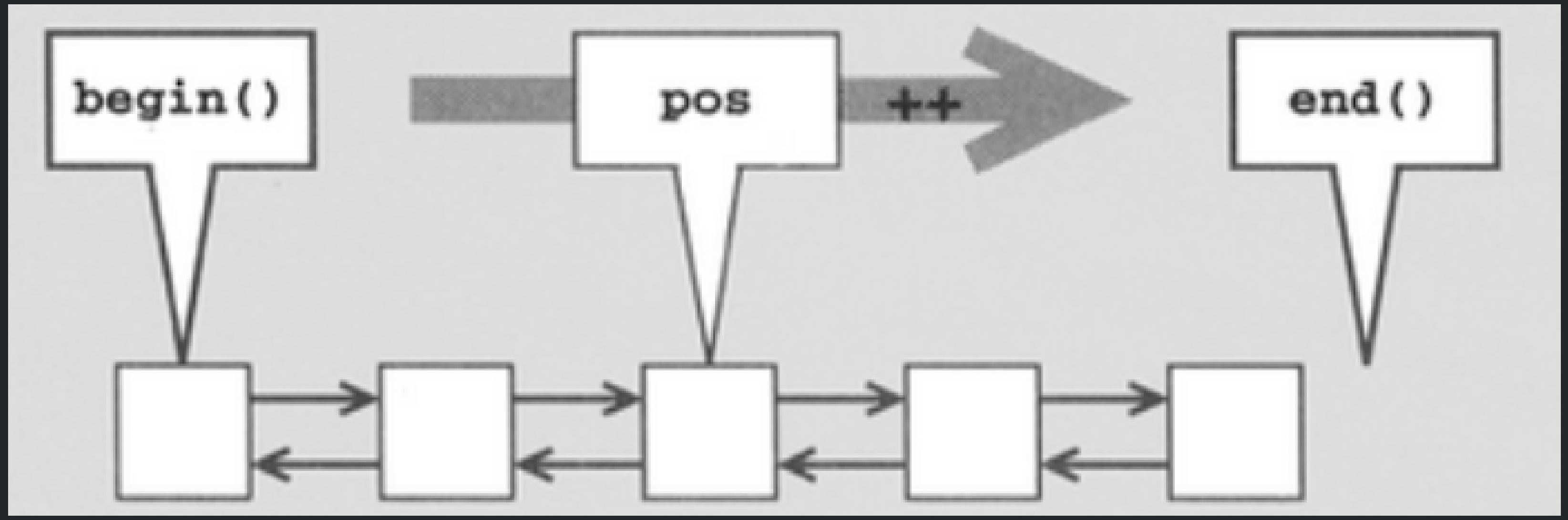
mp[key] = value;
mp.insert({key, value});
mp.erase(key);
mp.count(key);
mp.clear();



```
for(auto i:m){

    cout << i.first << " ";
    cout << i.second << endl;
}
```

iterators



1 v.begin()

2 v.end()

3 v.next()

4 v.begin() += n;

algorithms

sort

sort(first_iterator, last_iterator);

reverse

reverse(first_iterator, last_iterator);

erase

erase(first_iterator, last_iterator);

unique

unique(first_iterator, last_iterator);

*max_element

*max_element(first_iterator, last_iterator);

*min_element

*min_element(first_iterator, last_iterator);

https://cplusplus.com/