Dia 1 - Estruturas de dados

Emanuel Huber - emanuel.tesv@gmail.com

Container

Types of containers

Simple	Sequences	Containers Adaptors	Associative Containers	Other Types
pair	vector	queue	set/multiset	valArray
	list (double)	priority queue	map/multimap	bitset
	slist	stack	Hashmap	
	deque			

Mais informações: http://www.cplusplus.com/reference/stl/
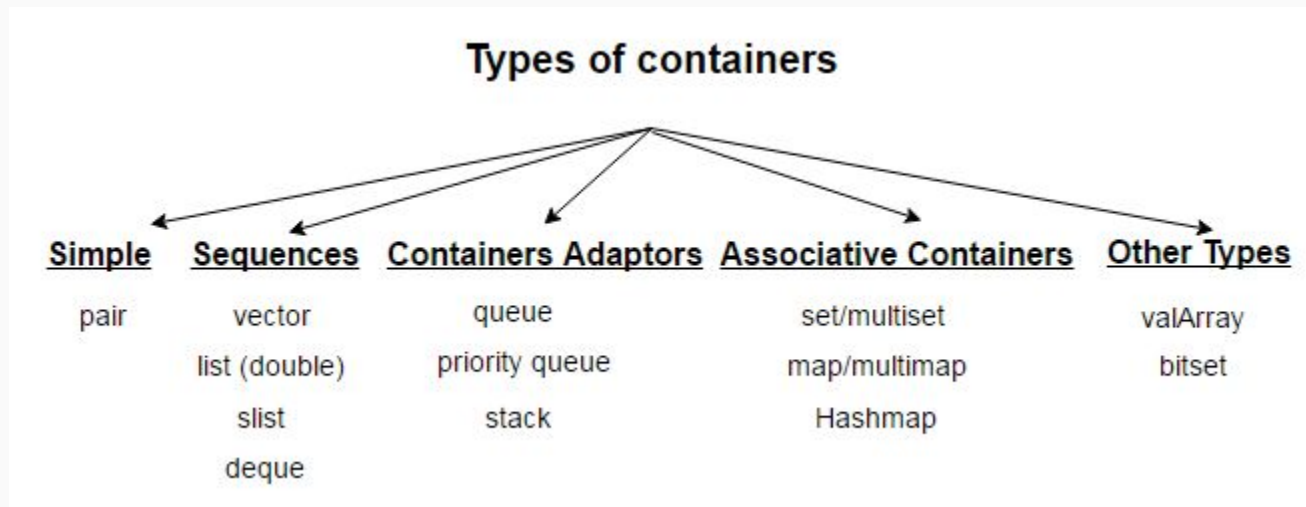
Simple container - Pair

```cpp
pair<int, string> p;
p = make_pair(1, "Pie");
cout << p.first << "\t" << p.second << endl;


pair<int, pair<int, string>> p2;
p2 = make_pair(2, p);
cout << p2.first << "\t" << p2.second.second << endl;
```

Sequencial container - vector

```cpp
vector<int> v (10, 0);
for (int i = 0; i < static_cast<int>(v.size()); ++i)
    cout << v[i] << "\t";
cout << endl;


vector<pair<char, double>> v1;
v1.push_back(make_pair('a', 2.1));
cout << v1[0].first << "\t" << v1[0].second << endl;
```

Container adaptors - queue

```cpp
queue<int> Q;
Q.push(1);
Q.push(2);
Q.push(3);
while (not Q.empty())
{
    cout << Q.front() << endl;
    Q.pop();
}
```

Container adaptors - stack

```cpp
stack<int> S;
S.push(1);
S.push(2);
S.push(3);
while (not S.empty())
{
    cout << S.top() << endl;
    S.pop();
}
```

Container adaptors - priority_queue

```cpp
priority_queue<int> Q;
Q.push(1);
Q.push(2);
Q.push(3);
while (not Q.empty())
{
    cout << Q.top() << endl;
    Q.pop();
}
```

Associative containers - set

```cpp
set<int> S;


S.insert(1);
S.insert(2);
S.insert(1);


for (auto it = S.begin(); it != S.end(); ++it)
    cout << *it << endl;
```

Associative containers - map

```cpp
map<string, double> M;

M["Pie"] = 10;
M["Cake"] = 0;

for (auto it = M.begin(); it != M.end(); ++it)
    cout << it->first << " - " << it->second << endl;
```

Sort

```cpp
vector<int> v = {6, 4, 3, 2, 1, 7, 7};


sort(v.begin(), v.end());


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

Comparator

```cpp
bool compare(const int a, const int b) {
    return a*a > b;
};


//main
vector<int> v = {6, 4, 3, 2, 1, 7, 7};
sort(v.begin(), v.end(), compare);


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

Macros

```
#define s(n) scanf("%d",&n)
#define ss(n) scanf(" %[^\n]", n);
#define INF (int)1e9
#define foreach(container, it) for(auto it = container.begin(); it!=container.end(); it++)
#define checkbit(n,b) ( (n >> b) & 1 )


vector<int> v (10, 1);
vector<int>::iterator it;
foreach(v, it)
    cout << *it << endl;
```

Mais macros: https://www.quora.com/What-are-some-macros-that-are-used-in-C++-programming-contests

Exercícios

Uri online judge: https://www.urionlinejudge.com.br/judge/en/tournaments/rank/746