Data Structures

Vector, Stack, Queue

Mostafa S. Ibrahim
Teaching, Training and Coaching since more than a decade!

Artificial Intelligence & Computer Vision Researcher
PhD from Simon Fraser University - Canada
Bachelor / Msc from Cairo University - Egypt
Ex-(Software Engineer / ICPC World Finalist)

STL

- C++ STL is very rich with the common data structures
- From now-on try to use them instead of our codes
- A great advantage is they use templates, which allows using it with different data types
- You don’t need to understand templates, just emulate how to code them.

```
3 #include<iostream>
4 #include<vector>
5 #include<stack>
6 #include<queue>
7 using namespace std;
```

Vector

```
 9 void play(vector<int> &v) {
10 }
11
12 void test_vector() {
13     vector<int> v1;
14
15     v1.push_back(30);
16     v1.push_back(10);
17     v1.push_back(20);
18     // Now we have 3 elements only
19
20     for (int i = 0; i < (int) v1.size(); ++i) {
21         cout << v1[i] << " ";   // 30 10 20
22     }
23     cout << "\n";
24
25     // vector of 3 strings
26     vector<string> str_vec { "mostafa", "ali", "me" };
27     // We can make nested vectors similar to 2D/3D arrays
28 }
```

Stack

```
30 void test_stack() {
31     stack<string> s;
32     s.push("ibrahim");
33     s.push("saad");
34     s.push("mostafa");
35
36     while (!s.empty()) {
37         cout << s.top() << " ";
38         s.pop();
39     }
40     cout << "\n";
41     // mostafa saad ibrahim
42 }
```

Queue

```
44 void test_queue() {
45     queue<int> q;
46     q.push(10);
47     q.push(20);
48     q.push(30);
49
50     cout << "Last element in Queue: " << q.back() << "\n"
51
52     cout << "Queue elements: ";
53     while (!q.empty()) {
54         cout << q.front() << " ";
55         q.pop();
56     }
57     cout << "\n";
58
59     queue<string> q_names;
60     q_names.push("mostafa");
61     string name = q_names.front();
62 }
```

Pair

- STL has a nice pair class that we may use a lot
- It has 2 members: first and second

```
64 void test_pairs() {
65     pair<int, string> p1 = make_pair(10, "ali");
66
67     cout << p1.first << " " << p1.second << "\n";   // 10 ali
68     p1.first += 3;
69
70     pair<float, pair<int, string>> p2 = make_pair(20.5, p1);
71     cout << p2.second.first << "\n";    // 13
72     p1 = p2.second;
73
74     vector<pair<int, string>> v1;
75     v1.push_back(p1);
76 }
```

"Acquire knowledge and impart it to the people."

"Seek knowledge from the Cradle to the Grave."