

컴퓨터공학 All in One

C/C++ 문법, 자료구조 및 심화 프로젝트 (나동빈)
제 55강 - C++ STL 시퀀스 컨테이너

학습 목표

C++ STL 시퀀스 컨테이너

- 1) C++ STL 시퀀스 컨테이너의 필요성과 사용 방법에 대해서 이해할 수 있습니다.

C++ STL 시퀀스 컨테이너

C++ STL 시퀀스 컨테이너

- 1) STL 시퀀스 컨테이너 라이브러리는 매우 활용도가 높은 자료구조를 제공합니다.
 - 2) 기존의 C언어를 이용하면 구현하기 까다로웠던 다양한 자료구조를 손쉽게 이용할 수 있습니다.
 - 3) 가장 많이 사용되는 시퀀스 컨테이너는 벡터(Vector)와 덱(Deque)입니다.
- Vector(벡터)
 - Deque(덱)
 - List(리스트)
 - Forward List(순방향 리스트)

C++ STL 시퀀스 컨테이너

C++ STL 시퀀스 컨테이너: 덱

덱(Deque)는 양 끝에서 데이터를 넣거나 뺄 수 있는 자료 구조(Data Structure)입니다.

- PUSH_FRONT: 덱의 앞에 데이터를 삽입합니다.
- POP_FRONT: 덱의 앞에서 데이터를 꺼냅니다.
- PUSH_BACK: 덱의 뒤에 데이터를 삽입합니다.
- POP_BACK: 덱의 뒤에서 데이터를 꺼냅니다.
- INSERT: 덱의 특정 위치에 데이터를 삽입합니다.

C++ STL 시퀀스 컨테이너

C++ STL 시퀀스 컨테이너: 덱

```
#include <iostream>
#include <deque>

using namespace std;

int main(void) {
    deque<int> d;
    d.push_front(3); d.push_back(7); d.pop_front(); d.push_front(4);
    for (int i = 0; i < d.size(); i++) { cout << d[i] << ' '; }
    cout << '\n';
    deque<int>::iterator iter;
    iter = d.begin();
    d.insert(iter + 1, 3, 5);
    iter = d.begin();
    d.insert(iter + 1, 1, 9);
    for (int i = 0; i < d.size(); i++) { cout << d[i] << ' '; }
    cout << '\n';
    d.clear();
    cout << d.empty() << '\n';
    system("pause");
}
```

C++ STL 시퀀스 컨테이너

C++ STL 시퀀스 컨테이너: 벡터

벡터(Vector)는 뒤쪽에서만 데이터를 넣거나 뺄 수 있는 자료 구조(Data Structure)입니다. 배열처럼 사용하기에 적합하다는 점에서 알고리즘 문제풀이에서 가장 많이 사용됩니다.

실제로 문제풀이에서는 앞뒤로 원소가 추가되는 경우가 적으며 임의 원소에 접근해야 하는 경우가 많아덱 자료구조와 비교했을 때 일반적인 경우 성능적으로 더 효율적입니다.

- PUSH_BACK: 벡터의 뒤에 데이터를 삽입합니다.
- POP_BACK: 벡터의 뒤에서 데이터를 꺼냅니다.
- INSERT: 벡터의 특정 위치에 데이터를 삽입합니다.

C++ STL 시퀀스 컨테이너

C++ STL 시퀀스 컨테이너: 벡터

```
#include <iostream>
#include <vector>

using namespace std;

int main(void) {
    vector<int> v;
    v.push_back(3); v.push_back(5); v.push_back(8);
    vector<int>::iterator iter;
    iter = v.begin();
    v.insert(iter + 1, 3, 7);
    for (int i = 0; i < v.size(); i++) {
        cout << v[i] << ' ';
    }
    cout << '\n';
    v.clear();
    cout << v.empty();
    system("pause");
}
```

배운 내용 정리하기

C++의 STL 시퀀스 컨테이너

- 1) C++에서는 STL 시퀀스 컨테이너에 속하는 Deque, Vector 등을 효과적으로 사용할 수 있습니다.