



실전 알고리즘 0x07강 덱

BaaaaaaaaaaaaaaaaarkingDog

목차



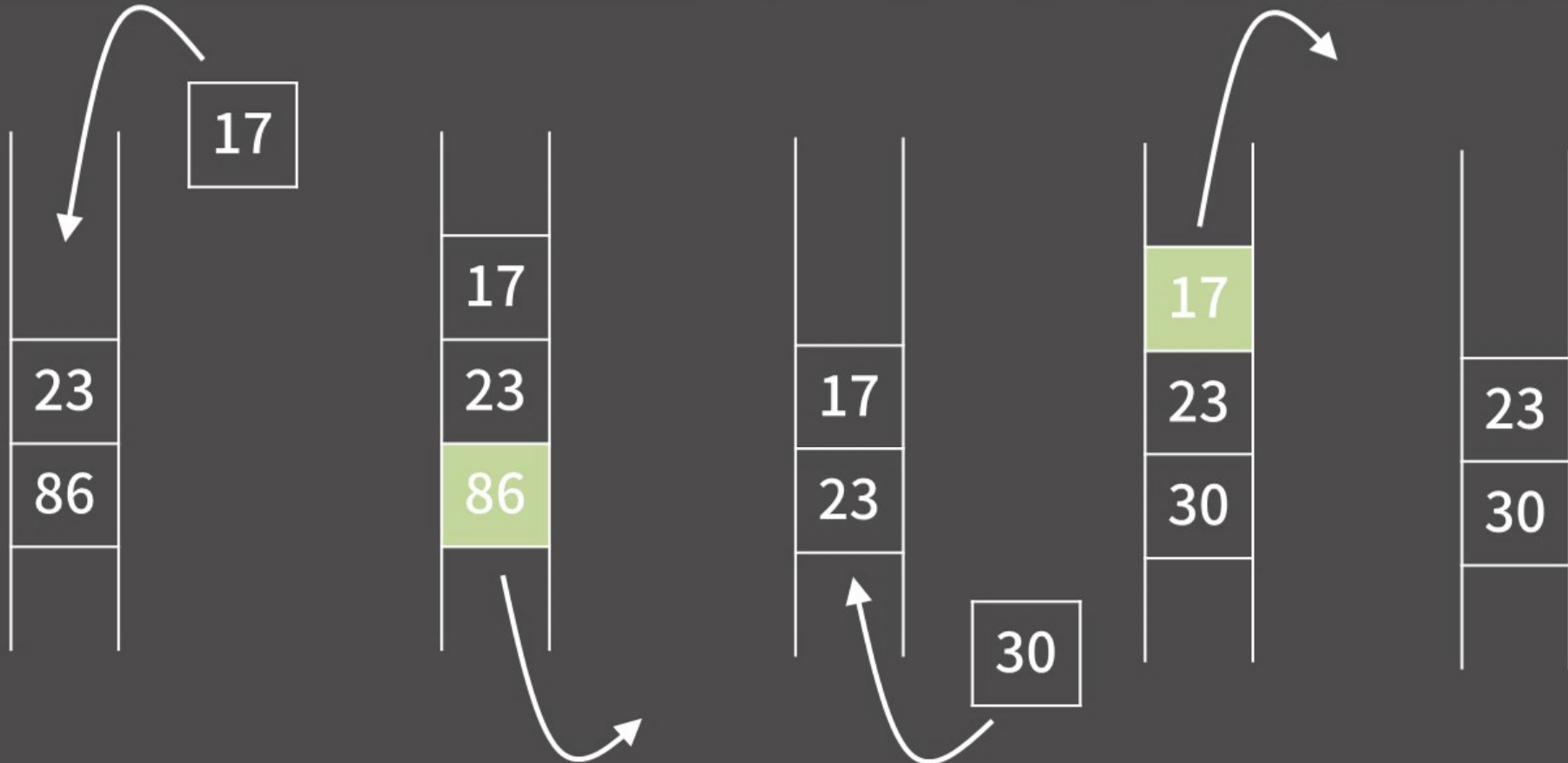
0x00 정의와 성질

0x01 기능과 구현

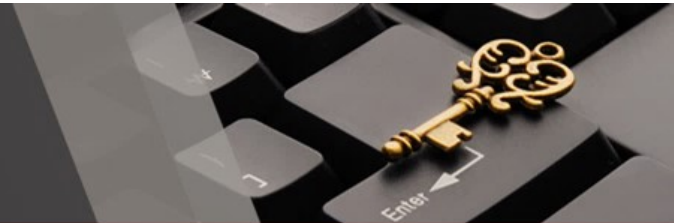
0x02 STL deque

0x03 연습문제

0x00 정의와 성질



0x00 정의와 성질



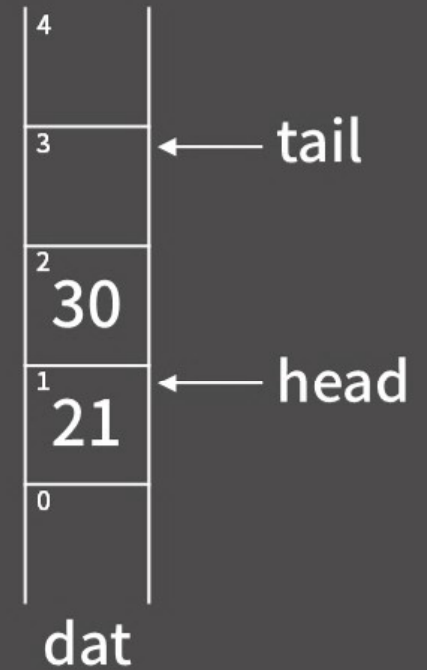
덱의 성질

1. 원소의 추가가 $O(1)$
2. 원소의 제거가 $O(1)$
3. 제일 앞/뒤의 원소 확인이 $O(1)$
4. 제일 앞/뒤가 아닌 나머지 원소들의 확인/변경이 원칙적으로 불가능

0x01 기능과 구현

구현

```
01  const int MX = 1000005;  
02  int dat[2*MX+1];  
03  int head = MX, tail = MX;
```



0x01 기능과 구현

구현



0x01 기능과 구현

구현

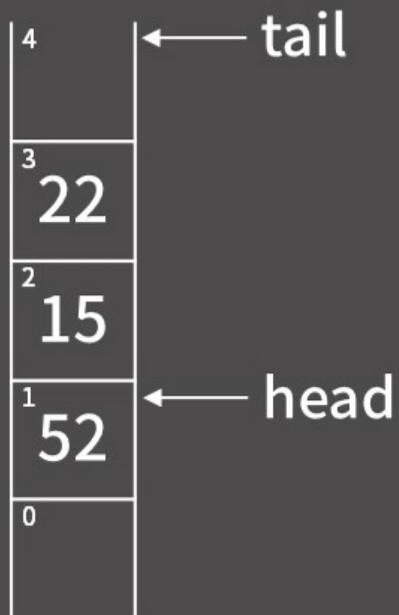
https://github.com/blisstoner/basic-algo-lecture-material/blob/master/0x07/deque_test.cpp

```
01  #include <bits/stdc++.h>
02  using namespace std;
03
04  const int MX = 1000005;
05  int dat[2*MX+1];
06  int head = MX, tail = MX;
07
08  void push_front(int x) {
09  }
10
11
12  void push_back(int x) {
13  }
14
15
16  void pop_front() {
17  }
18  }
```

```
19  void pop_back() {
20  }
21
22
23  int front() {
24  }
25
26
27  int back() {
28  }
29
30
31  void test() {
32      ...
33  }
34
35  int main(void) {
36      test();
37  }
```

0x01 기능과 구현

구현



```
01 void push_front(int x){
02     dat[--head] = x;
03 }
04
05 void push_back(int x){
06     dat[tail++] = x;
07 }
08
09 void pop_front(){
10     head++;
11 }
```

```
01 void pop_back(){
02     tail--;
03 }
04
05 int front(){
06     return dat[head];
07 }
08
09 int back(){
10     return dat[tail-1];
11 }
```

https://github.com/blisstoner/basic-algo-lecture-material/blob/master/0x07/deque_test_ans.cpp

0x02 STL deque

reference : <http://www.cplusplus.com/reference/deque/deque/>

https://github.com/blisstoner/basic-algo-lecture-material/blob/master/0x07/deque_example.cpp

```
01  #include <bits/stdc++.h>
02  using namespace std;
03
04  int main(void) {
05      deque<int> DQ;
06      DQ.push_front(10); // 10
07      DQ.push_back(50); // 10 50
08      DQ.push_front(24); // 24 10 50
09      for(auto x : DQ) cout<<x;
10      cout << DQ.size() << '\n'; // 3
11      if(DQ.empty()) cout << "DQ is empty\n";
12      else cout << "DQ is not empty\n"; // DQ is not empty
13      DQ.pop_front(); // 10 50
14      DQ.pop_back(); // 10
```

```
15      cout << DQ.back() << '\n'; // 10
16      DQ.push_back(72); // 10 72
17      cout << DQ.front() << '\n'; // 10
18      DQ.push_back(12); // 10 72 12
19      DQ[2] = 17; // 10 72 17
20      DQ.insert(DQ.begin()+1, 33); // 10 33 72 17
21      DQ.insert(DQ.begin()+4, 60); // 10 33 72 17 60
22      for(auto x : DQ) cout << x << ' ';
23      cout << '\n';
24      DQ.erase(DQ.begin()+3); // 10 33 72 60
25      cout << DQ[3] << '\n'; // 60
26      DQ.clear(); // DQ의 모든 원소 제거
27  }
```

0x03 연습문제

BOJ 10866번: 덱

https://github.com/blisstoner/basic-algo-lecture-material/blob/master/0x07/10866_1.cpp

```
01 #include <bits/stdc++.h>
02 using namespace std;
03
04 int main(void) {
05     ios::sync_with_stdio(0);
06     cin.tie(0);
07
08     deque<int> DQ;
09     int n;
10     cin >> n;
11     while (n-->0) {
12         string q;
13         cin >> q;
14         if (q == "push_back") {
15             int val;
16             cin >> val;
17             DQ.push_back(val);
18         }
```

```
19     else if (q == "push_front") {
20         int val;
21         cin >> val;
22         DQ.push_front(val);
23     }
24     else if (q == "pop_front") {
25         if(DQ.empty())
26             cout << -1 << '\n';
27         else{
28             cout << DQ.front() << '\n';
29             DQ.pop_front();
30         }
31     }
32     else if (q == "pop_back") {
33         if(DQ.empty())
34             cout << -1 << '\n';
35         else{
36             cout << DQ.back() << '\n';
37             DQ.pop_back();
38         }
39     }
```

```
40     else if (q == "size")
41         cout << DQ.size() << '\n';
42     else if (q == "empty")
43         cout << DQ.empty() << '\n';
44     else if (q == "front") {
45         if(DQ.empty())
46             cout << -1 << '\n';
47         else
48             cout << DQ.front() << '\n';
49     }
50     else { // back
51         if(DQ.empty())
52             cout << -1 << '\n';
53         else
54             cout << DQ.back() << '\n';
55     }
56 }
57 }
```

0x03 연습문제

BOJ 10866번: 덱

https://github.com/blisstoner/basic-algo-lecture-material/blob/master/0x07/10866_2.cpp

```
01 #include <bits/stdc++.h>
02 using namespace std;
03
04 const int MX = 1000005;
05 int dat[2*MX+1];
06 int head = MX, tail = MX;
07
08 ...
09
10 int main(void) {
11     ios::sync_with_stdio(0);
12     cin.tie(0);
13     int n;
14     cin >> n;
15     while (n--) {
16         string q;
17         cin >> q;
```

```
18         if (q == "push_back") {
19             int val;
20             cin >> val;
21             push_back(val);
22         }
23         else if (q == "push_front") {
24             int val;
25             cin >> val;
26             push_front(val);
27         }
28         else if (q == "pop_front") {
29             if (tail == head)
30                 cout << -1 << '\n';
31             else {
32                 cout << front() << '\n';
33                 pop_front();
34             }
35         }
36         else if (q == "pop_back") {
37             if (tail == head)
38                 cout << -1 << '\n';
```

```
39         else {
40             cout << back() << '\n';
41             pop_back();
42         }
43     }
44     else if (q == "size")
45         cout << DQ.size() << '\n';
46     else if (q == "empty")
47         cout << DQ.empty() << '\n';
48     else if (q == "front") {
49         if (DQ.empty())
50             cout << -1 << '\n';
51         else
52             cout << DQ.front() << '\n';
53     }
54     else { // back
55         if (DQ.empty())
56             cout << -1 << '\n';
57         else
58             cout << DQ.back() << '\n';
59     }
60 }
61 }
```


강의 정리

