

wifi 442. 5737 8435

Question @ startlink.io

C, C++, STL

Java

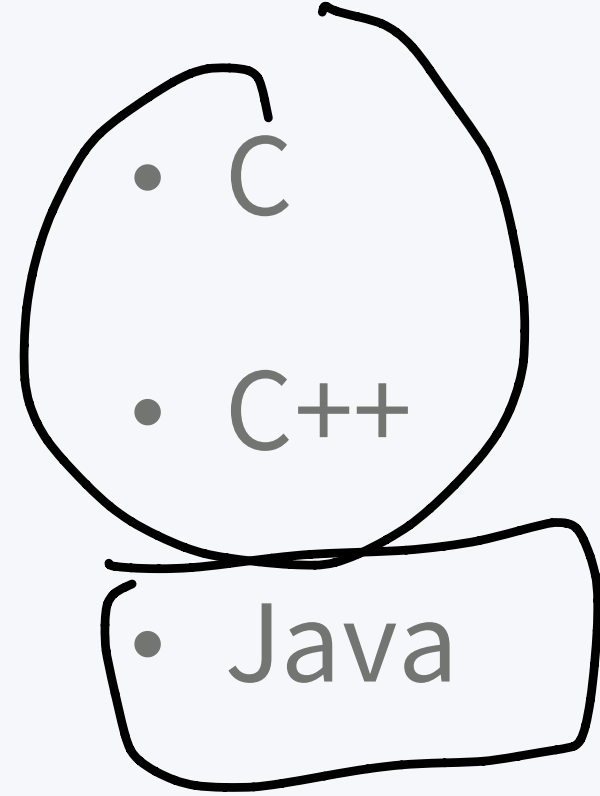
프로그래밍 언어

최백준 choi@startlink.io

프로그래밍 언어

프로그래밍 언어

Programming Language



2-1~

1E0

STL

Petr

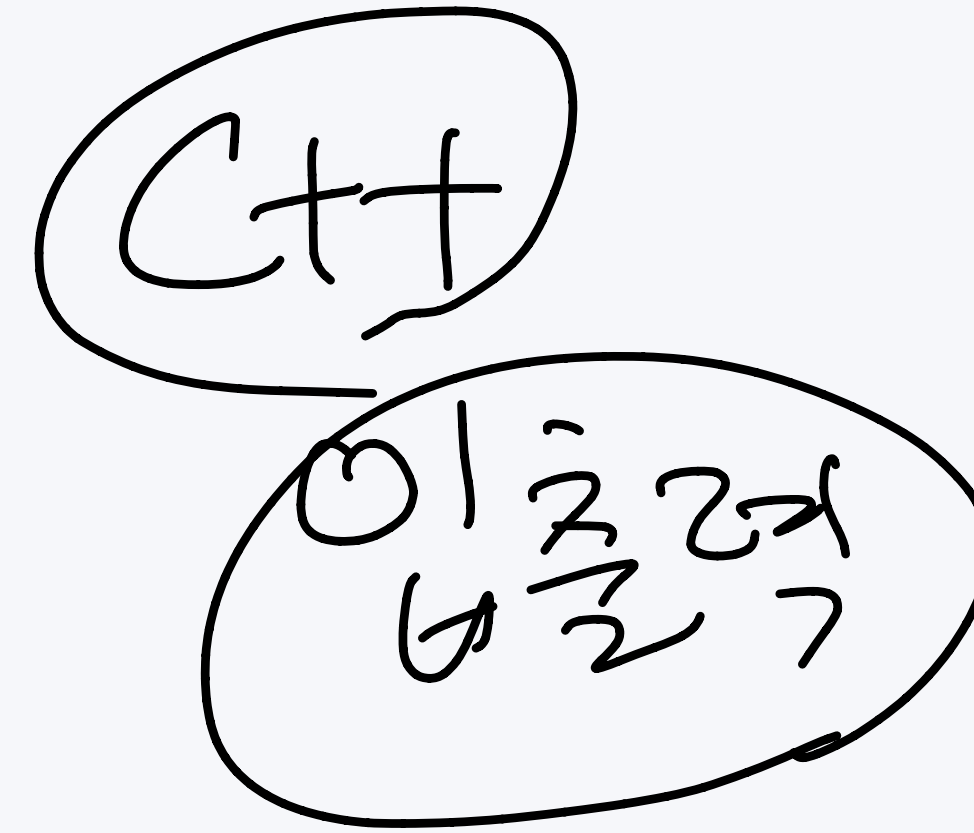
C++ \Rightarrow C# \rightarrow Java

C

scanf/printf

C

- scanf, printf는 입출력을 받는 함수



포맷 문자열

C

- `%d`: 정수
- `%i`: 정수 (형태에 따라 다르게 입력 받음)
- `%x`: 16진수 정수
- `%o`: 8진수 정수
- `%s`: 문자열
- `%c`: 문자
- `%f`: float
- `%lf`: double
- `%Lf`: long double

Short

%i

C

- %i는 수의 형태에 따라 다르게 입력 받는다.

```
#include <stdio>
```

```
int main() {
```

```
    int x,y;
```

```
    scanf("%d %i",&x,&y);
```

```
    printf("%d %d\n",x,y);
```

```
    return 0;
```

```
}
```

%i

C

- 10 10 -> 10 10
- 10 010 -> 10 8
- 10 0x10 -> 16

%i

C

- 연습문제: <https://www.acmicpc.net/problem/11816>

scanf의 리턴값

C

- scanf의 리턴값은 성공적으로 입력받은 인자의 개수이다.
- 따라서, 파일의 끝까지 입력받아야 하는 경우에는
- `while (scanf("%d %d",&a,&b) == 2)`
- 와 같이 사용할 수 있다.
- 연습 문제: <https://www.acmicpc.net/problem/10951>
-

파일의 끝

EOF

feof

10

공백과 줄바꿈은 무시한다

C

```
for (int i=0; i<5; i++) {  
    scanf("%d", &n);  
}
```

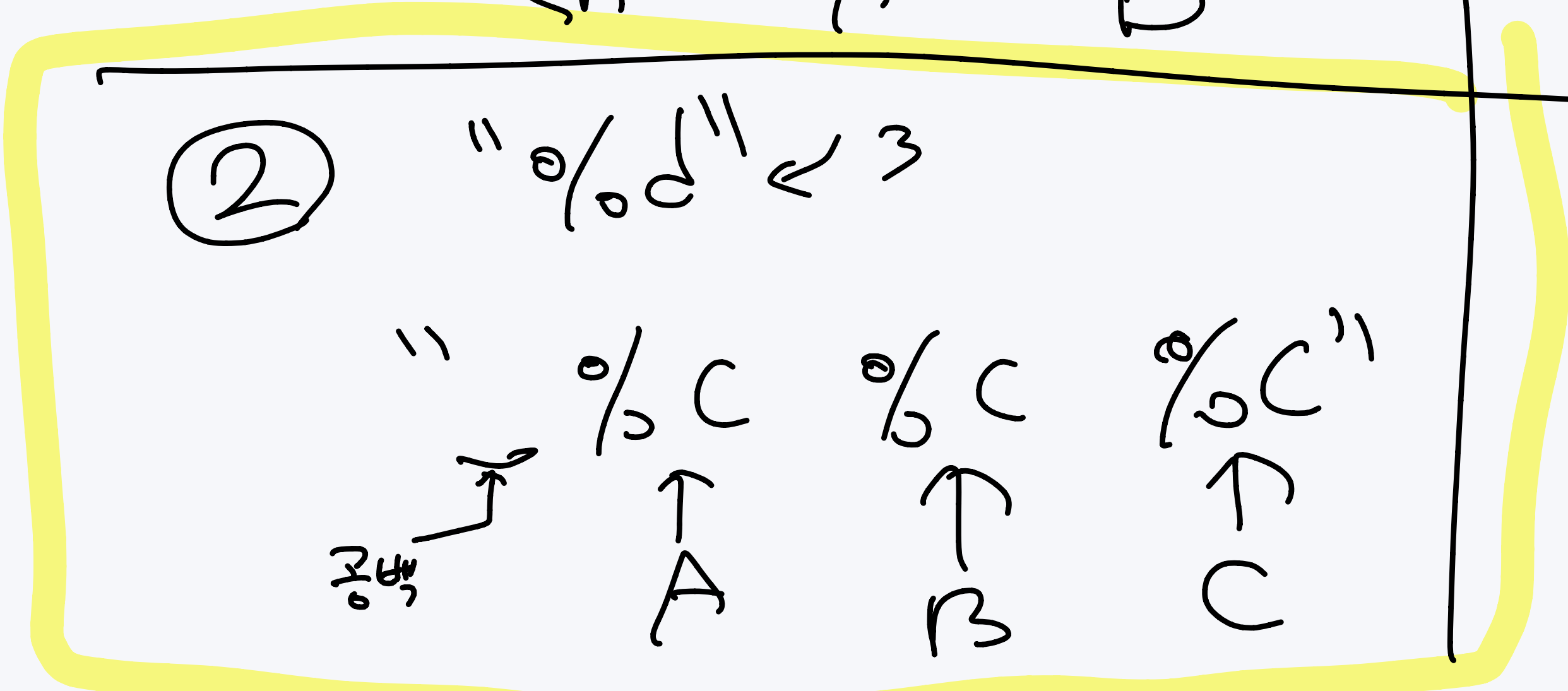
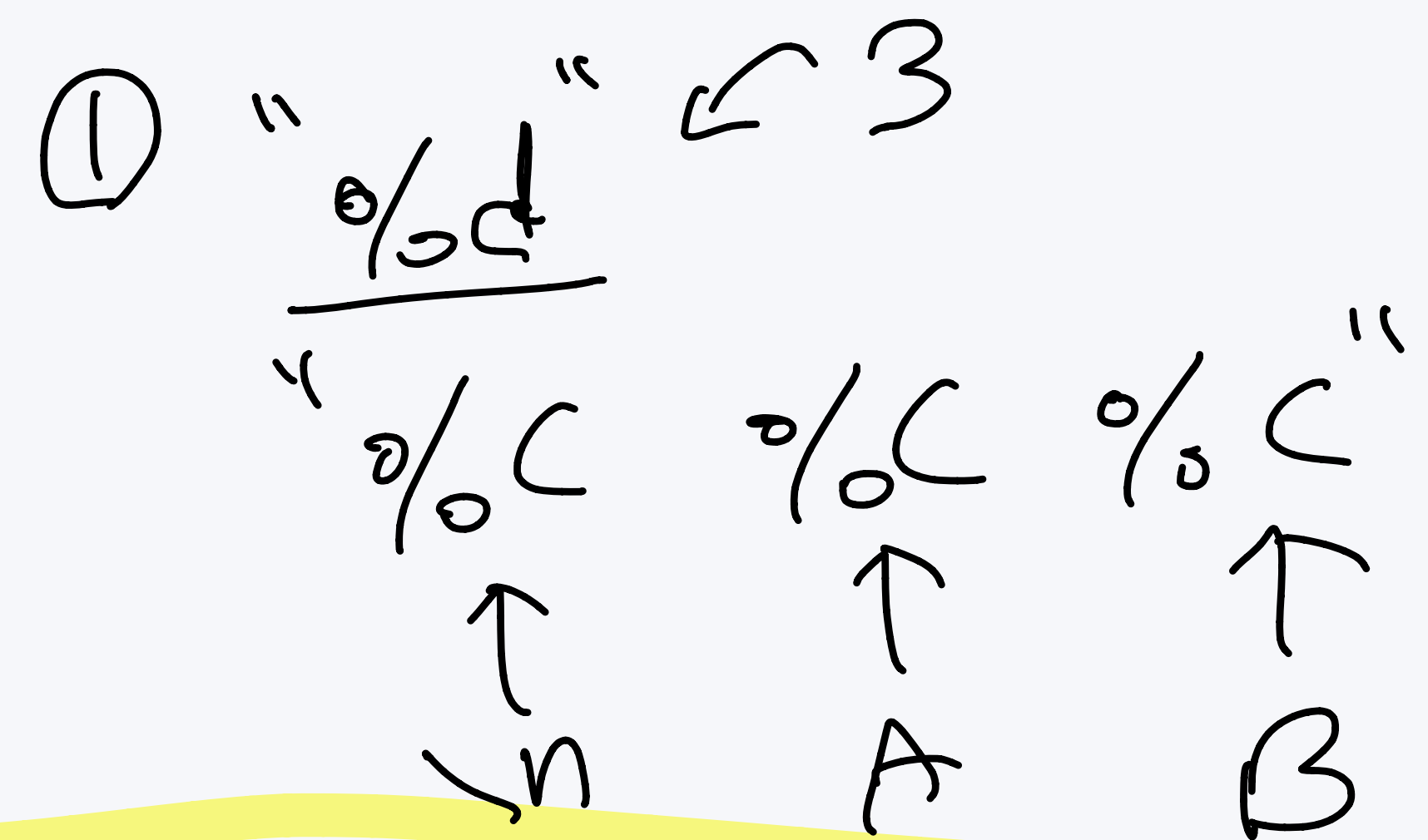
• 입력

10 20 30 40 50

• 입력

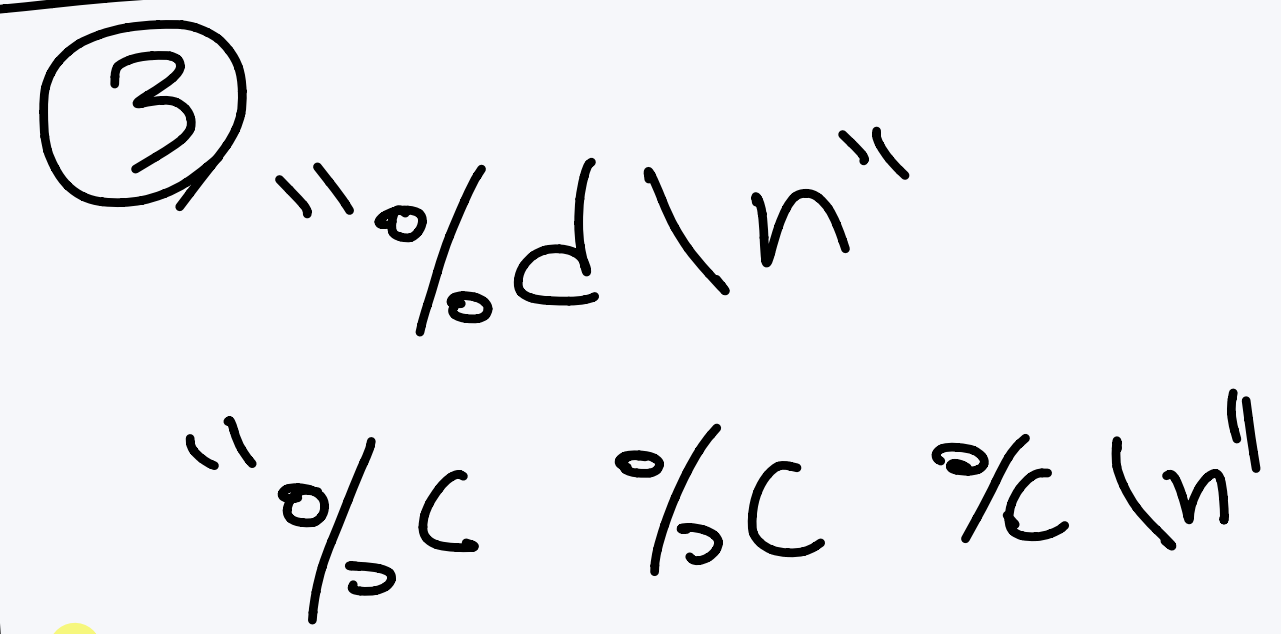
10
20 30
40
50

• 모두 같은 값이 들어간다



3

A	B	C
D	E	F
G	H	I



그대로 출력하기

<https://www.acmicpc.net/problem/11718>

```
scanf("%^\n\n", s);
```

- 줄바꿈을 입력받지 않기 때문에, 편리한 방법이지만, 각 줄의 앞 뒤에 있는 공백은 무시하고 입력을 받게 된다.
- 따라서, 빈 줄을 입력받을 수 없다.
- 또, 공백으로 시작하는 경우 공백을 무시하고 문자부터 입력받게 된다.
- 이 문제는 위의 두가지 경우가 없기 때문에 사용 가능.
- C/C++: <https://gist.github.com/Baekjoon/edc22d3d680a3bbd7ba7>

$\%s$: 12123456123
 $\%[123]$: 12123
12123456123

그대로 출력하기 2

<https://www.acmicpc.net/problem/11719>

```
scanf("%[^\\n]\\n", s);
```

- 줄바꿈을 입력받지 않기 때문에, 편리한 방법이지만, 각 줄의 앞 뒤에 있는 공백은 무시하고 입력을 받게 된다.
- 따라서, 빈 줄을 입력받을 수 없다.
- 또, 공백으로 시작하는 경우 공백을 무시하고 문자부터 입력받게 된다.
- 이 문제는 위의 두가지 경우가 있기 때문에 사용 불가능.
- C/C++: <https://gist.github.com/Baekjoon/66b4f9451a28dd416a33>

숫자의 합

<https://www.acmicpc.net/problem/11720>

```
scanf("%1d",&x);
```

- %d 사이에 수를 넣으면, 그 길이 만큼 입력을 받게 된다.
- 12345에 1, 2, 3, 4, 5 따로 따로 입력 받을 수 있다.
- C/C++: <https://gist.github.com/Baekjoon/0aba0c466380b9e10c2f>

%1d

%2d

12345
_ _ _ _
_ _ _ _

이제야 행려

10101

열 개씩 끊어 출력하기

<https://www.acmicpc.net/problem/11721>

```
scanf("%10s", s);
```

- %s의 경우도 개수를 지정해서 입력받을 수 있다.
- 만약, 입력받을 수 있는 것의 개수가 지정한 개수 보다 적으면 그만큼만 입력을 받게 된다.
- C/C++: <https://gist.github.com/Baekjoon/90de3b2f72d6300e3e5e>



C

```
#include <stdio>
```

```
int main() {
```

```
    int x,y;
```

```
    scanf("%d %*d %d", &x, &y);
```

Handwritten notes: An arrow points from the text "무시" (ignore) to the asterisk in the format string. To the right, there are three columns of handwritten text: "1", "2", and "3" at the top, and "x", "무시", and "y" below them, with arrows pointing from the format string to each column.

```
    printf("%d %d\n", x, y);
```

```
    return 0;
```

```
}
```

- *을 붙이면 입력은 받지만 변수에 저장하지는 않는다.

C++

getline

C++

String

```
getline(cin, s);
```

- 한 줄 다 입력 받기

setprecision

C++

```
#include <iomanip>
```

```
int main() {
```

```
    double f = 3.14159265358979;
```

```
    cout << setprecision(5) << f << '\n';
```

```
    cout << setprecision(8) << f << '\n';
```

```
    cout << setprecision(10) << f << '\n';
```

```
    return 0;
```

```
}
```

```
3.1416
```

```
3.1415927
```

```
3.141592654
```

5221 5221

cout << fixed << setp

endl
'\n'

STL

C++

20

- Standard Template Library
- 알고리즘
- 컨테이너
- 함수
- 이터레이터
- 로 이루어져 있다.

컨테이너

C++

21

- Standard Template Library
- 알고리즘
- 컨테이너
- 함수
- 이터레이터
- 로 이루어져 있다.

pair

pair

STL

- pair를 사용하면 두 자료형 T1과 T2를 묶을 수 있다
- 항상 두 개를 묶는다.
- 첫 번째 자료는 first
- 두 번째 자료는 second로 접근할 수 있다.

pair

STL

- `#include <utility>`
- 에 있는데,
- `algorithm`, `vector`와 같은 헤더파일에서 이미 `include` 하고 있기 때문에, 따로 `include` 하는 경우는 없다.
- `make_pair`를 이용하거나, 생성자를 이용해서 만들 수 있다.

pair

STL

first
second

pair 2
typedef pair<int, int>

```
pair<int, int> p1;
```

```
cout << p1.first << ' ' << p1.second << '\n';
```

10 20

```
p1 = make_pair(10, 20);
```

```
cout << p1.first << ' ' << p1.second << '\n';
```

10 20

```
p1 = pair<int, int>(30, 40);
```

pair(30, 40)

```
cout << p1.first << ' ' << p1.second << '\n';
```

30 40

```
pair<int, int> p2(50, 60);
```

```
cout << p2.first << ' ' << p2.second << '\n';
```

50 60

pair

STL

0 0

10 20

30 40

50 60

pair

STL

Struct, class

27

```
pair<pair<10int,20int>, pair<30int,40int>> p =  
make_pair(make_pair(10,20), make_pair(30,40));
```

```
cout << p.first.first10 << ' ' << p.first.second20 << ' ';&br/>cout << p.second.first30 << ' ' << p.second.second40 << '\n';
```

10 20 30 40

$\Rightarrow, \Leftarrow, <, <=, >, >=$

tuple

C++11

tuple

STL

29

- tuple은 pair와 같지만 여러 개를 묶을 수 있다
- .first, .second, .third, .fourth ... 가 아니고
- get을 이용해서 인덱스로 접근해야 한다
- tuple은 #include <tuple>에 정의되어 있다

tuple

STL

30

```
tuple<int, int, int> t1 = make_tuple(1, 2, 3);
```

```
cout << get<0>(t1) << ' '; 1  
cout << get<1>(t1) << ' '; 2  
cout << get<2>(t1) << '\n'; 3
```

```
/*  
for (int i=0; i<3; i++) {  
    cout << get<i>(t1) << '\n';  
}  
*/
```

tuple

STL

- `get<>` 사이에 변수를 넣을 수는 없다

tie

tie

STL

```
auto t = make_tuple(10, 20, 30);
```

```
int x = get<0>(t); 10
```

```
int y = get<1>(t); 20
```

```
int z = get<2>(t); 30
```

```
cout << x << ' ' << y << ' ' << z << '\n';
```

```
x = y = z = 0;
```

```
tie(x,y,z) = t;
```

```
cout << x << ' ' << y << ' ' << z << '\n';
```

$\text{tie}(x, y, z) = t$

10, 20, 30

$\text{tie}(a, b) = \text{make_pair}(b, a);$

$\text{tie}(x, y) = t$

tie

STL

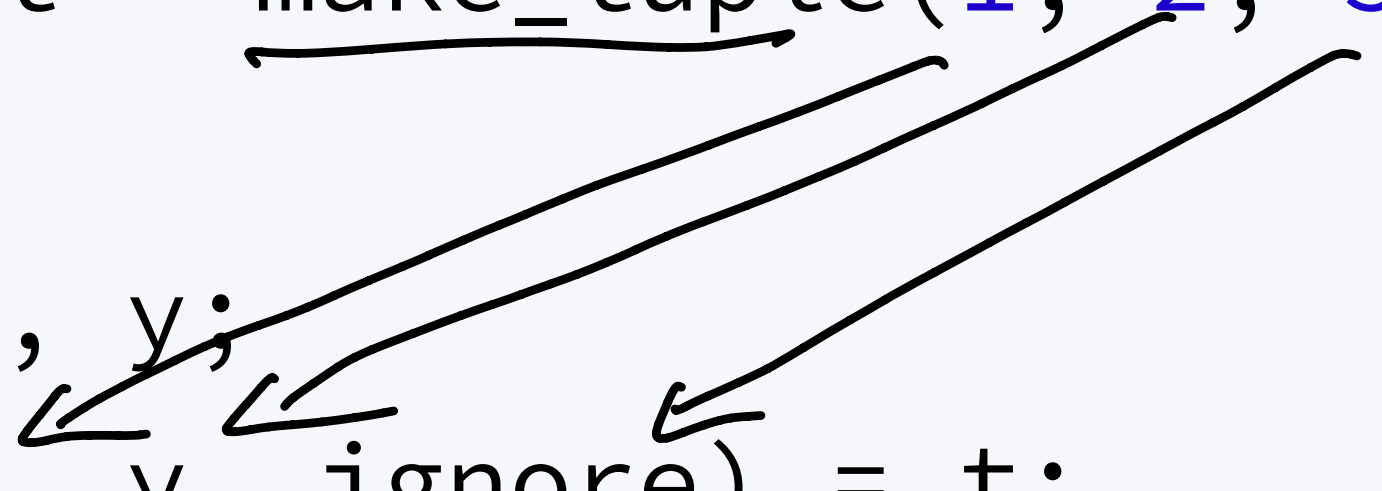
using namespace std;

34

- tie는 pair에도 사용할 수 있다
- 변수값을 무시해야 하는 경우에는 ignore를 사용한다

```
auto t = make_tuple(1, 2, 3);
```

```
int x, y;  
tie(x, y, ignore) = t;
```



```
cout << x << ' ' << y << '\n';
```

vector

길이와 변하는 배열

Linked List

[]

vector

STL

- vector는 배열이다
- 길이를 변경할 수 있는 배열이다
- `#include <vector>`

vector

STL

```
#include <iostream>
```

```
#include <vector>
```

```
using namespace std;
```

```
int main() {
```

```
    vector<int> v1; 크기가 0
```

```
    vector<int> v2(10); 크기가 10
```

```
    vector<int> v3(15, 1); 크기가 15, 초기값 1
```

```
    vector<int> v4 = {1, 2, 3, 4, 5};
```

```
    return 0;
```

```
}
```

C++11
↓
Initializer List

vector

STL

```
#include <iostream>
#include <vector>
using namespace std;
```

```
int main() {
```

```
    vector<pair<int,int>> v5;
```

```
    vector<pair<int,int>> v6 = {{1, 2}, {3, 4}};
```

```
    vector<vector<int>> v7;
```

```
    int n = 10, m = 20;
```

```
    vector<vector<int>> v8(n, vector<int>(m));
```

```
    return 0;
```

```
}
```

Vector<int> a(10)

// a[10]

○

[[1, 2], [3, 4]]
v6[0] v6[1]

$n \times m$

v8[n][m]

vector

39

STL

```
vector<int> a = {1, 2, 3, 4, 5};
```

```
a.push_back(6); // [1, 2, 3, 4, 5, 6]
```

```
a.push_back(7); // [1, 2, 3, 4, 5, 6, 7]
```

```
a.pop_back(); // [1, 2, 3, 4, 5, 6]
```

```
a.pop_back(); // [1, 2, 3, 4, 5]
```

```
a.pop_back(); // [1, 2, 3, 4]
```

```
a.clear(); // []
```

```
a.resize(5); // [0, 0, 0, 0, 0]
```

vector

40

STL

```
a.clear(); // []
```

```
a.push_back(1); // [1]
```

```
a.push_back(2); // [1, 2]
```

```
a.resize(5); // [1, 2, 0, 0, 0]
```

```
a.resize(3); // [1, 2, 0]
```

```
a.clear(); // []
```


vector

STL

```
vector<int> a = {1, 2, 3, 4};
```

```
cout << "size = " << a.size() << '\n';
```

 4

```
a.push_back(5);
```

```
cout << "size = " << a.size() << '\n';
```

 5

```
cout << "empty = " << a.empty() << '\n';
```

 false

```
a.clear();
```

```
cout << "size = " << a.size() << '\n';
```

```
cout << "empty = " << a.empty() << '\n';
```

 true

vector

STL

```
vector<int> a = {1, 2, 3};  
cout << "front = " << a.front() << '\n';  
cout << "a[1] = " << a[1] << '\n';  
cout << "back = " << a.back() << '\n';
```

```
a.push_back(4);
```

```
for (int i=0; i<a.size(); i++) {  
    cout << a[i] << ' ';  
}  
cout << '\n';
```

a.at(1)
a[1]

vector

STL

```
front = 1
```

```
a[1] = 2
```

```
back = 3
```

```
1 2 3 4
```

vector

STL

```
vector<int> a = {1, 2, 3, 4, 5};
```

```
for (int i=0; i<a.size(); i++) {
    cout << a[i] << ' ';
}
```

```
cout << '\n';
```

Const

```
for (const int &x : a) {
    cout << x << ' ';
}
```

```
cout << '\n';
```

(2 3 4 5

1 2 3 4 5
↑ ↑ ↑ ↑ ↑
x x x x x

++i

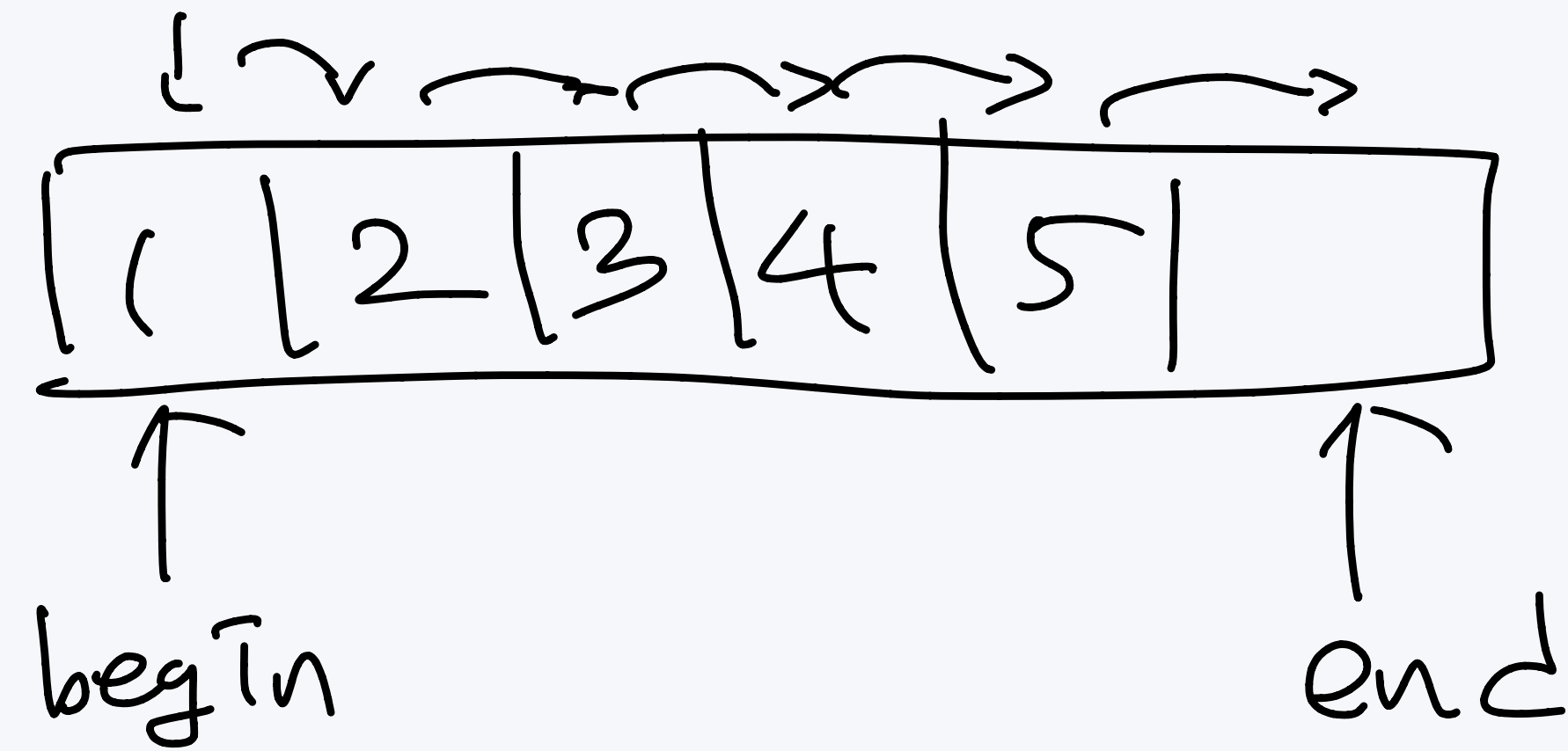
foreach

Range based for

vector

STL

```
vector<int> a = {1, 2, 3, 4, 5};
```



45

```
for (vector<int>::iterator it = a.begin(); it != a.end(); ++it) {  
    cout << *it << ' ' ;  
}
```

1 2 3 4 5

```
cout << '\n';
```

```
for (auto it = a.begin(); it != a.end(); ++it) {  
    cout << "a[" << (it - a.begin()) << "] = " << *it << '\n';  
}
```

auto a, b;

vector

46

STL

```
vector<pair<int,int>> a;  
a.emplace_back(1, 2);  
a.push_back({3, 4});  
a.push_back(make_pair(5,6));
```

```
for (auto &x : a) {  
    cout << x.first << ' ' << x.second << '\n';  
}
```

(x).first

```
for (auto it = a.begin(); it != a.end(); ++it) {  
    cout << it->first << ' ' << it->second << '\n';  
}
```

vector

STL

```
vector<int> a = {1, 2, 3};
```

```
print(a);
```

```
auto it = a.begin();
```

```
a.insert(it, 4); print(a);  $O(N)$ 
```

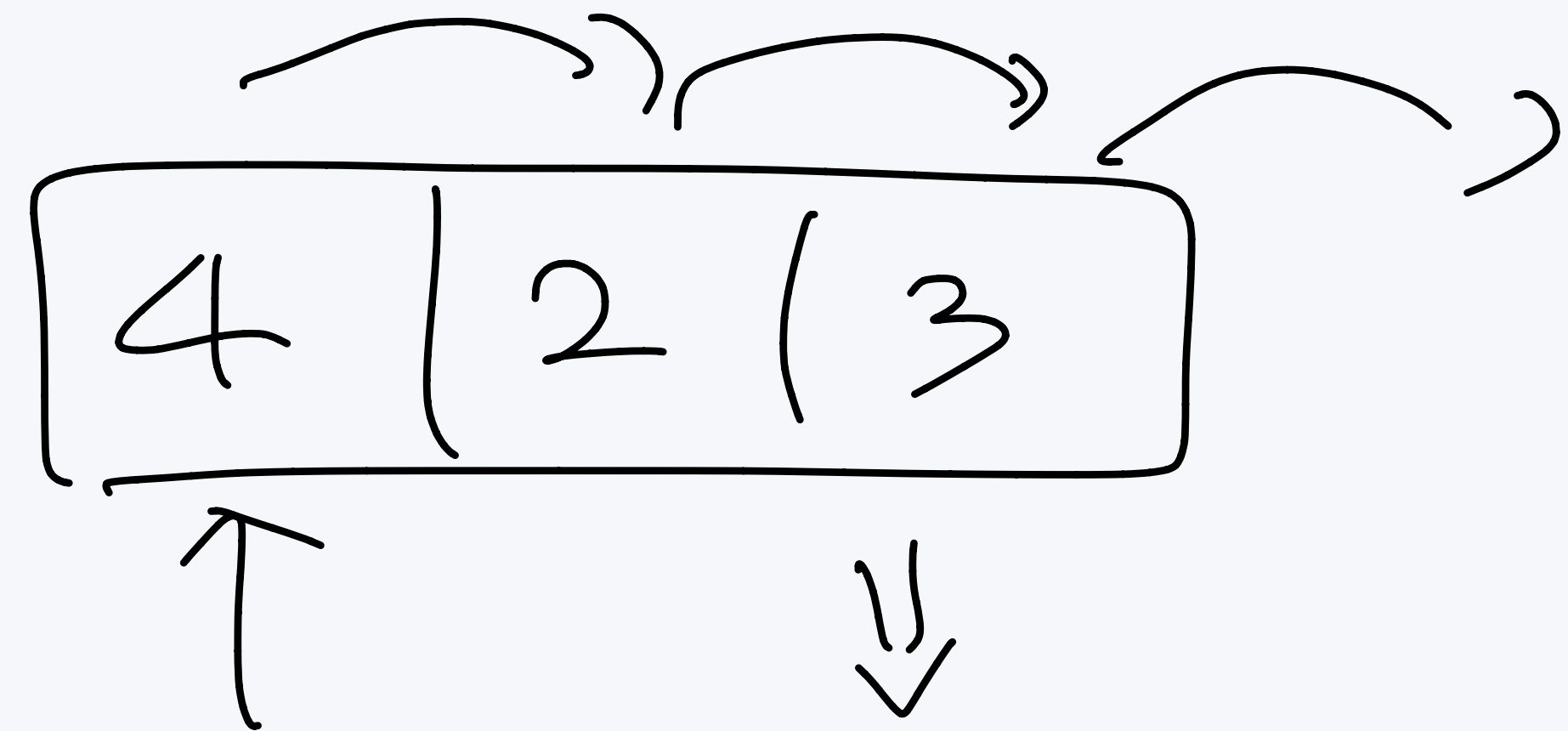
```
it = a.begin() + 1;
```

```
a.insert(it, 5, 0); print(a);
```

```
it = a.begin() + 2;
```

```
vector<int> b = {10, 20};
```

```
a.insert(it, b.begin(), b.end()); print(a);
```



begin

4 1 2 3



4 0 0 0 0 0 1 2 3

↑ ↑ ↑
b b+1 b+2

$O(5 \times \frac{N}{2})$ $O(\frac{N^2}{2})$

$10, 20$ $O(\frac{N}{2})$ $O(\frac{N^2}{2})$

4 0 10 20 0 0 0 0

vector

48

STL

```
void print(vector<int> &a) {  
    for (int x : a) {  
        cout << x << ' ';  
    }  
    cout << '\n';  
}
```


vector

STL

1 2 3

4 1 2 3

4 0 0 0 0 0 1 2 3

4 0 10 20 0 0 0 0 1 2 3

vector

STL

```
vector<int> a = {1, 2, 3, 4, 5};
```

```
print(a);
```

```
a.erase(a.begin()+2);
```

```
print(a);
```

```
a.erase(a.begin()+1, a.begin()+3);
```

```
print(a);
```

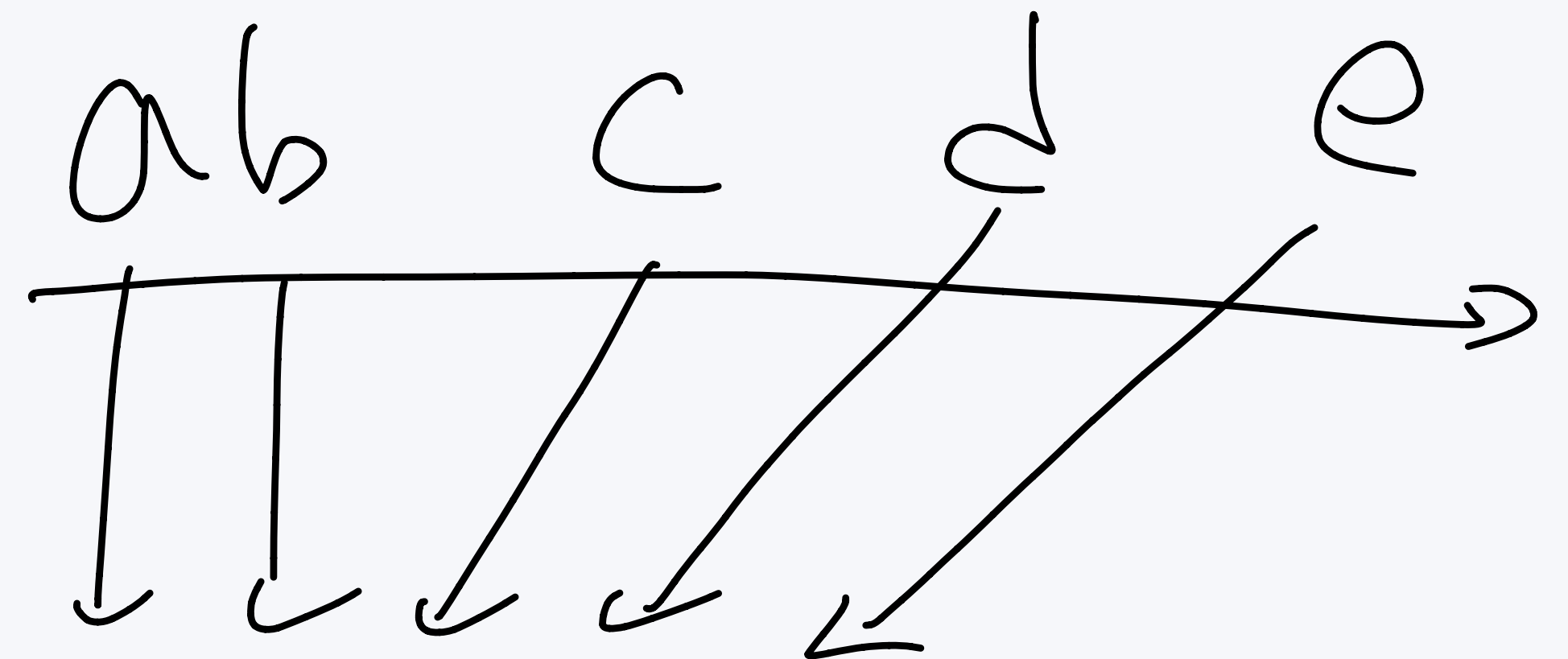
1 2 3 4 5

1 2 4 5

1 5

$\Theta(N)$

{ 2 4 5 }



Double Ended Queue.

push_front

pop_back



pop_front

push_back

deque

deque

STL

52

```
deque<int> d;
```

```
d.push_back(1); print(d);
```

```
d.push_front(2); print(d);
```

```
d.push_back(3); print(d);
```

```
d.pop_back(); print(d);
```

```
d.pop_front(); print(d);
```

1
2 1
2 1 3
2 1
↑
/

deque

STL

1

2 1

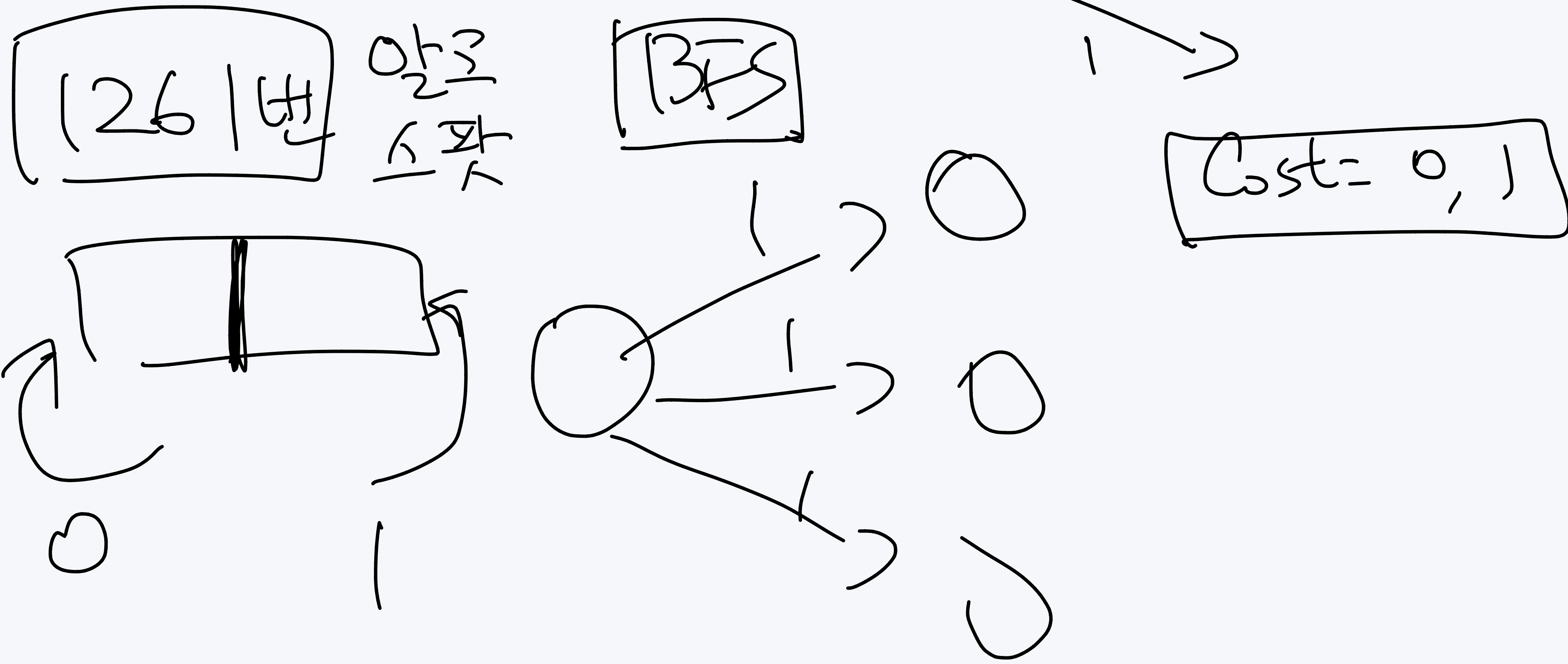
2 1 3

2 1

1

<https://www.acmicpc.net/problem/10866>

- <https://gist.github.com/Baekjoon/3e348d29fa84fc256782>



오/지 연결 리스트

vector의 차이점

list

	vector	list
insert erase	$O(N)$	$O(1)$
[1]	0	X
		Merge Sort

list

<functional>

56

STL

```
list<int> l = {2, 1, -5, 4, -3, 6, -7}; print(l);
```

```
l.sort(); print(l);
```

-7 -5 -3 1 2 4 6

```
l.sort(greater<int>()); print(l);
```

6 4 2 1 -3 -5 -7

```
l.sort([](int &u, int &v) {  
    return abs(u) < abs(v);  
});  
print(l);
```

↖ 0 1 2 3 4 5 6 7, 6 4 2 1 3 5 7

```
l.reverse(); print(l);
```


list

STL

2 1 -5 4 -3 6 -7

-7 -5 -3 1 2 4 6

6 4 2 1 -3 -5 -7

1 2 -3 4 -5 6 -7

-7 6 -5 4 -3 2 1

풍선 터뜨리기

58

<https://www.acmicpc.net/problem/2346>

- <https://gist.github.com/Baekjoon/34f32467072792589cc1>
- <https://gist.github.com/Baekjoon/d4483e012a28248ed5cd>

에디터

<https://www.acmicpc.net/problem/1406>

- <https://gist.github.com/Baekjoon/a2028fb9878c7bf82e35>



set

BST

Binary
Search
Tree

$\left. \begin{array}{l} \text{Aho} \\ \text{u} \end{array} \right\} \lg N$
 $\left. \begin{array}{l} \text{Aho} \\ \text{u} \end{array} \right\} \lg N$
 $\left. \begin{array}{l} \text{Aho} \\ \text{u} \end{array} \right\} \lg N$

set

STL

61

중복 X

```
set<int> s1; 0
```

```
set<int> s2 = {1, 2, 3, 4, 5}; 5
```

```
set<int> s3 = {1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 3, 3, 3}; 3
```

```
cout << "s1.size() = " << s1.size() << '\n';
```

```
cout << "s2.size() = " << s2.size() << '\n';
```

```
cout << "s3.size() = " << s3.size() << '\n';
```

```
set<int, greater<int>> s4;
```

<

set

STL

set: 중복 X

62

```
set<int> s;  
s.insert(1); s.insert(3); s.insert(2);
```

```
cout << "s.size() = " << s.size() << '\n';
```

삽입한 결과

↓ 성공

```
pair<set<int>::iterator, bool> result = s.insert(4);  
cout << "result iterator = " << (*result.first) << '\n'; 4  
cout << "result bool = " << result.second << '\n'; true
```

```
auto result2 = s.insert(3);  
cout << "result2 iterator = " << *result2.first << '\n';  
cout << "result2 bool = " << result2.second << '\n';  
false
```

set

STL

```
s.size() = 3
```

```
result iterator = 4
```

```
result bool = 1
```

```
result2 iterator = 3
```

```
result2 bool = 0
```

set

STL

```
set<int> s = {1, 2, 3, 4, 5};
s.erase(s.begin()); (lgN)
cout << "s.size() = " << s.size() << '\n';
```

~~begin + 1~~

연산자

```
auto it = s.begin();
++it; O(lgN)
```

S = {2, 3, 4, 5}

```
cout << "*it = " << *it << '\n';
```

```
it = s.erase(it);
```

```
cout << "*it = " << *it << '\n';
```

```
cout << "s.size() = " << s.size() << '\n';
```

{ 2, 3, 4, 5 }

{ 2, 4, 5 }

set

STL

```
s.size() = 4
```

```
*it = 3
```

```
*it = 4
```

```
s.size() = 3
```

set

STL

66

$N \geq 1$

```
set<int> s = {5, 2, 4, 1, 3, 7, 6};
```

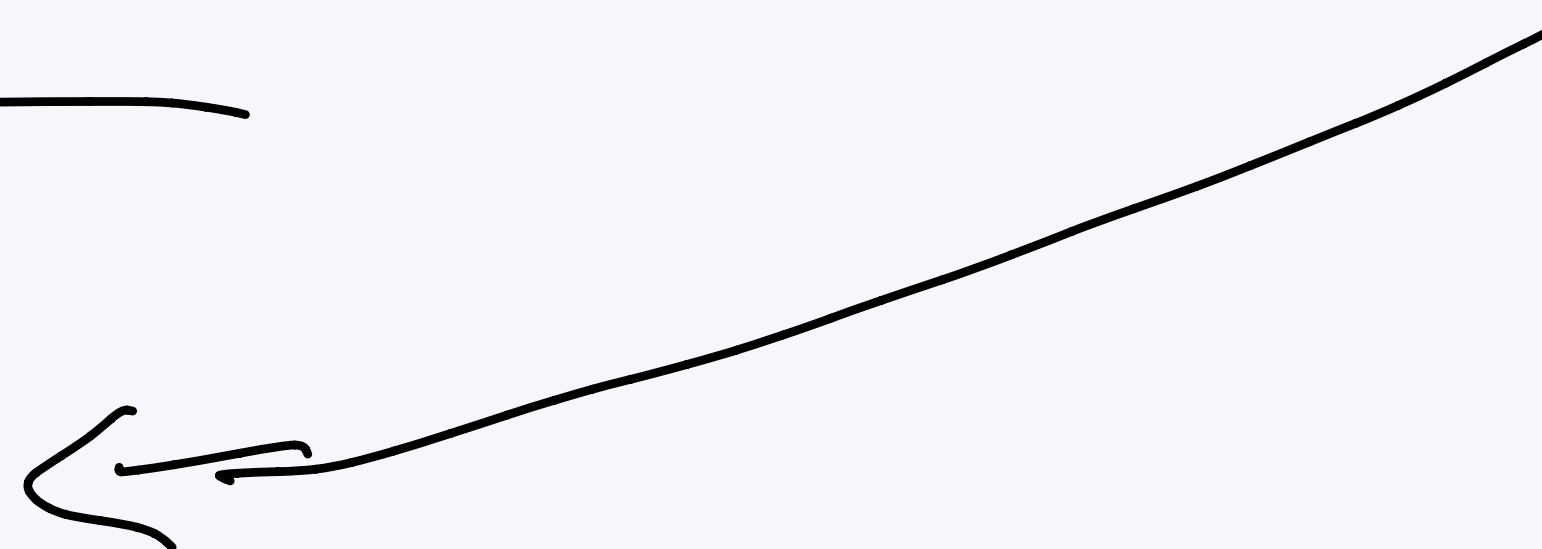
```
for (auto it = s.begin(); it != s.end(); ++it) {  
    cout << *it << ' '  
}  
cout << '\n';
```

```
for (auto x : s) {  
    cout << x << ' '  
}  
cout << '\n';
```

$(\lg N)$

$O(N \lg N)$

1 2 3 4 5 6 7



중복 빼고 정렬하기

67

<https://www.acmicpc.net/problem/10867>

- <https://gist.github.com/Baekjoon/7b42ea52c76c026bbbf9>

1 1 2 3 4 4 5

1 2 3 4 5

set

STL

68

```
set<int> s = {7, 5, 3, 1};
```

```
auto it = s.find(1);  
print(s, it);
```

1st Iterator

```
it = s.find(2);  
print(s, it);
```

s.end()

```
it = s.find(3);  
print(s, it);
```

3

```
it = s.find(4);  
print(s, it);
```

s.end()

set

STL

69

```
void print(set<int> &s, set<int>::iterator it) {  
    if (it == s.end()) {  
        cout << "end\n";  
    } else {  
        cout << *it << '\n';  
    }  
}
```

set

STL

Set

0

X

70

```
set<int> s = {7, 1, 5, 3};
```

```
for (int i=1; i<=9; i++) {  
    cout << "s.count(" << i << ") = " << s.count(i) << '\n';  
}
```

1, 0

```
if (s.count(i) > 0) {  
    s.find(i) != s.end()  
}
```

숫자 카드

<https://www.acmicpc.net/problem/10815>

- <https://gist.github.com/Baekjoon/39dc82f84e81ddc2b905>

숫자 카드 2

72

<https://www.acmicpc.net/problem/10816>

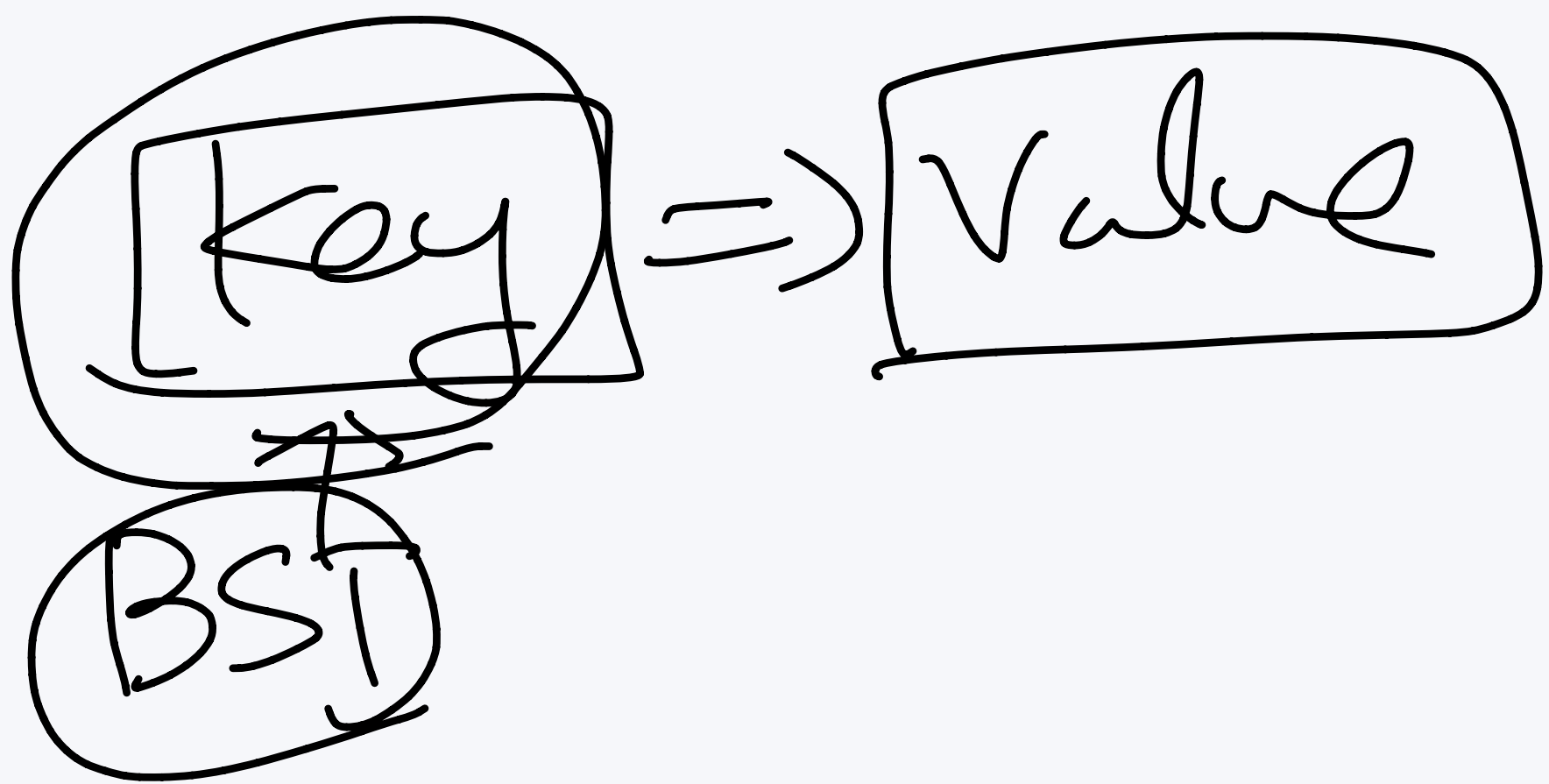
- <https://gist.github.com/Baekjoon/7a9151ba2beefaaf5944>

multiset

Set + 정렬

map

set



hypo₂

a[0] = 3
↑ ↑
key value

map key 자크트
STL ↙ val 자크트

$$d2[1] = 2$$

$$d2[3] = 4$$

$$d2[5] = 6$$

```
map<int, int> d1;
```

```
map<int, int> d2 = {{1, 2}, {3, 4}, {5, 6}};
```

```
cout << "d1.size() = " << d1.size() << '\n'; 0
```

```
cout << "d2.size() = " << d2.size() << '\n'; 3
```

```
d1[10] = 20;
```

```
cout << "d1[10] = " << d1[10] << '\n';
```

```
cout << "d2[1] = " << d2[1] << '\n'; 2
```

```
cout << "d2[2] = " << d2[2] << '\n'; 0
```

```
cout << "d2[3] = " << d2[3] << '\n'; 4
```

```
cout << "d2[4] = " << d2[4] << '\n'; 0
```

```
cout << "d2[5] = " << d2[5] << '\n'; 6
```

map

STL

```
map<int, int> d1;
```

```
map<int, int> d2;
```

```
for (int i=1; i<=9; i+=2) {
    d1[i] = i*i;
    d2[i] = i*i;
}
```

```
cout << "d1.size() = " << d1.size() << '\n'; 5
cout << "d2.size() = " << d2.size() << '\n'; 5
```

$$d[1] = 1$$

$$d[3] = 9$$

$$d[5] = 25$$

$$d[7] = 49$$

$$d[9] = 81$$

map

STL

```
for (int i=1; i<=10; i++) {  
    if (d1[i]) { 0 0 | 0 1 0 1 0 1  
        cout << i << ' ';    1 3 5 7 9  
    }  
}
```

```
cout << '\n';  
for (int i=1; i<=10; i++) {  
    if (d2.count(i)) {  
        cout << i << ' ';    1 3 5 7 9  
    }  
}
```

```
cout << '\n';  
cout << "d1.size() = " << d1.size() << '\n';  
cout << "d2.size() = " << d2.size() << '\n';
```

$d1[2] = 0$

$d1$ 3 7 1 10
 $d2$ 3 7 1 5

$O(\lg N)$

map

STL

```
d1.size() = 5
```

```
d2.size() = 5
```

```
1 3 5 7 9
```

```
1 3 5 7 9
```

```
d1.size() = 10
```

```
d2.size() = 5
```

map

STL

```
map<int, int> d = {{1, 2}, {3, 4}, {5, 6}};
```

```
for (auto it = d.begin(); it != d.end(); ++it) {
    cout << (it->first) << ' ' << (it->second) << '\n';
}
```

1 2

```
for (auto p : d) {
    cout << p.first << ' ' << p.second << '\n';
}
```

1 2

3 4

5 6

저항

<https://www.acmicpc.net/problem/1076>

- <https://gist.github.com/Baekjoon/9a5d86db0452703ec384>

등보잡

<https://www.acmicpc.net/problem/1764>

- <https://gist.github.com/Baekjoon/d6ee3fbcda9555e1103b>

stack

stack

STL

```
stack<int> s1;
```

```
stack<int, list<int>> s2;
```

```
deque<int> d = {1, 2, 3, 4, 5};  
stack<int> s3(d);
```

stack

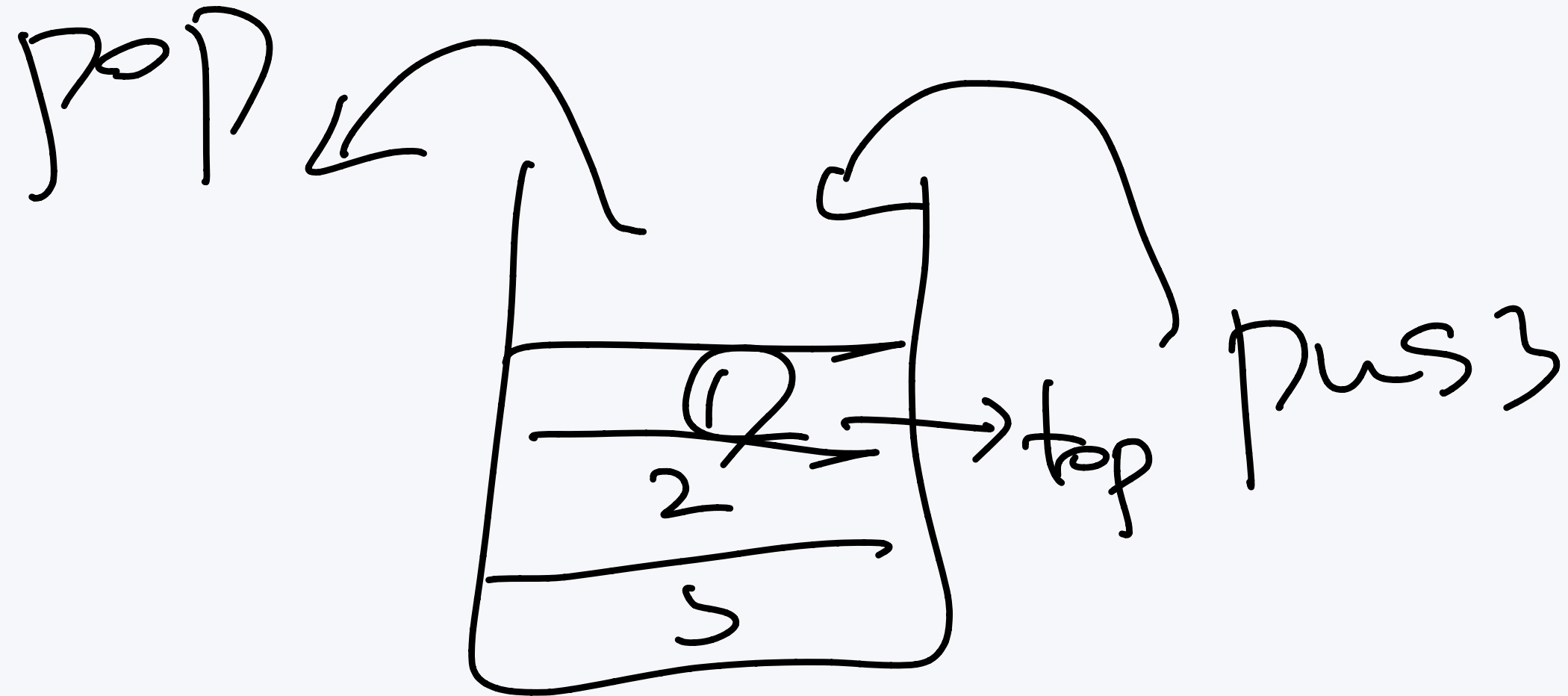
STL

```
stack<int> s;
```

```
for (int i=1; i<=5; i++) {  
    s.push(i);  
}
```

```
for (int i=0; i<2; i++) {  
    cout << s.top() << '\n';  
    s.pop();  
}
```

```
cout << "size = " << s.size() << '\n';
```



stack

STL

```
for (int i=10; i>=6; i--) {  
    s.push(i);  
}
```

```
cout << "size = " << s.size() << '\n';  
cout << "empty = " << s.empty() << '\n';
```

```
while (!s.empty()) {  
    cout << s.top() << '\n';  
    s.pop();  
}
```

```
cout << "size = " << s.size() << '\n';  
cout << "empty = " << s.empty() << '\n';
```

stack

STL

```
stack<pair<int,int>> s;  
s.push(make_pair(1,2));  
s.push({3, 4});  
s.emplace(5, 6);
```

```
while (!s.empty()) {  
    auto p = s.top();  
    cout << p.first << ' ' << p.second << '\n';  
    s.pop();  
}
```

<https://www.acmicpc.net/problem/10828>

- <https://gist.github.com/Baekjoon/1f45c9069e527209fdc0>

queue

queue

STL

88

```
queue<int> q1;
```

```
queue<int, list<int>> q2;
```

```
deque<int> d = {1, 2, 3, 4, 5};  
queue<int> q3(d);
```


queue

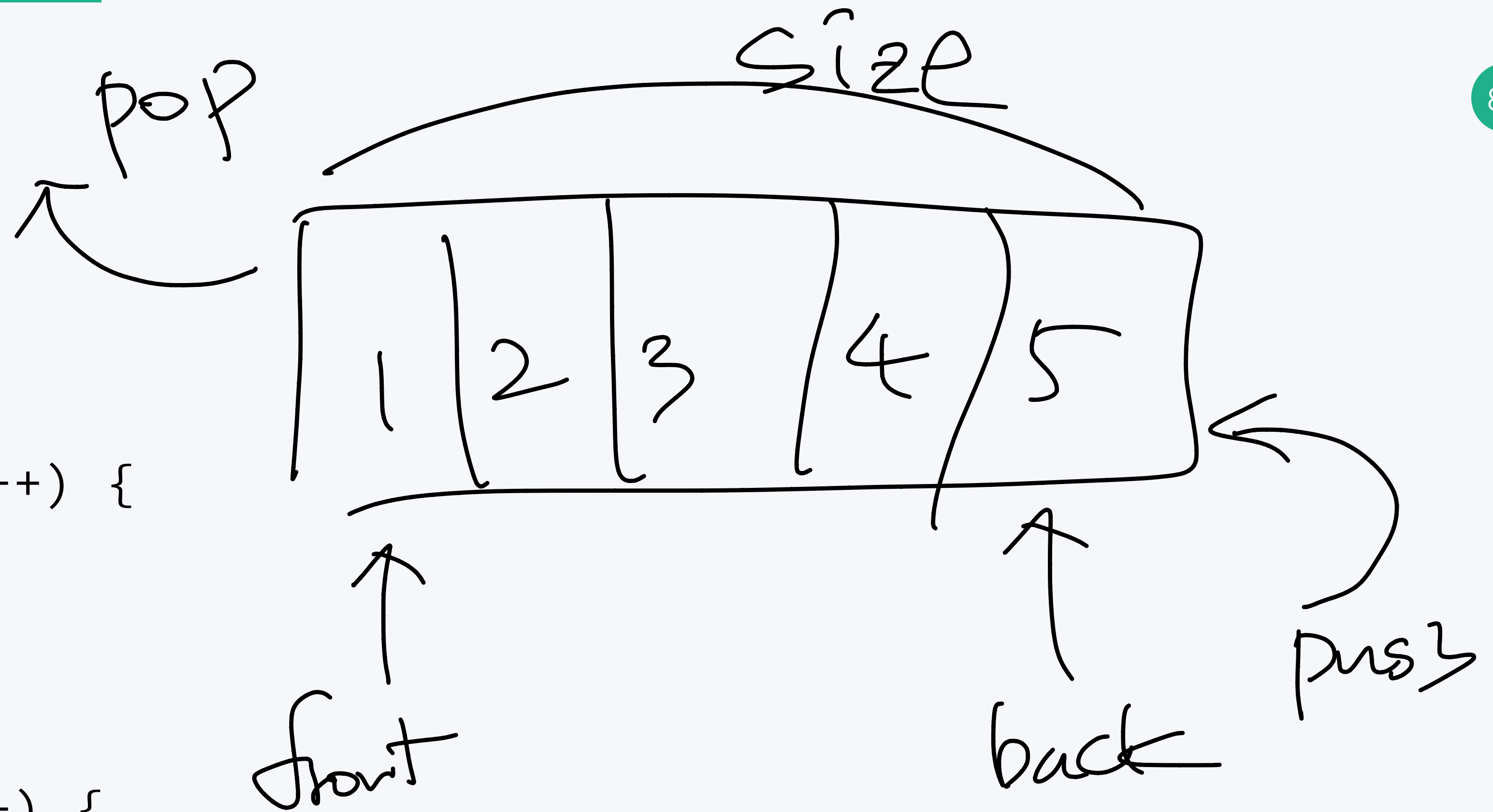
STL

```
queue<int> q;
```

```
for (int i=1; i<=5; i++) {  
    q.push(i);  
}
```

```
for (int i=0; i<2; i++) {  
    cout << q.front() << ' ' << q.back() << '\n';  
    q.pop();  
}
```

```
cout << "size = " << q.size() << '\n';  
cout << "empty = " << q.empty() << '\n';
```



queue

STL

```
for (int i=6; i<=10; i++) {  
    q.push(i);  
    cout << "back = " << q.back() << '\n';  
}
```

```
while (!q.empty()) {  
    cout << q.front() << ' ' << q.back() << '\n';  
    q.pop();  
}
```

```
cout << "size = " << q.size() << '\n';  
cout << "empty = " << q.empty() << '\n';
```

queue

STL

```
queue<pair<int,int>> q;  
q.push(make_pair(1,2));  
q.push({3,4});  
q.emplace(5,6);
```

```
while (!q.empty()) {  
    auto p = q.front();  
    cout << p.first << ' ' << p.second << '\n';  
    q.pop();  
}
```



<https://www.acmicpc.net/problem/10845>

- <https://gist.github.com/Baekjoon/275f19126d5d0f54641f>

조세퍼스 문제

<https://www.acmicpc.net/problem/1158>

- <https://gist.github.com/Baekjoon/8b4b4a815349c97b369d>

priority_queue

Set

Ex: 1, 2, 3, 4, 5

BST

Pq

Ex: 1, 2, 3, 4, 5

Heap.

priority_queue

STL

```
vector<int> a = {5, 2, 4, 1, 3};
```

```
priority_queue<int> q1;
```

```
for (int x : a) {  
    q1.push(x);  
}
```

```
while (!q1.empty()) {  
    cout << q1.top() << ' '  
    q1.pop();  
}  
cout << '\n';
```

Queue (큐)

Pop (빼기)

5 2 4 1 3

5 4 3 2 1

priority_queue

STL

```
vector<int> a = {5, 2, 4, 1, 3};
```

```
priority_queue<int> q2;
```

```
for (int x : a) {  
    q2.push(-x);  
}
```

```
while (!q2.empty()) {  
    cout << -q2.top() << ' '  
    q2.pop();  
}
```

```
cout << '\n';
```

Max Heap

Min Heap

$$\frac{a < b}{-a > -b}$$

int [-2 1 4 7 4 8 3 6 4 8]

if (x == -x)

priority_queue

STL

97

```
vector<int> a = {5, 2, 4, 1, 3};
```

```
priority_queue<int, vector<int>, greater<int>> q3;
```

```
for (int x : a) {
```

```
    q3.push(x);
```

```
}
```

```
while (!q3.empty()) {
```

```
    cout << q3.top() << ' ';
```

```
    q3.pop();
```

```
}
```

```
cout << '\n';
```

priority_queue

STL

98

```
priority_queue<int> q;
```

```
for (int x : {2, 1, 4, 3, 5}) {  
    cout << "x = " << x << '\n';  
    q.push(x);  
    cout << "top = " << q.top() << '\n';  
}
```

```
cout << "size = " << q.size() << '\n';  
cout << "empty = " << q.empty() << '\n';
```

priority_queue

STL

99

```
while (!q.empty()) {  
    cout << "top = " << q.top() << '\n';  
    q.pop();  
}
```

```
cout << "size = " << q.size() << '\n';  
cout << "empty = " << q.empty() << '\n';
```

최소 힙

100

<https://www.acmicpc.net/problem/1927>

- <https://gist.github.com/Baekjoon/a5de8034d60ad0466a24>
- <https://gist.github.com/Baekjoon/9ad24438f9124c26a461>

최소 힙

101

<https://www.acmicpc.net/problem/1927>

- <https://gist.github.com/Baekjoon/a5de8034d60ad0466a24>
- <https://gist.github.com/Baekjoon/9ad24438f9124c26a461>

string

string

STL

string s1; *이 문자열*

char c[] = "c string";

string s2(c);

string s3 = c;

c[1] = '\0'; *"\c"*

string s4(c); *"\c"*

string s5 = c; *"\c"*

string s6(10, '!');

string s7 = "abcdefg";

string

STL

c string

c string

c

c

!!!!!!!!!!

abcdefg

string

STL

```
string s1, s2;  
cin >> s1 >> s2;  
cout << s1 << ' ' << s2;
```

단어의 개수

106

<https://www.acmicpc.net/problem/1152>

- <https://gist.github.com/Baekjoon/32d83ef9853706fb2386>

문자열 분석

107

<https://www.acmicpc.net/problem/10820>

- <https://gist.github.com/Baekjoon/228b31a4f2fb84e49f0a>

정수의 개수

108

<https://www.acmicpc.net/problem/10821>

- <https://gist.github.com/Baekjoon/dcd8751c1ff39204483a>
- getline의 세 번째 인자는 구분자

string

STL

```
string s1, s2;  
cin >> s1 >> s2;  
cout << s1 << ' ' << s2;
```

string

STL

```
string s = "abc";  
printf("%s\n", s.c_str());
```

```
s += "def";  
printf("%s\n", s.c_str());
```

```
s = "";  
printf("%s\n", s.c_str());
```

```
s = "abcdefghijklmnopqrstuvwxyz";  
printf("%s\n", s.c_str());
```

Const Char *

CString 문자열 ⇒ String
생성, =

⇔

.c_str()

string

STL

```
string s = "book";  
cout << s << ": " << s.size() << '\n';  
cout << s << ": " << s.length() << '\n';
```

s = "";

```
cout << s << ": " << s.size() << '\n';  
cout << s << ": " << s.size()-1 << '\n';
```

-1 (X)

Vector list

Set

↓
unsigned

string

STL

book: 4

book: 4

: 0

: 18446744073709551615

571 0

112

for (int i=0; i < S.size()-1; i++)

O(1)

for (int i=0; i < (int) S.size()-1; i++)

int len = S.size()

len-1

단어 길이 재기

<https://www.acmicpc.net/problem/2743>

$$\sum_{i=1}^N i = N$$

113

- <https://gist.github.com/Baekjoon/ca6054e85bad2a517002>

for(int i=0;

strlen(S); i++) {

O(N)

(N²)

}

string

STL

114

```
string s = "book";  
cout << s << ": " << s.empty() << '\n';
```

```
s = "";  
cout << s << ": " << s.empty() << '\n';
```

string

STL

```
string s1 = "string";
string s2 = "stirng";

if (s1 == s2) {
    cout << "s1 == s2" << '\n';
} else if (s1 != s2) {
    cout << "s1 != s2" << '\n';
}

if (s1 < s2) {
    cout << "s1 < s2" << '\n';
} else if (s1 > s2) {
    cout << "s1 > s2" << '\n';
}
```

115

~~String~~

7267 $a == b$ $O(1)$
2452 $O(201)$

Set<String>
 $N \log N \times 201$

Trie

string

STL

```
string a = "Hello";
```

```
string b = "World";
```

```
string hello_world = a (+) " " (+) b;
```

```
hello_world += "!";
```

```
cout << hello_world << '\n';
```

~~StrCat~~

string

STL

117

```
string a = "Hello";
```

```
string b = "World";
```

```
string hello_world = a + " " + b;
```

```
hello_world.push_back('!');
```

```
cout << hello_world << "\n";
```

string

118

STL

```
string a = "He";  
a.append(2, 'l'); // "Hell"  
a.append("o").append(1, ' '); // "Hello "  
  
string b = "";  
const char *c = "World";  
b.append(c); // "World"  
  
string hello_world = a; // "Hello "  
hello_world.append(b); // "Hello World"  
hello_world.push_back('!'); // "Hello World!"  
  
cout << hello_world << '\n';
```

string

119

STL

```
string s = "e"; // "e"
```

```
s.insert(0, "H"); // "He"
```

```
s.insert(2, "o"); // "Heo"
```

```
s.insert(2, 2, 'l').append(" "); // "Hello "
```

```
string world = "Half the World Away";
```

```
s.insert(6, world, 9, 5).push_back('!'); // "Hello World!"
```

string

STL

```
string str = "10";  
int number = stoi(str); 10  
print(str, number);
```

```
number = stoi(str, 0, 2);  
print(str, number); 2
```

```
str = "fffff";  
number = stoi(str, 0, 16);  
print(str, number);
```

String
to
int

string

STL

```
str = "21 Guns";  
number = stoi(str); 21  
print(str, number);
```

```
str = "3.141592";  
number = stod(str); 3  
print(str, number);
```

Stoll long long
Stof float
Stod double

string

STL

```
/*  
str = "2147483648";  
number = stoi(str);  
print(str, number);
```

```
str = "hello";  
number = stoi(str);  
print(str, number);
```

```
*/
```

int 변환 2147483648)

더하기

123

<https://www.acmicpc.net/problem/10822>

- <https://gist.github.com/Baekjoon/c5a8983b405951c9e4a2>

더하기 2

124

<https://www.acmicpc.net/problem/10823>

- <https://gist.github.com/Baekjoon/308a504b631264363a20>
- string을 표준 입출력 처럼 사용하려면 istream 을 사용한다

string

STL

125

- unsigned long: `stoul`
- unsigned long long: `stoull`
- float: `stof`
- double: `stod`
- long double: `stold`

string

STL

```
int n1 = 1;
```

```
int n2 = 2;
```

```
string s1 = to_string(n1);
```

```
string s2 = to_string(n2);
```

```
cout << s1 + ' ' + s2 << '\n';
```

S → 1 2

1 2

string

STL

127

```
long long l1 = 2147483647;
```

```
long long l2 = 2147483647;
```

```
s1 = to_string(l1);
```

```
s2 = to_string(l2);
```

```
cout << s1 + ' ' + s2 << '\n';
```

string

STL

128

```
double d = 3.141592;
```

```
float f = 65358979.0;
```

```
s1 = to_string(d);
```

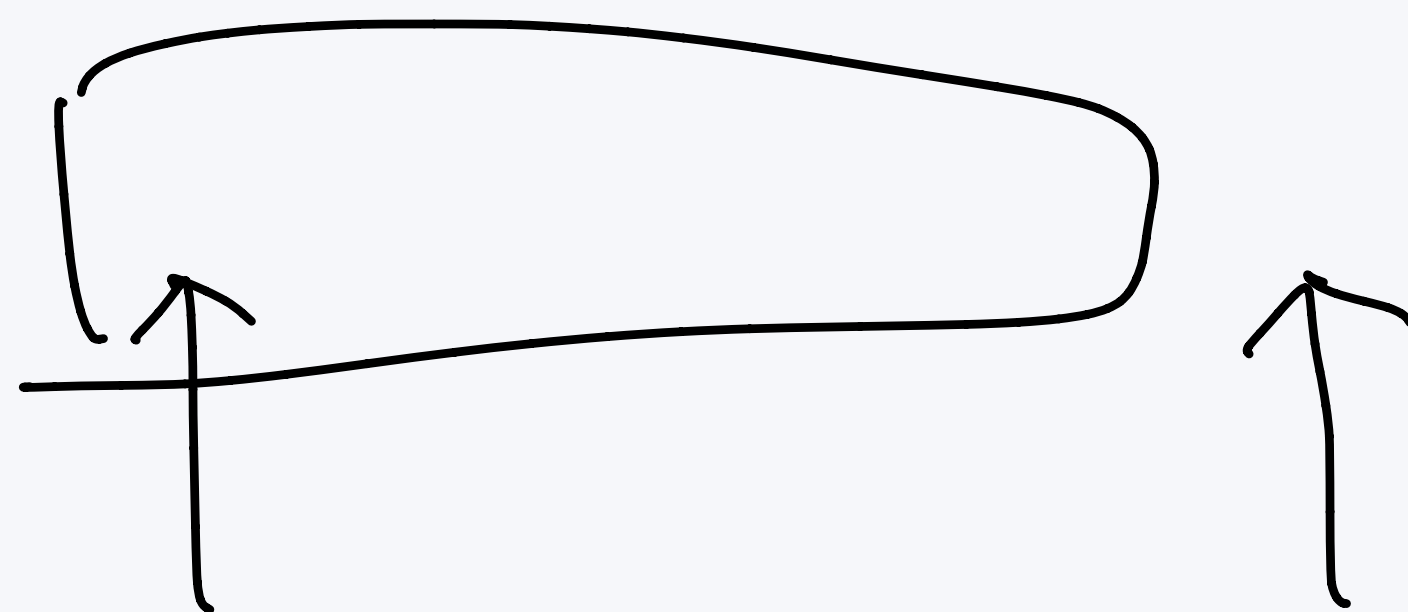
```
s2 = to_string(f);
```

```
cout << s1 + ' ' + s2 << '\n';
```


네수

<https://www.acmicpc.net/problem/10824>

- <https://gist.github.com/Baekjoon/88d1ecca80dd87b17dd2>



$[begin, end)$

↑ ↑ ↻

algorithm

Offline. startlink, help

count

STL

`count(begin, end, value)`

[begin, end)에 포함되어 있는 원소 중에서 value의 개수를 찾는다

`count_if(begin, end, (p))`

[begin, end)에 포함되어 있는 원소 중에서 조건 p에 해당하는 것의 개수를 찾는다

시간복잡도: $O(N)$

count

STL

```
vector<int> a = {1, 4, 1, 2, 4, 2, 4, 2, 3, 4, 4};  
for (int i=1; i<=5; i++) {  
    cout << i << "의 개수: " << count(a.begin(), a.end(), i);  
    cout << '\n';  
}
```

1의 개수: 2

2의 개수: 3

3의 개수: 1

4의 개수: 5

5의 개수: 0

count

STL

```
bool is_even(int x) {  
    return x % 2 == 0;  
}
```

133

```
vector<int> a = {1, 4, 1, 2, 4, 2, 4, 2, 3, 4, 4};
```

```
int even = count_if(a.begin(), a.end(), [](int x) { is_even);  
    return x % 2 == 0;  
});
```

```
int odd = count_if(a.begin(), a.end(), [](int x) {  
    return x % 2 == 1;  
});
```

```
cout << "짝수의 개수: " << even << '\n';
```

```
cout << "홀수의 개수: " << odd << '\n';
```

개수 세기

<https://www.acmicpc.net/problem/10807>

- <https://gist.github.com/Baekjoon/8a8cde8357d2b74414e0>

알파벳 개수

135

<https://www.acmicpc.net/problem/10808>

- <https://gist.github.com/Baekjoon/938052f312534091f3f4>

find

STL

136



`find(begin, end, value)`

[begin, end)에 포함되어 있는 원소 중에서 `value`의 이터레이터

`find_if(begin, end, p)`

[begin, end)에 포함되어 있는 원소 중에서 조건 `p`에 해당하는 것의 이터레이터

두 함수 모두 못 찾으면 `end`를 리턴

시간복잡도: $O(N)$

find

137

STL

```
vector<int> a = {1, 4, 1, 2, 4, 2, 4, 2, 3, 4, 4};  
for (int i=1; i<=5; i++) {  
    auto it = find(a.begin(), a.end(), i);  
    cout << i << "의 위치: ";  
    if (it == a.end()) {  
        cout << "찾을 수 없음";  
    } else {  
        cout << (it-a.begin());  
    }  
    cout << '\n';  
}
```

find

STL

1의 위치: 0

2의 위치: 3

3의 위치: 8

4의 위치: 1

5의 위치: 찾을 수 없음

find

139

STL

```
vector<int> a = {1, 4, 1, 2, 4, 2, 4, 2, 3, 4, 4};  
auto even = find_if(a.begin(), a.end(), [](int x) {  
    return x % 2 == 0;  
});  
auto odd = find_if(a.begin(), a.end(), [](int x) {  
    return x % 2 == 1;  
});  
  
cout << "첫 번째 짝수: " << (even - a.begin()) << '\n';  
cout << "첫 번째 홀수: " << (odd - a.begin()) << '\n';
```

알파벳 찾기

140

<https://www.acmicpc.net/problem/10809>

- <https://gist.github.com/Baekjoon/1bd4360bf97be15fb5a5>

fill

STL

`fill(begin, end, value)`
[begin, end)을 `value`로 채운다

시간복잡도: $O(N)$

fill

142

STL

```
vector<int> a = {1, 4, 1, 2, 4, 2, 4, 2, 3, 4, 4};  
for (int x : a) {  
    cout << x << ' ' ;  
}  
cout << '\n';  
fill(a.begin(), a.end(), 0);  
for (int x : a) {  
    cout << x << ' ' ;  
}  
cout << '\n';
```

Handwritten annotations: Above the first vector initialization, there are 11 circles, each containing a zero (0), representing the state of the vector before the fill operation. An arrow points from the first '0' to the first element '1' in the vector. Another arrow points from the last '0' to the last element '4' in the vector. In the second vector initialization, the '0' in the third parameter of the fill function is circled, and an arrow points from it to the first element '1' in the vector.

공 냥기

143

<https://www.acmicpc.net/problem/10810>

- <https://gist.github.com/Baekjoon/b1ac23fec956727f3a9b>

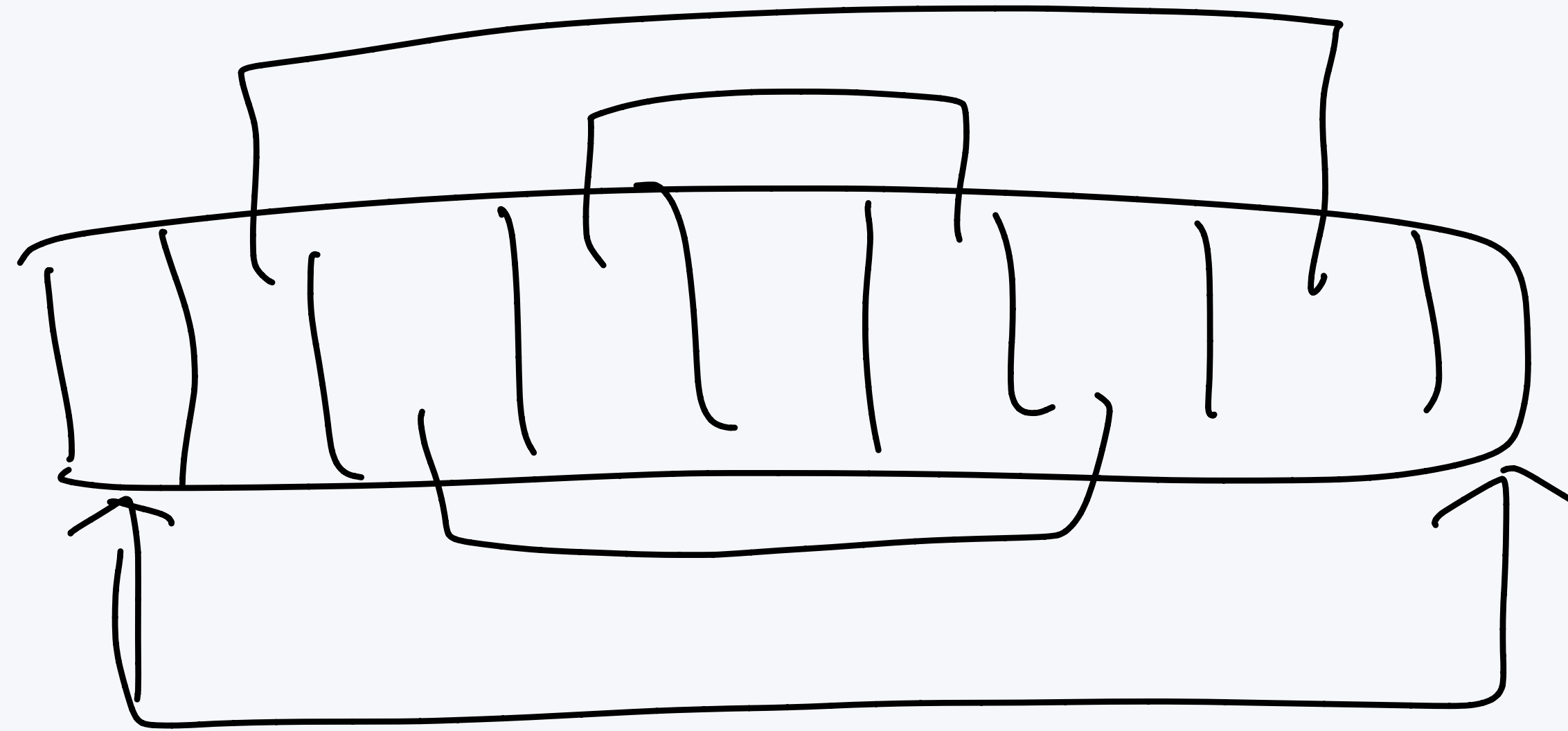
reverse

STL

`reverse(begin, end)`

[begin, end)의 순서를 역순으로 만든다

시간복잡도: $O(N)$



reverse

STL

```
vector<int> a = {1, 4, 1, 2, 4, 2, 4, 2, 3, 4, 4};  
for (int x : a) {  
    cout << x << ' ';  
}  
cout << '\n';  
reverse(a.begin(), a.end());  
for (int x : a) {  
    cout << x << ' ';  
}  
cout << '\n';
```

4 4 3 2 4 2 4 2 1 4 1

바구니 뒤집기

146

<https://www.acmicpc.net/problem/10811>

- <https://gist.github.com/Baekjoon/32f1755a421ce0be5e7f>

rotate

147

STL

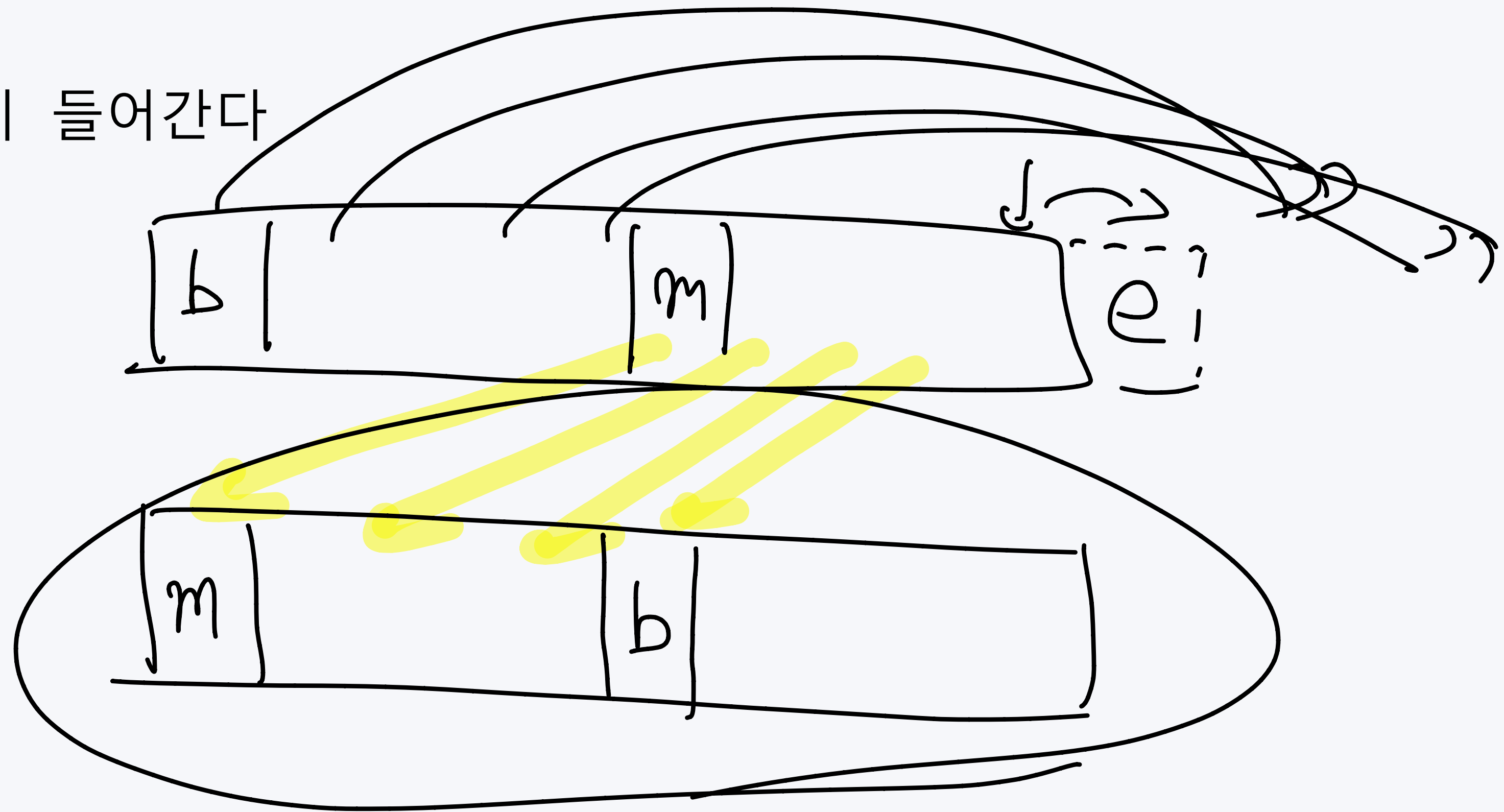
`rotate(begin, mid, end)`

[begin, end)을 mid를 기준으로 왼쪽으로 회전시킨다

begin에는 mid에 있던 값이

end-1에는 mid-1에 들어있던 값이 들어간다

시간복잡도: $O(N)$



rotate

STL

```
vector<int> a = {0, 1, 2, 3, 4, 5};
```

```
rotate(a.begin(), a.begin()+2, a.end());
```

```
for (int x : a) {
```

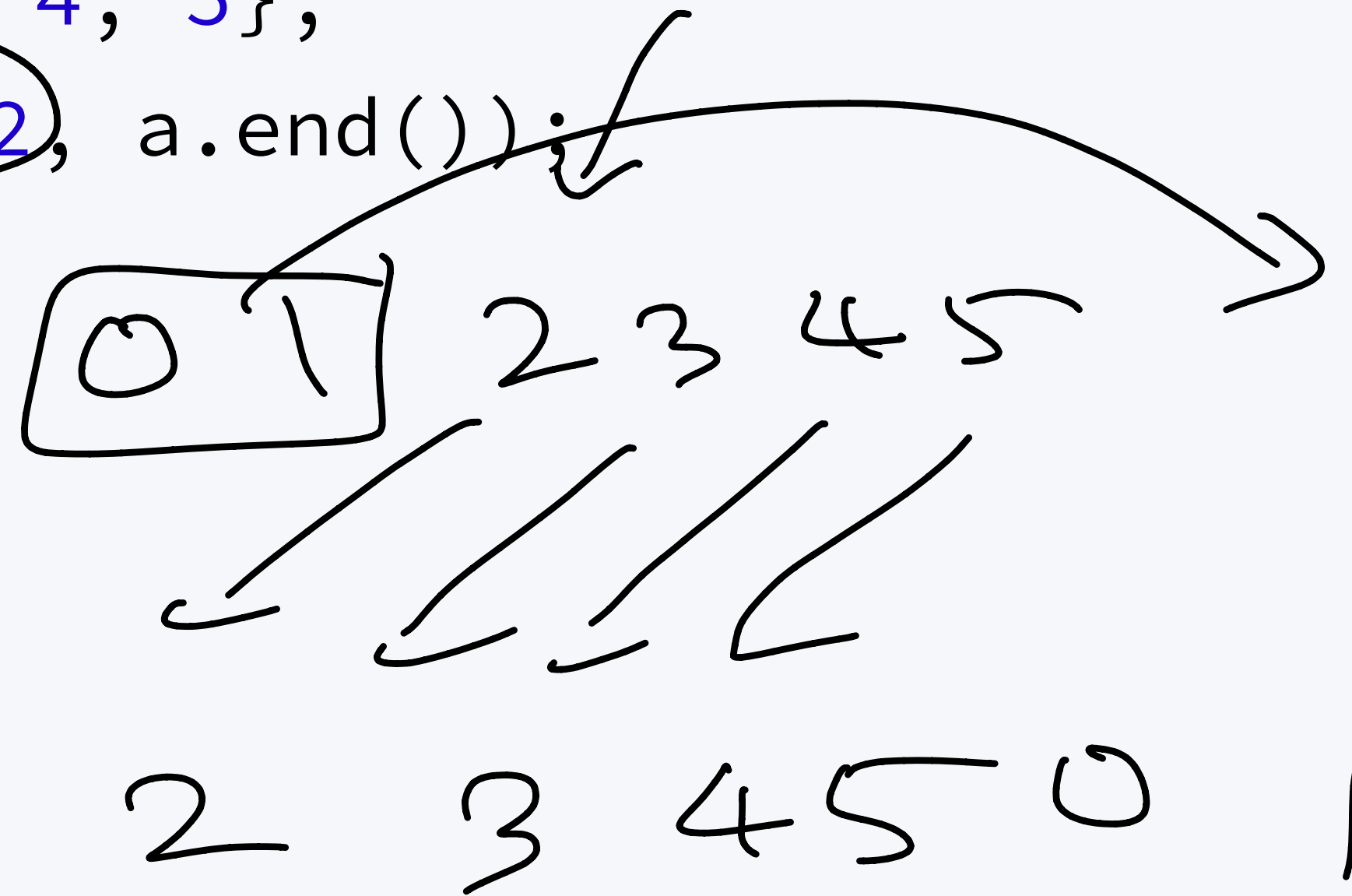
```
    cout << x << ' ';
```

```
}
```

```
cout << '\n';
```

```
0 1 2 3 4 5
```

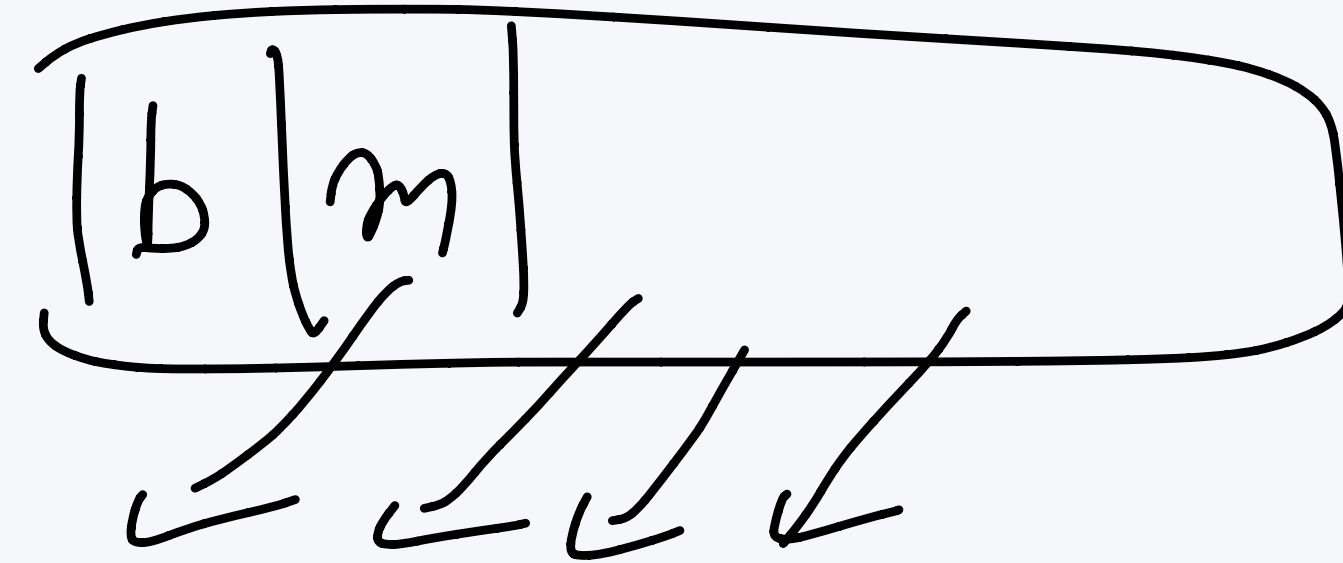
```
2 3 4 5 0 1
```



rotate

STL

```
vector<int> a = {0, 1, 2, 3, 4, 5};  
int n = a.size();  
for (int i=0; i<n; i++) {  
    rotate(a.begin(), a.begin()+1, a.end());  
    print(a);  
}
```



0 1 2 3 4 5

1 2 3 4 5 0

2 3 4 5 0 1

rotate

STL

```
vector<int> a = {0, 1, 2, 3, 4, 5};
```

```
int n = a.size();
```

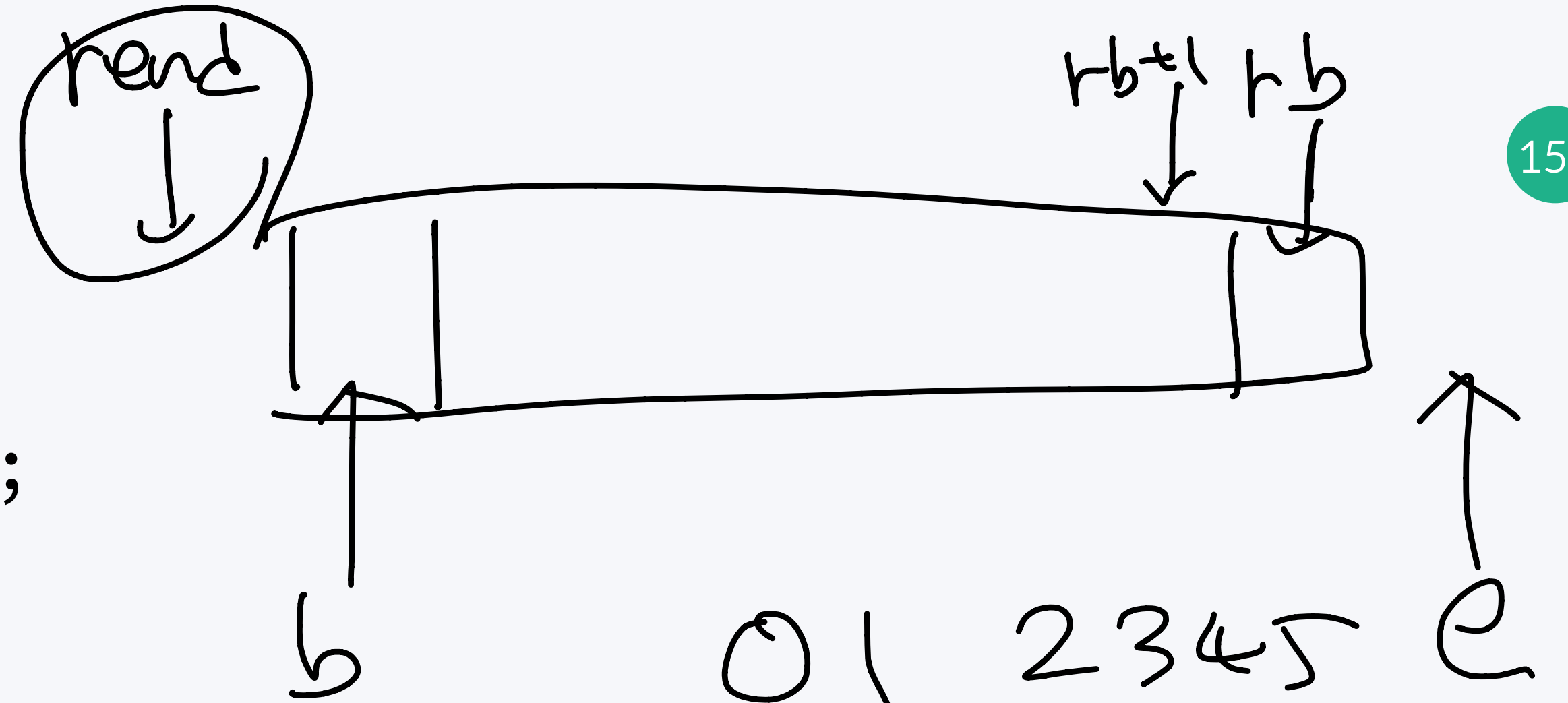
```
for (int i=0; i<n; i++) {
```

```
    //rotate(a.begin(), a.begin()+(n-1), a.end());
```

```
    rotate(a.rbegin(), a.rbegin()+1, a.rend());
```

```
    print(a);
```

```
}
```



```
for (it = a.rbegin(); it != re(); it++) {
```

바구니 순서 바꾸기

151

<https://www.acmicpc.net/problem/10812>

- <https://gist.github.com/Baekjoon/7ca7893ac05d3472b31c>

swap

STL

`swap(a, b)`

a와 b에 들어있던 값을 바꾼다

swap

STL

```
int a = 10, b = 20;  
cout << a << ' ' << b << '\n';  
swap(a,b);  
cout << a << ' ' << b << '\n';  
swap(a,b);  
cout << a << ' ' << b << '\n';
```

swap

STL

```
vector<int> a = {1, 2};  
vector<int> b = {3, 4};  
cout << a[0] << ' ' << b[0] << '\n';  
swap(a,b);  
cout << a[0] << ' ' << b[0] << '\n';  
swap(a,b);  
cout << a[0] << ' ' << b[0] << '\n';
```

공 바꾸기

155

<https://www.acmicpc.net/problem/10813>

- <https://gist.github.com/Baekjoon/3efc6fceb418062428f>

unique

STL

156

`unique(begin, end)`

`unique`는 `[begin, end)` 구간에서 연속되는 같은 값을 하나를 제외하고 제거

실제로 컨테이너의 크기를 줄이거나 제거하지 않는다
중복을 덮어쓰우거나 시프트 시키는 방식으로 작동한다

중복을 제거한 후의 `end` 이터레이터를 리턴한다

unique

STL

```
vector<int> a = {1, 1, 2, 2, 2, 3, 1, 1, 1, 2, 2, 2, 2};
for (int x : a) {
    cout << x << ' ';
}
cout << '\n';
auto last = unique(a.begin(), a.end());
for (int x : a) {
    cout << x << ' ';
}
cout << '\n';
for (auto it = a.begin(); it != last; ++it) {
    cout << *it << ' ';
}
cout << '\n';
```

Handwritten annotations: A yellow box highlights the first five elements of the vector {1, 1, 2, 2, 2}. A pink box highlights the variable 'last'. A pink arrow points from the word 'last' to the first element of the second unique group (the '3' in the original image, which is the first '1' in the provided code). A yellow highlight is under the loop body of the final range-based for loop.

unique

STL

1 1 2 2 2 3 1 1 1 2 2 2 2

1 2 3 1 2 3 1 1 1 2 2 2 2

1 2 3 1 2

unique

159

STL

```
vector<int> a = {1, 1, 2, 2, 2, 3, 1, 1, 1, 2, 2, 2, 2};
```

```
auto last = unique(a.begin(), a.end());  
a.erase(last, a.end());  
for (int x : a) {  
    cout << x << ' ';  
}  
cout << '\n';
```

1, 2, 3

1 2 3 1 2

unique

160

STL

```
vector<int> a = {1, 1, 2, 2, 2, 3, 1, 1, 1, 2, 2, 2, 2};
sort(a.begin(), a.end());
auto last = unique(a.begin(), a.end());
a.erase(last, a.end());
for (int x : a) {
    cout << x << ' ';
}
cout << '\n';
```

1 2 3

sort

STL

```
sort(begin, end)
```

[begin, end)를 <를 기준으로 정렬한다.

```
sort(begin, end, cmp)
```

[begin, end)를 cmp를 기준으로 정렬한다.

sort

STL

162



```
vector<int> a = {5, 3, 2, 1, 4};
```

```
for (int x : a) {  
    cout << x << ' '  
}
```

```
cout << '\n';
```

```
sort(a.begin(), a.end());
```

```
for (int x : a) {  
    cout << x << ' '  
}
```

```
cout << '\n';
```

수 정렬하기

<https://www.acmicpc.net/problem/2750>

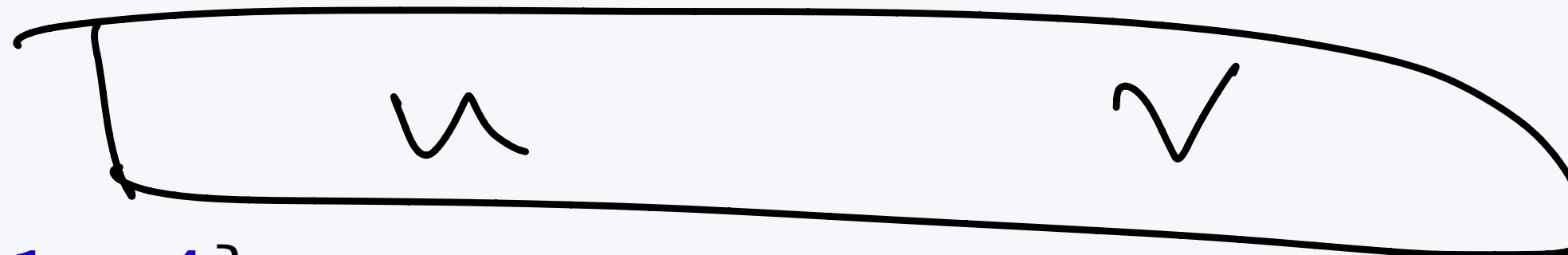
sort

STL

```
bool cmp(const int &u, const int &v) {
    return u > v;
}
```

자크 순서로 u 다음 v

올바르지 않은 true
false



```
vector<int> a = {5, 3, 2, 1, 4};
```

```
sort(a.begin(), a.end(), greater<int>());
```

```
sort(a.begin(), a.end(), cmp);
```

```
sort(a.begin(), a.end(), [](int u, int v) {
```

```
    return u > v;
```

```
});
```

Sort(a.begin(), a.end());

내림차순

단어 정렬

165

<https://www.acmicpc.net/problem/1181>

```
sort(a.begin(), a.end(), [](string u, string v) {  
    if (u.size() < v.size()) {  
        return true;  
    } else {  
        return false;  
    }  
});
```

단어 정렬

166

<https://www.acmicpc.net/problem/1181>

```
sort(a.begin(), a.end(), [](string u, string v) {  
    if (u.size() < v.size()) {  
        return true;  
    } else if (u.size() == v.size()) {  
        return u < v;  
    } else {  
        return false;  
    }  
});
```

단어 정렬

167

<https://www.acmicpc.net/problem/1181>

```
sort(a.begin(), a.end(), [](string u, string v) {  
    if (u.size() == v.size()) {  
        return u < v;  
    } else {  
        return u.size() < v.size();  
    }  
});
```

단어 정렬

168

<https://www.acmicpc.net/problem/1181>

```
sort(a.begin(), a.end(), [](string u, string v) {  
    return (u.size() < v.size()) || (u.size() == v.size() && u < v);  
});
```


단어 정렬

<https://www.acmicpc.net/problem/1181>

- <https://gist.github.com/Baekjoon/18a26d0cbbf4068c8e31>

좌표 정렬하기

<https://www.acmicpc.net/problem/11650>

- (x, y) 가 여러 개 있을 때, x 가 증가하는 순으로, 같으면 y 가 증가하는 순서로 정렬하는 문제
- `pair`를 사용하면 편한다.

```
int n;
scanf("%d", &n);
vector<pair<int, int>> a(n);
for (int i=0; i<n; i++) {
    scanf("%d %d", &a[i].first, &a[i].second);
}
sort(a.begin(), a.end());
for (int i=0; i<a.size(); i++) {
    printf("%d %d\n", a[i].first, a[i].second);
}
```

좌표 정렬하기

<https://www.acmicpc.net/problem/11650>

X 증가, 같을 Y 증가

171

- 직접 struct를 구현하는 경우에는 비교 함수를 만들어 줘야 한다.

```
struct Point {  
    int x, y;  
}  
  
bool cmp(const Point &u, const Point &v) {  
    if (u.x < v.x) {  
        return true;  
    } else if (u.x == v.x) {  
        return u.y < v.y;  
    } else {  
        return false;  
    }  
}
```

Ⓢ

Ⓥ 알

순서 0 \Rightarrow true

X \Rightarrow false

좌표 정렬하기

<https://www.acmicpc.net/problem/11650>

- cmp 함수는 u가 v의 앞에 오는 것이면 true, 아니면 false 이다.
- const와 &는 붙여야 한다.

```
bool cmp(const Point &u, const Point &v) {  
    if (u.x < v.x) {  
        return true;  
    } else if (u.x == v.x) {  
        return u.y < v.y;  
    } else {  
        return false;  
    }  
}
```

좌표 정렬하기

<https://www.acmicpc.net/problem/11650>

- 비교 함수를 만드는 경우에는 3번째 인자로 함수 이름을 넘겨줘야 한다.

```
bool cmp(const Point &u, const Point &v) {  
    if (u.x < v.x) {  
        return true;  
    } else if (u.x == v.x) {  
        return u.y < v.y;  
    } else {  
        return false;  
    }  
}  
  
sort(a.begin(), a.end(), cmp);
```

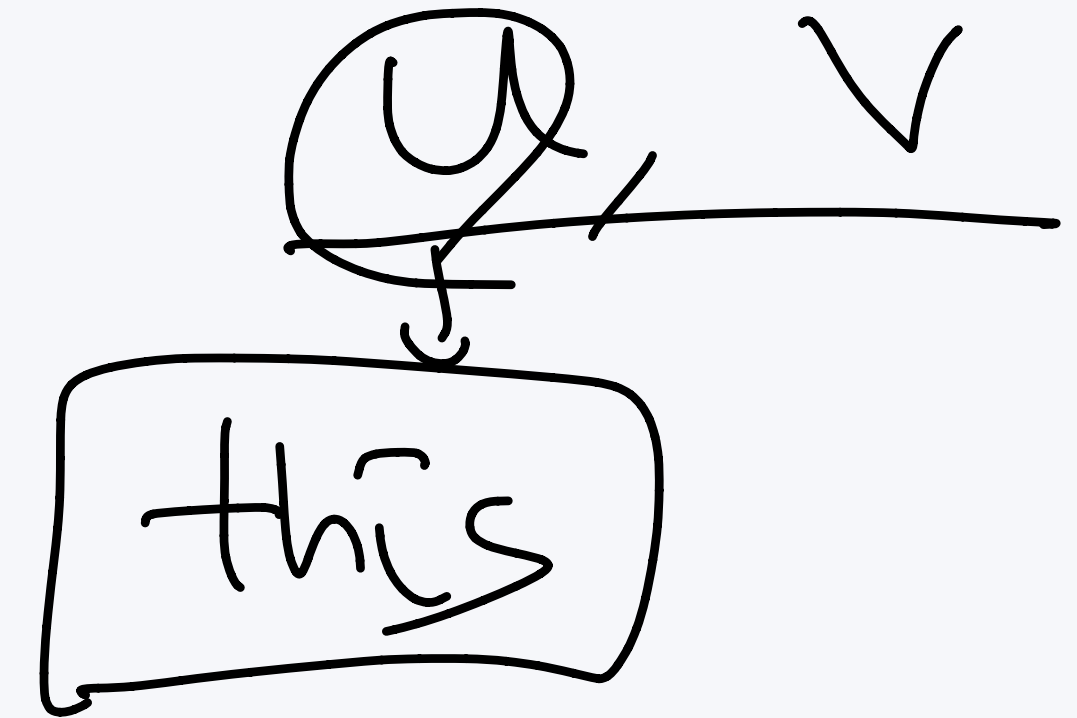
좌표 정렬하기

174

<https://www.acmicpc.net/problem/11650>

- < 연산자를 over loading 할 수도 있다. 이 경우에는 3번째 인자가 필요 없다.

```
struct Point {  
    int x, y;  
    bool operator< (const Point &v) const {  
        if (x < v.x) {  
            return true;  
        } else if (x == v.x) {  
            return y < v.y;  
        } else {  
            return false;  
        }  
    }  
}
```



좌표 정렬하기

175

<https://www.acmicpc.net/problem/11650>

- C++ (pair): <https://gist.github.com/Baekjoon/c6c920ca4fd31eb9960e>
- C++ (struct, cmp): <https://gist.github.com/Baekjoon/f3d180e438e2087a5a97>
- C++ (struct, 연산자 오버로딩): <https://gist.github.com/Baekjoon/180de19581f0b55c433c>

좌표 정렬하기 2

176

<https://www.acmicpc.net/problem/11651>

- (x, y) 가 여러 개 있을 때, y 가 증가하는 순으로, 같으면 x 가 증가하는 순서로 정렬하는 문제

좌표 정렬하기 2

177

<https://www.acmicpc.net/problem/11651>

- C++ (pair, cmp): <https://gist.github.com/Baekjoon/df02385467e2c6e36d59>
- C++ (pair, 뒤집어서 저장): <https://gist.github.com/Baekjoon/7cee7721722d6789769c>

<https://www.acmicpc.net/problem/10825>

- 도현이네 반 학생 N명의 이름과 국어, 영어, 수학 점수가 주어진다.
 - 다음과 같은 조건으로 학생의 성적을 정렬하는 문제
1. 국어 점수가 감소하는 순서로
 2. 국어 점수가 같으면 영어 점수가 증가하는 순서로
 3. 국어 점수와 영어 점수가 같으면 수학 점수가 감소하는 순서로
 4. 모든 점수가 같으면 이름이 사전 순으로 증가하는 순서로

국영수

179

<https://www.acmicpc.net/problem/10825>

N^2

- C++: <https://gist.github.com/Baekjoon/112bdd3b8d53d14d0f35>
- C++ (tuple 사용): <https://gist.github.com/Baekjoon/cfbc58f340f2043213d0>

	Q	H
Best	$N \lg N$	$N \lg N$
Worst	N^2	$N \lg N$

STL Sort

Intro Sort

Quick + Heap

depth

나이순 정렬

<https://www.acmicpc.net/problem/10814>

- 온라인 저지에 가입한 사람들의 나이와 이름이 가입한 순서대로 주어진다.
- 회원들을 나이가 증가하는 순으로, 나이가 같으면 먼저 가입한 사람이 앞에 오는 순서로 정렬하는 문제
- 가입한 순서는 입력으로 들어오지 않기 때문에, 따로 저장해줘야 한다.

```
struct Person {  
    int age;  
    string name;  
    int join;  
};
```

나이순 정렬

181

<https://www.acmicpc.net/problem/10814>

- <https://gist.github.com/Baekjoon/633c9eaf24cd4606ef01>

stable_sort

182

STL

```
stable_sort(begin, end)
```

[begin, end)를 <를 기준으로 정렬한다.

```
stable_sort(begin, end, cmp)
```

[begin, end)를 cmp를 기준으로 정렬한다.

Stable Sorting

Stable Sorting

- 예를 들어 다음과 같이 카드가 있는 경우를 생각해보자.

• 7♠, 5♥, 2♥, 5♠ 같은

- 위의 카드를 번호가 증가하는 순서로 정렬했을 때, 5♥와 5♠의 순서에 대해 생각해본다.

• 2♥, 5♥, 5♠, 7♠ 입력 순서와 같다

- 와 같이 정렬이 될 수도 있고

• 2♥, 5♠, 5♥, 7♠

- 와 같이 정렬이 될 수도 있다.

- 같은 것이 있는 경우에 정렬하기 전의 순서가 유지되는 정렬 알고리즘을 Stable Sorting 알고리즘이라고 한다.

Stable Sort
비분 Sort
(Merge Sort)

나이순 정렬

184

<https://www.acmicpc.net/problem/10814>

- <https://gist.github.com/Baekjoon/e183e261cb91790a5618>

binary_search

185

STL

`binary_search(begin, end, value)`

`[begin, end)`에서 `value`를 찾으면 `true`, 못 찾으면 `false`

`stable_sort(begin, end, value, cmp)`

`[begin, end)`에서 `value`를 `cmp`를 기준으로 찾으면 `true`, 못 찾으면 `false`

binary_search

186

STL

```
vector<int> a = {1, 5, 6, 7, 10, 20};
```

```
for (int i=1; i<=10; i++) {  
    cout << i << ": ";  
    cout << binary_search(a.begin(), a.end(), i) << '\n';  
}
```

숫자 외우기

187

<https://www.acmicpc.net/problem/10815>

- <https://gist.github.com/Baekjoon/bc5f8660300aadcd60d4>
- <https://gist.github.com/Baekjoon/6446a941d1bbb8186af6>

lower_bound/upper_bound

STL

```
lower_bound(begin, end, value)
```

```
lower_bound(begin, end, value, cmp)
```

```
upper_bound(begin, end, value)
```

```
upper_bound(begin, end, value, cmp)
```

lower_bound: [begin, end)에서 value보다 작지 않은 첫 번째 이터레이터

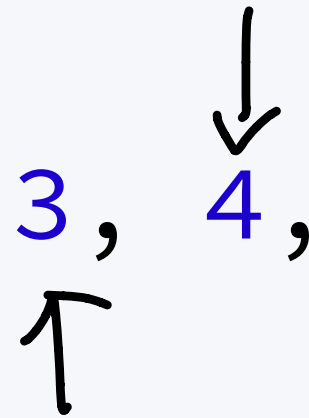
upper_bound: [begin, end)에서 value보다 큰 첫 번째 이터레이터

lower_bound/upper_bound

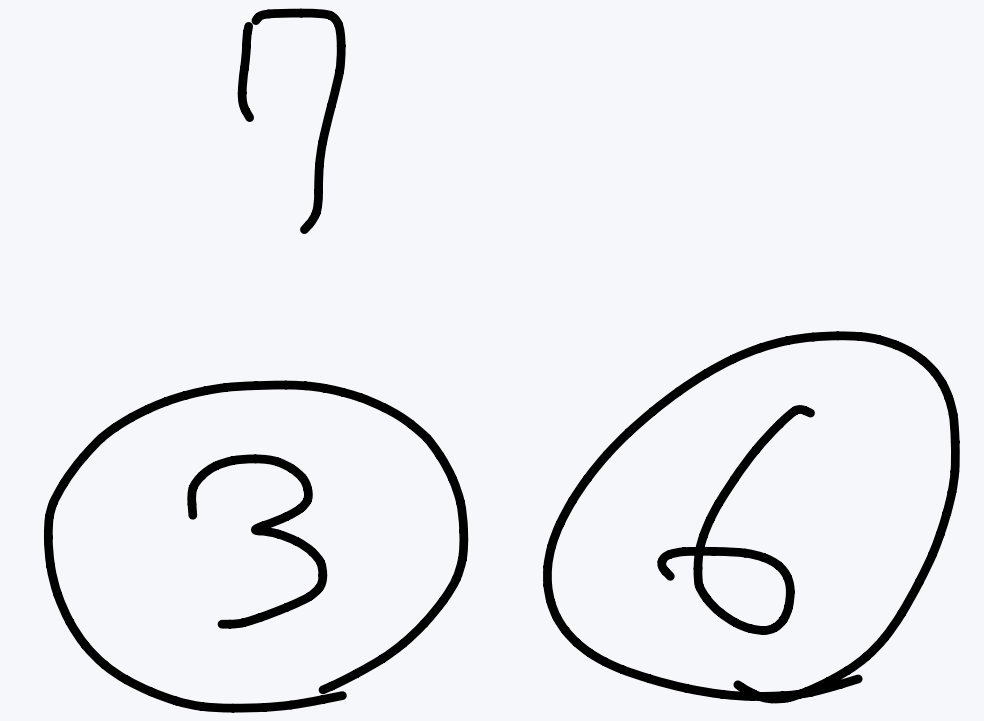
189

STL

```
vector<int> a = {1, 3, 4, 5, 7, 7, 8};
```



```
for (int i=1; i<=10; i++) {  
    auto l = lower_bound(a.begin(), a.end(), i);  
    auto r = upper_bound(a.begin(), a.end(), i);  
  
    cout << i << ": ";  
    cout << "lower_bound: " << (l-a.begin()) << ' ';  
    cout << "upper_bound: " << (r-a.begin());  
    cout << '\n';  
}
```



lower_bound/upper_bound

190

STL

```
1: lower_bound: 0 upper_bound: 1
2: lower_bound: 1 upper_bound: 1
3: lower_bound: 1 upper_bound: 2
4: lower_bound: 2 upper_bound: 3
5: lower_bound: 3 upper_bound: 4
6: lower_bound: 4 upper_bound: 4
7: lower_bound: 4 upper_bound: 6
8: lower_bound: 6 upper_bound: 7
9: lower_bound: 7 upper_bound: 7
10: lower_bound: 7 upper_bound: 7
```

숫자 카드 2

191

<https://www.acmicpc.net/problem/10815>

- <https://gist.github.com/Baekjoon/cd8471e453d9289207f7>

equal_range

STL

192

`equal_range(begin, end, value)`

`equal_range(begin, end, value, cmp)`

`lower_bound`, `upper_bound` 를 pair 형태로 리턴한다

숫자 카드 2

193

<https://www.acmicpc.net/problem/10815>

- <https://gist.github.com/Baekjoon/3eb1336d7c146e00824d>

min/max

194

STL

`min(a, b)`

`min(a, b, cmp)`

`min(initializer_list)`

`min(initializer_list, cmp)`



`max(a, b)`

`max(a, b, cmp)`

`max(initializer_list)`

`max(initializer_list, cmp)`

min/max

STL

274

195

```
cout << min(2, 3) << '\n';
```

```
cout << max(2, 3) << '\n';
```

```
int a = 10, b = 20, c = 30;
```

```
cout << min(min(a, b), c) << '\n';
```

```
cout << min({a, b, c}) << '\n';
```

```
cout << max(max(a, b), c) << '\n';
```

```
cout << max({a, b, c}) << '\n';
```

min/max

196

STL

```
string u = "long string";
```

```
string v = "short";
```

```
cout << min(u, v) << '\n';
```

```
cout << min(u, v, [](string u, string v) {
```

```
    return u.size() < v.size();
```

```
}) << '\n';
```

세 수

<https://www.acmicpc.net/problem/10817>

- <https://gist.github.com/Baekjoon/b71b19d4c9f206c43b3b>

minmax

198

STL

```
minmax(a, b)
```

```
minmax(a, b, cmp)
```

```
minmax(initializer_list)
```

```
minmax(initializer_list, cmp)
```

min과 max를 동시에 구할 수 있다

세 수

<https://www.acmicpc.net/problem/10817>

- <https://gist.github.com/Baekjoon/d65b66ee56b79b86d612>

min_element/max_element

200

STL

```
min_element(begin, end)
```

```
min_element(begin, end, cmp)
```

```
max_element(begin, end)
```

```
max_element(begin, end, cmp)
```

[begin, end)에서 최소/최대값의 이터레이터를 구한다

min_element/max_element

201

STL

```
vector<int> a = {4, 2, 1, 5, 7, 3};  
  
auto it = min_element(a.begin(), a.end());  
cout << "최소: " << *it << ", 위치: " << (it-a.begin()) << '\n';  
  
it = max_element(a.begin(), a.end());  
cout << "최대: " << *it << ", 위치: " << (it-a.begin()) << '\n';
```

최소, 최대

202

<https://www.acmicpc.net/problem/10818>

- <https://gist.github.com/Baekjoon/1d6fddc9dfb9b16e49dc>

minmax_element

203

STL

```
minmax_element(begin, end)
```

```
minmax_element(begin, end, cmp)
```

[begin, end)에서 최소, 최대값의 이터레이터를 구한다

최소, 최대

204

<https://www.acmicpc.net/problem/10818>

- <https://gist.github.com/Baekjoon/1159eb8cd4f7e69b2a90>

next_permutation

STL

`next_permutation(begin, end)`

`next_permutation(begin, end, cmp)`

`prev_permutation(begin, end)`

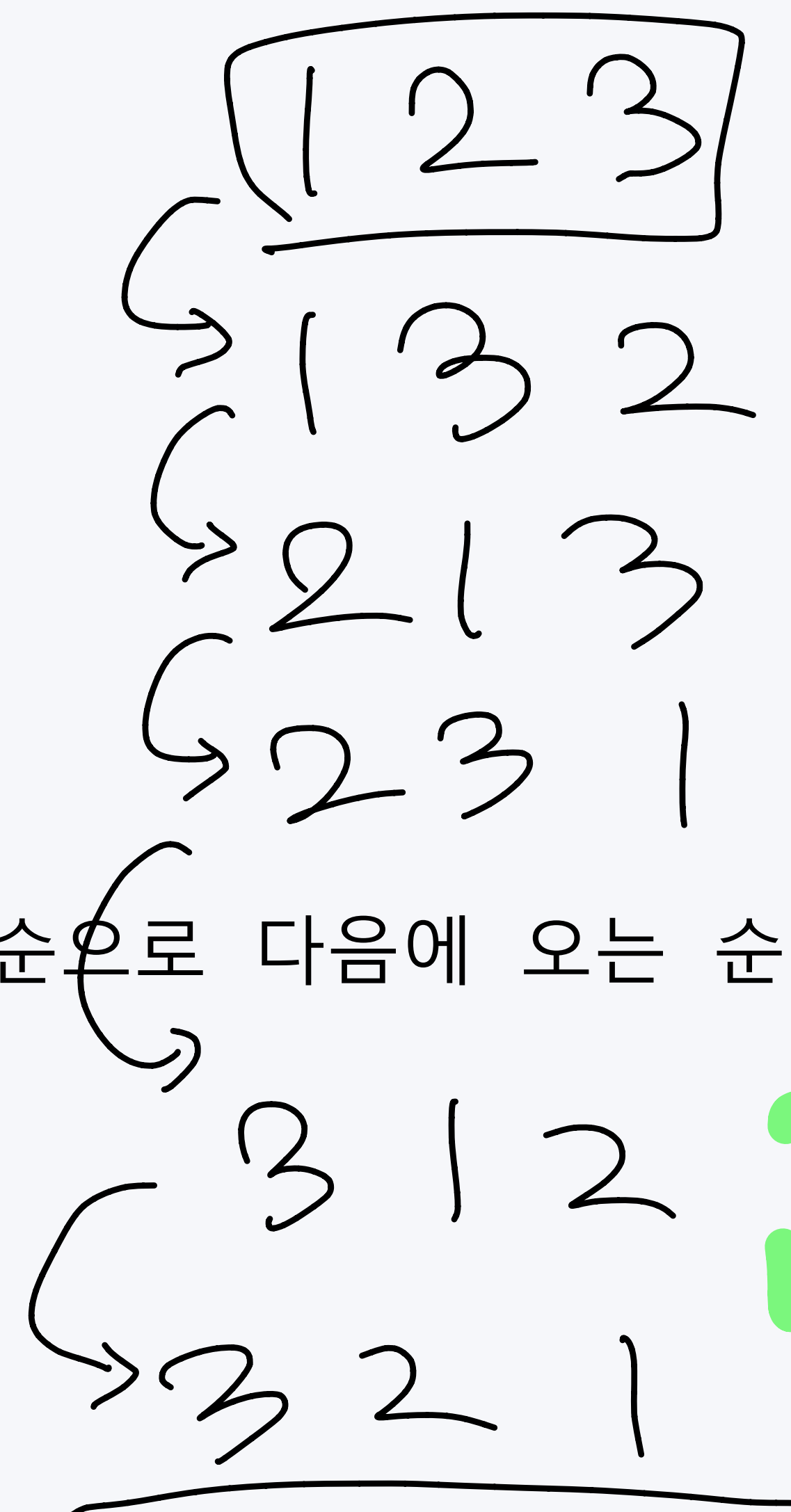
`prev_permutation(begin, end, cmp)`

[begin, end)를 순열이라고 생각했을 때, 사전 순으로 다음에 오는 순열을 만든다

마지막 순열이면 false를 리턴

2!보다 N인 모든 순열

$O(N \times N!)$



$20(N)$

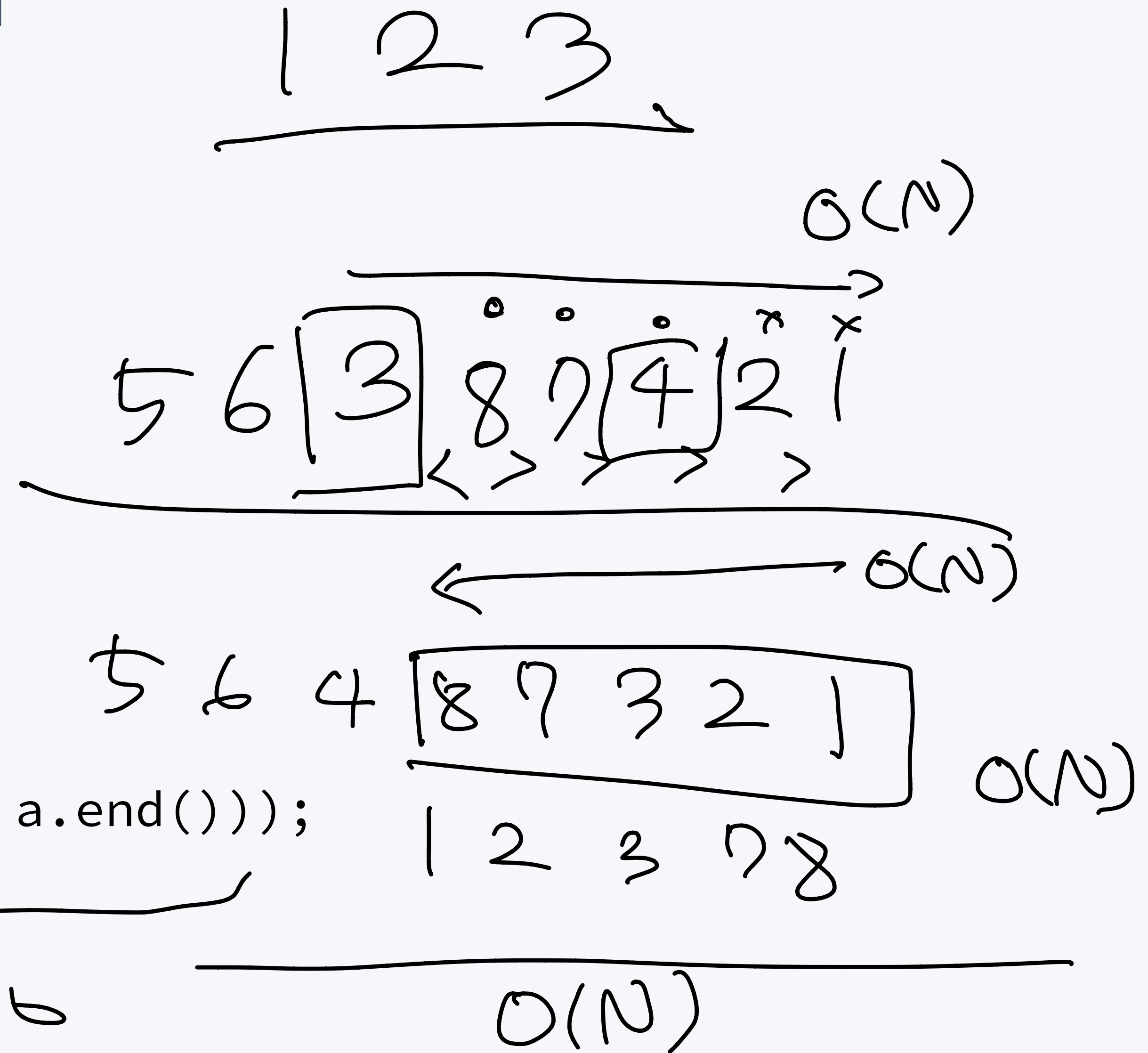
$N \leq 10$

next_permutation

STL

```
vector<int> a = {3, 1, 2};
sort(a.begin(), a.end()); ←
```

```
↓
do {
    for (int x: a) {
        cout << x << ' ';
    }
    cout << '\n';
} while(next_permutation(a.begin(), a.end()));
```



$$10! = 3628800$$

(40)

차이를 최대로

207

<https://www.acmicpc.net/problem/10819>

- <https://gist.github.com/Baekjoon/f141eea58a7313b555fd>

long long

10! Next
2²⁰ 1048576
2¹⁰ DP, Bit
2¹⁶ 65536

$$2^{31} - 1 = 2147483647 \quad (\text{int})$$

C++11

auto

auto

C++

- 컴파일러가 타입을 추론해서 타입을 결정한다
- 변수의 타입을 명확하게 알 수 있어야 한다.
- 아래 코드는 컴파일 에러

```
auto a,b;
```

```
cin >> a >> b;
```

```
cout << a + b << '\n';
```

auto

C++

- 컴파일러가 타입을 추론해서 타입을 결정한다
- 변수의 타입을 명확하게 알 수 있어야 한다.
- 아래 코드는 올바른 코드

```
auto a = 0, b = 0;
```

```
cin >> a >> b;
```

```
cout << a + b << '\n';
```

a = 10

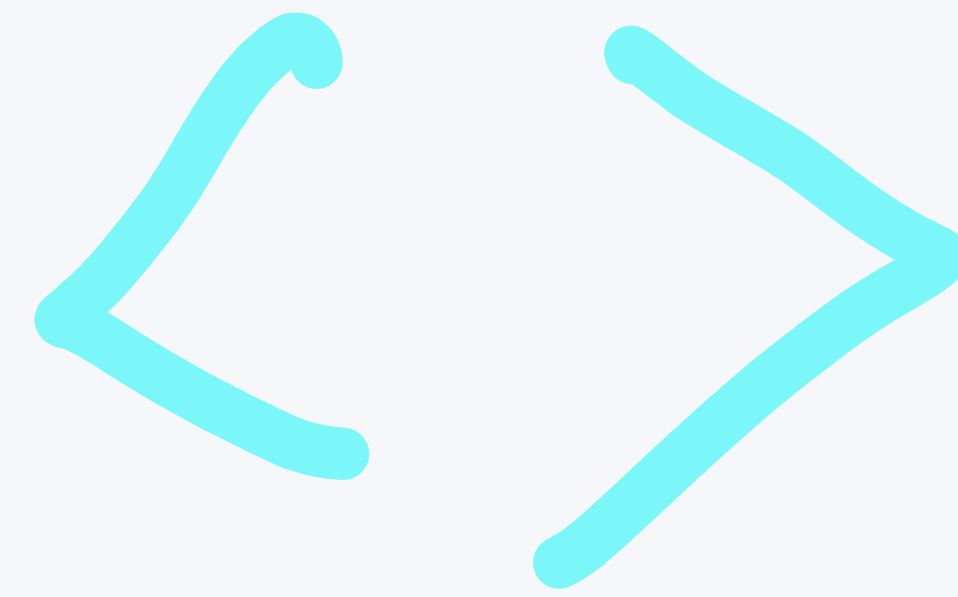
{

a = "abc";

auto

C++

211



- 이터레이터를 사용할 때 매우 편리한다

```
map<pair<int,int>,vector<pair<int,string>>> d;  
for (map<pair<int,int>,vector<pair<int,string>>>::iterator it = d.begin();  
     it != d.end(); ++it) {  
  
}
```

- 이런 코드를

```
map<pair<int,int>,vector<pair<int,string>>> d;  
for (auto it = d.begin(); it != d.end(); ++it) {  
  
}
```

- 이렇게 줄일 수 있다.

Range-based for

Range-based for

C++

```
vector<int> a = {1, 2, 3, 4, 5};
```

```
for (int i=0; i<a.size(); i++) {  
    cout << a[i] << ' '  
}
```

```
cout << '\n';
```

```
for (int x : a) {  
    cout << x << ' '  
}
```

```
cout << '\n';
```

Range-based for

C++

```
vector<pair<int,int>> a = {{1, 2}, {3, 4}, {5, 6}};
```

```
for (int i=0; i<a.size(); i++) {  
    cout << a[i].first + a[i].second << ' '  
}
```

```
cout << '\n';
```


```
for (auto &p : a) {  
    cout << p.first + p.second << ' '  
}
```

```
cout << '\n';
```

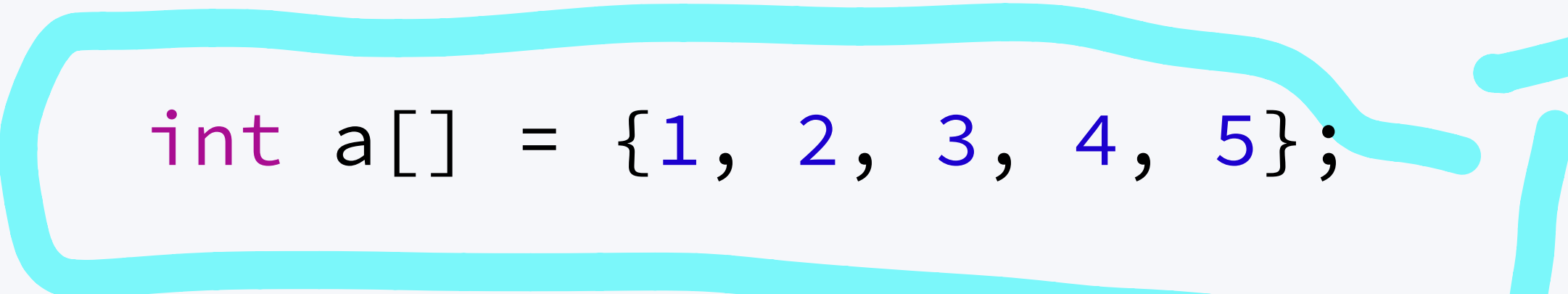
Range-based for

C++


```
int sum = 0;  
for (auto x : {1, 2, 3, 4}) {  
    sum += x;  
}  
cout << "sum = " << sum << '\n';
```



```
int a[] = {1, 2, 3, 4, 5};
```



```
sum = 0;  
for (auto x : a) {  
    sum += x;  
}  
cout << "sum = " << sum << '\n';
```



Range-based for

C++

```
const char cstr[] = "string";  
sum = 0;  
for (auto x : cstr) {  
    sum += 1;  
}  
cout << "sum = " << sum << '\n';
```

7

NULL

```
string str = "string";  
sum = 0;  
for (auto x : str) {  
    sum += 1;  
}  
cout << "sum = " << sum << '\n';
```

6

초기화 리스트

초기화 리스트

218

C++

```
vector<int> a;  
a.push_back(1);  
a.push_back(3);  
a.push_back(7);  
a.push_back(13);  
a.push_back(50);
```

```
vector<int> a = {1, 3, 7, 13, 50};
```

초기화 리스트

219

C++

```
struct Person {  
    string name;  
    int age;  
};
```

```
Struct Person {  
    int age = 10;  
    int birth = 20;  
};
```

```
set<int> s = {1, 2, 3, 4, 5};  
map<int, string> m = { {20, "a"}, {10, "hi"} };  
Person p = {"you", 20};  
map<int, vector<pair<int, int>>> m2 = {  
    {10, {{1, 2}, {3, 4}}},  
    {20, {{5, 6}, {7, 8}, {9, 10}}}  
};
```

람다 함수

4.8

VS

- Std = C++11
C++0x

98
C++03 ??
-?

g++ -O2 -O Main -Std=C++11
Main.cc

람다 함수

C++

[캡처] (함수 인자) {함수 내용}

```
int sum(int x, int y) {  
    return x + y;  
}
```

```
cout << sum(1, 2) << '\n';
```

```
cout << [](int x, int y) {  
    return x + y;  
}(1, 2) << '\n';
```

람다 함수

222

C++

[캡처] (함수 인자) {함수 내용}

```
auto sum2 = [] (int x, int y) {  
    return x + y;  
};
```

```
cout << sum2(1, 2) << '\n';
```

생일 출력하기

223

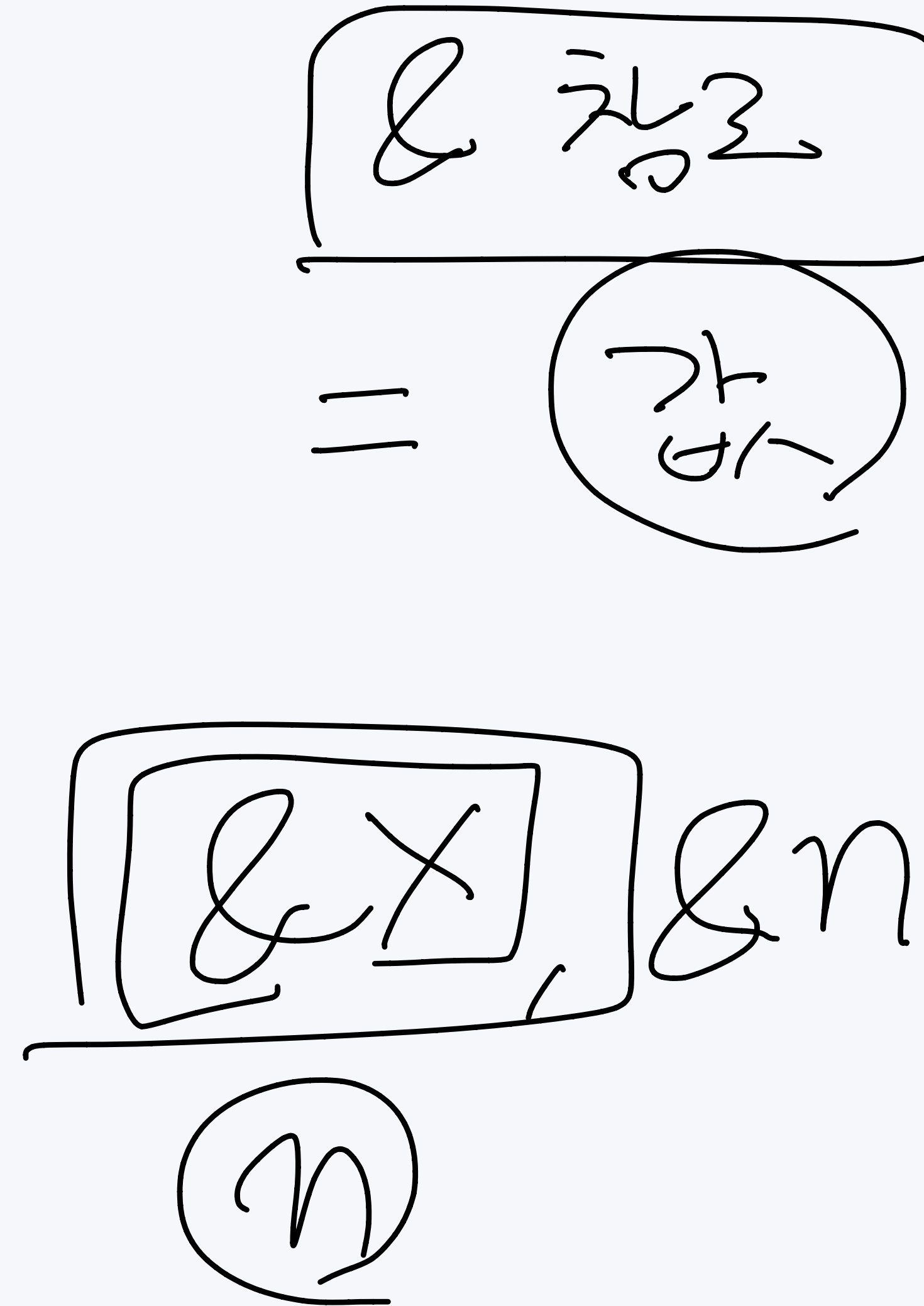
<https://www.acmicpc.net/problem/2555>

```
#include <iostream>
using namespace std;
int main() {
    auto print = []{
        cout << "10/14" << "\n";
    };
    print();
}
```

X보다 작은 수

<https://www.acmicpc.net/problem/10871>

```
int n, x;
cin >> n >> x;
auto is_less = [&](int number) {
    return number < x;
};
for (int i=0; i<n; i++) {
    int num;
    cin >> num;
    if (is_less(num)) {
        cout << num << ' ';
    }
}
cout << '\n';
```



람다 함수

225

C++

[캡처] (함수 인자) {함수 내용}

캡처에 &를 넣으면 선언하는 시점에서 바깥에 있는 변수를 모두 사용할 수 있다

&x와 같이 어떤 변수를 사용할 것인지 적을 수도 있다

&는 참조이고, =는 값 복사이다

여러 개는 ,를 이용할 수 있다

람다 함수

C++

```
int x = 10;
```

```
int y = 20;
```

```
auto f = [&x, y]() {
```

```
    x += 10;
```

```
    //y += 10;
```

```
};
```

mutable

```
cout << "x = " << x << ", y = " << y << "\n";
```

```
f();
```

```
cout << "x = " << x << ", y = " << y << "\n";
```

```
f();
```

```
cout << "x = " << x << ", y = " << y << "\n";
```

람다 함수

227

C++

앞 페이지에서 주석을 해제하면 컴파일 에러를 받게 되는데

컴파일을 하려면

```
auto f = [&x,y]() mutable { 로 선언은 바꿔야 하지만
```

실행 결과는 달라지지 않는다

람다 함수

C++

함수의 변수 타입은 `#include <functional>` 에 선언되어 있다

`function<리턴타입(콤마로 구분한 인자의 타입들)>`

람다 함수

C++

```
function<void()> print = [] {  
};
```

```
function<void(int)> print2 = [](int x) {  
};
```

```
function<int(int, int)> sum = [](int x, int y) {  
    return x+y;  
};
```

return

→ int

[] () → 2개의 인자

{

피보나치 수 5

230

<https://www.acmicpc.net/problem/10870>

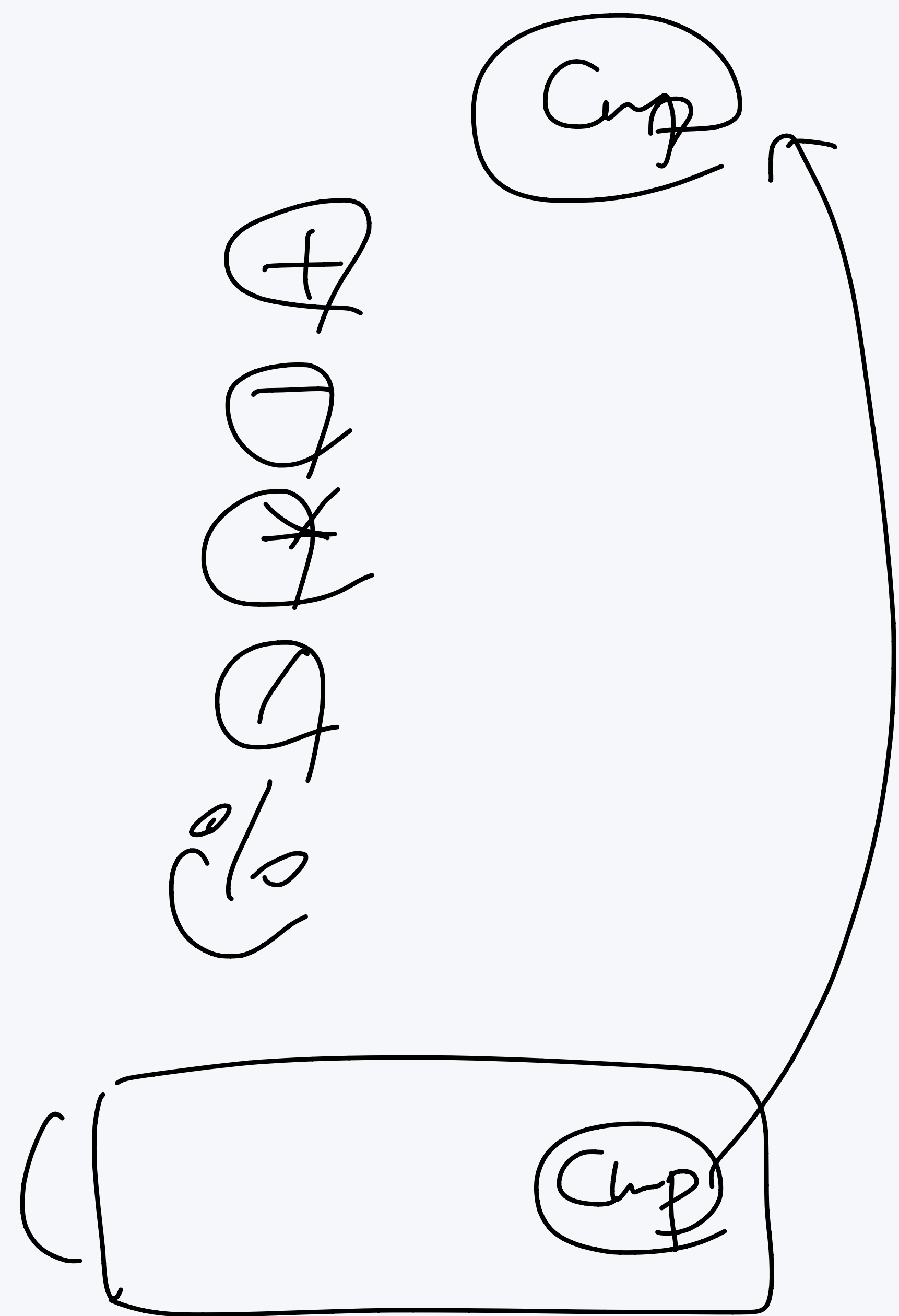
→ double

```
#include <functional>
#include <iostream>
using namespace std;
int main() {
    int n;
    cin >> n;
    function<int(int)> f = &f[&](int n) {
        if (n <= 1) return n;
        else return f(n-1) + f(n-2);
    };
    cout << f(n) << '\n';
    return 0;
}
```

사칙연산

<https://www.acmicpc.net/problem/10869>

```
vector<function<int(int,int)>> d;  
d.push_back([](int x, int y) {  
    return x + y;  
});  
d.push_back([](int x, int y) {  
    return x - y;  
});  
d.push_back([](int x, int y) {  
    return x * y;  
});  
d.push_back([](int x, int y) {  
    return x / y;  
});
```



사칙연산

<https://www.acmicpc.net/problem/10869>

```
d.push_back([](int x, int y) {  
    return x % y;  
});
```

```
for (auto &f : d) {  
    cout << f(a, b) << '\n';  
}
```

Math

N, M

0

Q

$0 \sim N-1$

R

$N \sim N+M-1$

M

Java

Hello World

234

<https://www.acmicpc.net/problem/2557>

```
public class Main {  
    public static void main(String args[]) {  
        System.out.println("Hello World!");  
    }  
}
```

Java

235

Java

파일 이름은 클래스 이름과 같아야 한다

Main.java

컴파일: `javac Main.java`

실행: `java Main`

A+B

236

<https://www.acmicpc.net/problem/1000>

```
import java.util.*;
public class Main {
    public static void main(String args[]) {
        Scanner sc = new Scanner(System.in);
        int a, b;
        a = sc.nextInt();
        b = sc.nextInt();
        System.out.println(a + b);
    }
}
```

Java

237

Java

```
import java.util.*;
public class Main {
    public static void main(String args[]) {
        Scanner sc = new Scanner(System.in);
        int sum = 0;
        while (sc.hasNextInt()) {
            sum += sc.nextInt();
        }
        System.out.println(sum);
    }
}
```

BigInteger

Java

```
import java.math.*;
```

BigInteger

Java

239

```
BigInteger a = new BigInteger("10000");  
BigInteger b = new BigInteger("1000");  
BigInteger c = a.add(b);
```

```
System.out.println("a = " + a);  
System.out.println("b = " + b);  
System.out.println("c = " + c);
```

BigInteger

Java

240

```
System.out.println("a+b = " + a.add(b));
System.out.println("a-b = " + a.subtract(b));
System.out.println("a*b = " + a.multiply(b));
System.out.println("a/b = " + a.divide(b));
System.out.println("a%b = " + a.remainder(b));
System.out.println("gcd(a,b) = " + a.gcd(b));
System.out.println("-a = " + a.negate());

System.out.println("ZERO = " + BigInteger.ZERO);
System.out.println("ONE = " + BigInteger.ONE);
System.out.println("TEN = " + BigInteger.TEN);

System.out.println("a^10 = " + a.pow(10));
```


BigInteger

Java

241

```
BigInteger a = sc.nextBigInteger();
BigInteger b = sc.nextBigInteger();
int c = a.compareTo(b);
if (c < 0) {
    System.out.println("<");
} else if (c == 0) {
    System.out.println("==");
} else {
    System.out.println(">");
}
System.out.println("compareTo = " + c);
```

BigInteger

Java

242

```
boolean e = a.equals(b);  
if (e) {  
    System.out.println("==");  
} else {  
    System.out.println("!=");  
}  
System.out.println("equals = " + e);
```

피보나치 수 4

<https://www.acmicpc.net/problem/10826>

```
BigInteger[] d = new BigInteger[Math.max(n+1,2)];  
d[0] = BigInteger.ZERO;  
d[1] = BigInteger.ONE;  
for (int i=2; i<=n; i++) {  
    d[i] = d[i-1].add(d[i-2]);  
}  
System.out.println(d[n]);
```

<https://gist.github.com/Baekjoon/11f5438a1b7c90dab738>

피보나치 수 4

244

<https://www.acmicpc.net/problem/10826>

```
BigInteger a = BigInteger.ZERO;
BigInteger b = BigInteger.ONE;
BigInteger c = BigInteger.ZERO;
for (int i=2; i<=n; i++) {
    c = a.add(b);
    a = b;
    b = c;
}
System.out.println(c);
```

<https://gist.github.com/Baekjoon/6563dca42979bf549019>

큰 수 A+B

245

<https://www.acmicpc.net/problem/10757>

```
import java.math.*;
import java.util.*;
public class Main{
    public static void main(String args[]){
        Scanner sc = new Scanner(System.in);
        BigInteger a, b;
        a = sc.nextBigInteger();
        b = sc.nextBigInteger();
        System.out.println(a.add(b));
    }
}
```

a^b

246

<https://www.acmicpc.net/problem/10827>

```
import java.math.*;
import java.util.*;
public class Main{
    public static void main(String args[]){
        Scanner sc = new Scanner(System.in);
        BigDecimal a = sc.nextBigDecimal();
        int b = sc.nextInt();

        BigDecimal c = a.pow(b);
        System.out.println(c.toString());
    }
}
```