

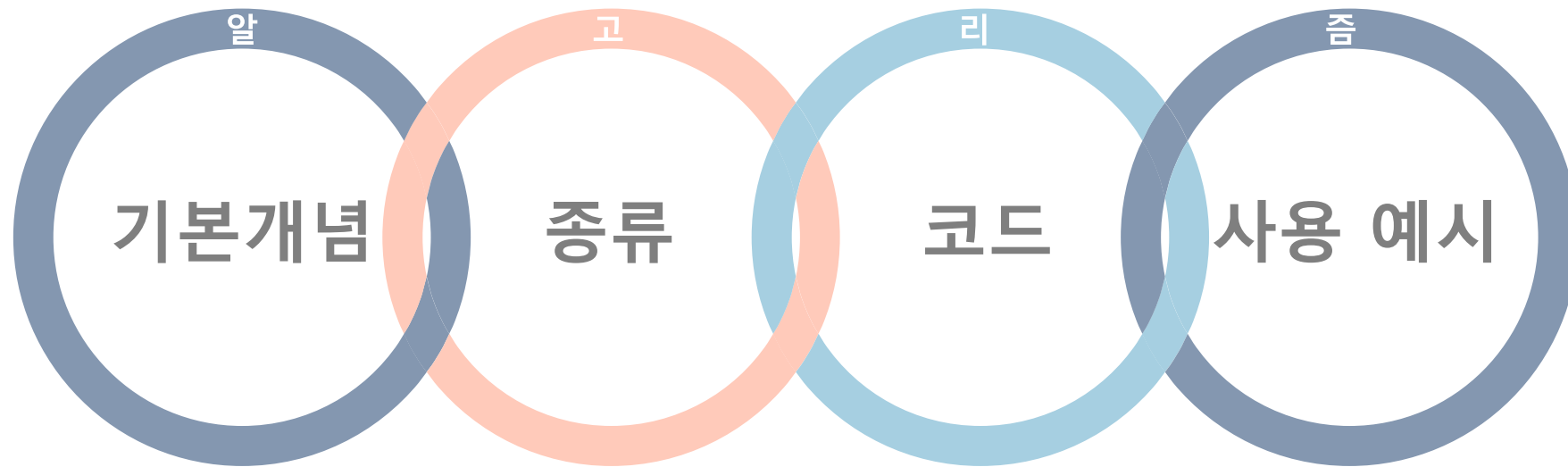
STL_3

Associative Container



글로벌미디어학부
20150783 장명지

목차



연관 컨테이너란?

키(key)		값(value)
나현숙	→	2
양승민	→	1
홍지만	→	2
김병기	→	0
최지웅	→	1

일정규칙에 따라 자료를 조직화하여 저장
자료를 정렬하여 저장하기 때문에 검색 유리
-> 많은 양의 자료 / 빠른 검색

이름(key)을 바탕으로 수업 수(value)
를 얻을 수 있다.

Set

정의: key라 불리는 원소(value)의 집합으로 이루어진 컨테이너

`#include<set>`

`Set<[Data Type]> [변수이름]`

노드 기반 컨테이너이며 균형 이진트리로 구현

Key값은 중복 허용 하지 않음

원소가 삽입되면 자동으로 정렬됨

Map과 다르게 key만 저장

Set

정의: key라 불리는 원소(value)의 집합으로 이루어진 컨테이너

```
#include<set>
```

```
Set<[Data Type]> [변수이름]
```

<이 조건에 사용하면 좋다!>

정렬해야 한다.

key가 있는지 없는지 알아야 할 때

많은 자료를 저장하고, 검색 속도가 빨라야 할 때

Set

정의: key라 불리는 원소(value)의 집합으로 이루어진 컨테이너

함수	설명
insert(i)	i 추가
find(i)	i 와 연관된 원소의 반복자 반환
erase(i)	i 위치의 원소 삭제
...	...

Set

정의: key라 불리는 원소(value)의 집합으로 이루어진 컨테이너

<Insert>

```
set<int> s;  
s.insert(40);  
s.insert(50);  
s.insert(10);  
s.insert(30);  
  
set<int>::iterator it;  
for (it = s.begin(); it != s.end(); it++)  
    cout << *it << " ";  
cout << endl;  
//중복값 넣어보기  
s.insert(10);  
for (it = s.begin(); it != s.end(); it++)  
    cout << *it << " ";  
cout << endl;
```

```
10 30 40 50  
10 30 40 50
```

<erase>

```
//원소 제거하기  
s.erase(30);  
for (it = s.begin(); it != s.end(); it++)  
    cout << *it << " ";  
cout << endl;  
//없는 원소 제거하기  
s.erase(20);  
for (it = s.begin(); it != s.end(); it++)  
    cout << *it << " ";  
cout << endl;
```

```
10 40 50  
10 40 50
```

Set

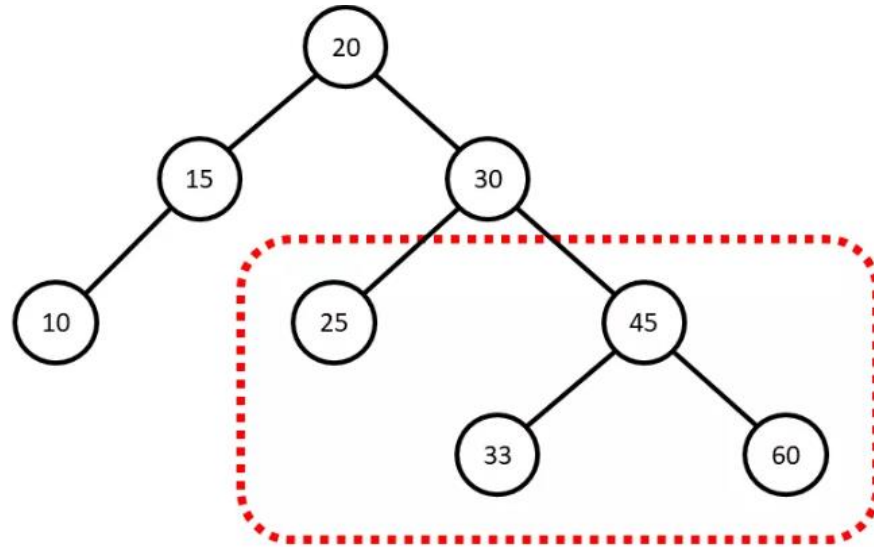
정의: key라 불리는 원소(value)의 집합으로 이루어진 컨테이너

<find>

```
//존재 하지 않는 원소 찾기
it = s.find(20);
if (it != s.end()) {
    cout <<*it<< ": 존재" << endl;
}
else {
    cout << "20 없음" << endl;
}

//존재 하는 원소 찾기
it = s.find(30);
if (it != s.end()) {
    cout <<*it<< ": 존재" << endl;
}
else {
    cout << "30 없음" << endl;
}
```

```
20 없음
30: 존재
10 15 20 25 30 33 45 60
```



$O(\log n)$

multiset

정의: key라 불리는 원소(value)의 집합으로 이루어진 컨테이너

#include<set>

Set과 구별되는 가장 큰 특징

multiset<[Data Type]> [변수이름]

Key 값이 **중복**된다

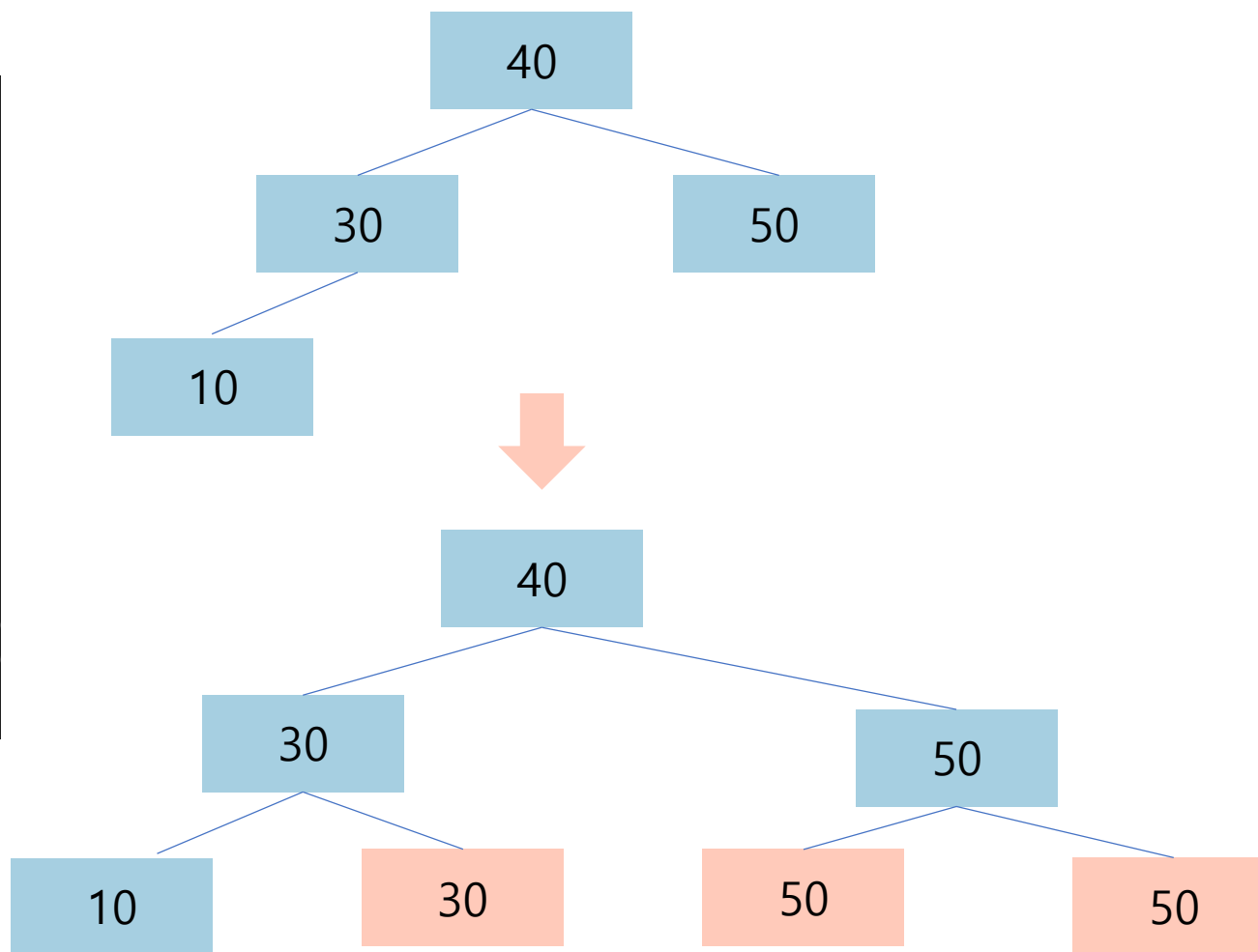
multiset

정의: key라 불리는 원소(value)의 집합으로 이루어진 컨테이너

<Insert>

```
multiset<int> s;  
s.insert(40);  
s.insert(50);  
s.insert(10);  
s.insert(30);  
  
set<int>::iterator it;  
for (it = s.begin(); it != s.end(); it++)  
    cout << *it << " ";  
cout << endl;  
//중복값 넣어보기  
s.insert(30);  
s.insert(50);  
s.insert(50);  
for (it = s.begin(); it != s.end(); it++)  
    cout << *it << " ";  
cout << endl;
```

```
10 30 40 50  
10 30 30 40 50 50 50
```



multiset

정의: key라 불리는 원소(value)의 집합으로 이루어진 컨테이너

<erase>

```
main() {  
    multiset<int> s;  
  
    s.insert(40);  
    s.insert(50);  
    s.insert(10);  
    s.insert(30);  
  
    s.insert(30);  
    s.insert(30);  
    s.insert(30);  
    s.insert(30);  
    //s.erase(30);  
    multiset<int>::iterator i;  
    for (i = s.begin(); i != s.end(); i++)  
        cout << *i << " ";  
}
```

10 30 30 30 30 30 40 50 [



10 40 50 [

Set

정의: key라 불리는 원소(value)의 집합으로 이루어진 컨테이너

중복+정렬

백준 7785 회사에 있는 사람

백준 1181 단어 정렬

백준 1822 차집합

Set 정의: key라 불리는 원소(value)의 집합으로 이루어진 컨테이너

백준 7785 회사에 있는 사람

상근이는 세계적인 소프트웨어 회사 기글에서 일한다. 이 회사의 가장 큰 특징은 자유로운 출퇴근 시간이다. 따라서, 직원들은 반드시 9시부터 6시까지 회사에 있지 않아도 된다.

각 직원은 자기가 원할 때 출근할 수 있고, 아무때나 퇴근할 수 있다.

상근이는 모든 사람의 출입카드 시스템의 로그를 가지고 있다. 이 로그는 어떤 사람이 회사에 들어왔는지, 나갔는지가 기록되어져 있다. 로그가 주어졌을 때, 현재 회사에 있는 모든 사람을 구하는 프로그램을 작성하시오.

백준 1181 단어 정렬

알파벳 소문자로 이루어진 N개의 단어가 들어오면 아래와 같은 조건에 따라 정렬하는 프로그램을 작성하시오.

1. 길이가 짧은 것부터
2. 길이가 같으면 사전 순으로

백준 1822 차집합

몇 개의 자연수로 이루어진 두 집합 A와 B가 있다. 집합 A에는 속하면서 집합 B에는 속하지 않는 모든 원소를 구하는 프로그램을 작성하시오.

Set

정의: key라 불리는 원소(value)의 집합으로 이루어진 컨테이너

백준 7785 회사에 있는 사람

상근이는 세계적인 소프트웨어 회사 기글에서 일한다. 이 회사의 가장 큰 특징은 자유로운 출퇴근 시간이다. 따라서, 직원들은 반드시 9시부터 6시까지 회사에 있지 않아도 된다.

각 직원은 자기가 원할 때 출근할 수 있고, 아무때나 퇴근할 수 있다.

상근이는 모든 사람의 출입카드 시스템의 로그를 가지고 있다. 이 로그는 어떤 사람이 회사에 들어왔는지, 나갔는지가 기록되어져 있다. 로그가 주어졌을 때, 현재 회사에 있는 모든 사람을 구하는 프로그램을 작성하시오.

출력

현재 회사에 있는 사람의 이름을 사전 순의 역순으로 한 줄에 한 명씩 출력한다.

예제 입력 1 복사

```
4
Baha enter
Askar enter
Baha leave
Artem enter
```

예제 출력 1 복사

```
Askar
Artem
```

```
Set<string> s;
Cin >> str >> name;
If(str=="enter")
    s.insert(name);
Else
    s.erase(name);
```

Set

정의: key라 불리는 원소(value)의 집합으로 이루어진 컨테이너

백준 1181 단어 정렬

알파벳 소문자로 이루어진 N개의 단어가 들어오면 아래와 같은 조건에 따라 정렬하는 프로그램을 작성하시오.

1. 길이가 짧은 것부터
2. 길이가 같으면 사전 순으로

출력

조건에 따라 정렬하여 단어들을 출력한다. 단, 같은 단어가 여러 번 입력된 경우에는 한 번씩만 출력한다.

예제 입력 1 복사

```
13
but
i
wont
hesitate
no
more
no
more
it
cannot
wait
im
yours
```

예제 출력 1 복사

```
i
im
it
no
but
more
wait
wont
yours
cannot
hesitate
```

```
Set<pair<long,string> > s;
For(int i=0;i<n;i++)
    Cin>>str;
    s.insert(make_pair(str.length(),str));
```

Set

정의: key라 불리는 원소(value)의 집합으로 이루어진 컨테이너

백준 1822 차집합

몇 개의 자연수로 이루어진 두 집합 A와 B가 있다. 집합 A에는 속하면서 집합 B에는 속하지 않는 모든 원소를 구하는 프로그램을 작성하시오.

출력

첫째 줄에 집합 A에는 속하면서 집합 B에는 속하지 않는 원소의 개수를 출력한다. 다음 줄에는 구체적인 원소를 빈 칸을 사이에 두고 증가하는 순서로 출력한다. 집합 A에는 속하면서 집합 B에는 속하지 않는 원소가 없다면 첫째 줄에 0만을 출력하면 된다.

예제 입력 1 복사

```
4 3
2 5 11 7
9 7 4
```

예제 출력 1 복사

```
3
2 5 11
```

```
set<int> s;
For(int i=0;i<A;i++)
    cin>>num;
    s.insert(num);
For(int i=0;i<B;i++)
    It=s.find(num);
    If(it!=end()) //존재하면
        s.erase(it);
    Else
        continue; //없으면
```


map 정의: key와 value가 한번에 저장되는 컨테이너

#include<map>

map<[Data Type1], Data Type2>> [변수이름]

노드 기반 컨테이너이며 균형 이진트리로 구현

Key값은 중복 허용 하지 않음

Key와 value로 이루어져 있으며 pair 객체 형태로 저장

삽입되면서 자동 정렬

map 정의: key와 value가 한번에 저장되는 컨테이너

#include<map>

map<[Data Type1], Data Type2>> [변수이름]

<이 조건에 사용하면 좋다!>

정렬해야 한다.

많은 자료를 저장하고 검색이 빨라야 할 때

빈번하게 삽입, 삭제하지 않을 때

map 정의: key와 value가 한번에 저장되는 컨테이너

함수	설명
insert(i)	i 추가
find(i)	i 와 연관된 원소의 반복자 반환
erase(i)	i 위치의 원소 삭제
...	...

map 정의: key와 value가 한번에 저장되는 컨테이너

<insert>

```
map<int, string> m;
//삽입하기
m.insert(pair<int, string>(10, "도라에몽"));
m.insert(pair<int, string>(40, "노진구"));
m.insert(pair<int, string>(50, "통통이"));
m.insert(pair<int, string>(20, "비실이"));
m.insert(pair<int, string>(30, "이슬이"));
map<int, string>::iterator it;
for (it = m.begin(); it != m.end(); it++)
    cout << "[" << it->first << ", " << it->second << " ] ";
//cout << "[" << (it*)->first << ", " << (it*)->second << " ] ";도 같은 기능
cout << endl;
cout << endl;
cout << endl;
//중복값 넣어보기
m.insert(pair<int, string>(10, "도라에몽"));
for (it = m.begin(); it != m.end(); it++)
    cout << "[" << it->first << ", " << it->second << " ] ";
```

[10,도라에몽] [20,비실이] [30,이슬이] [40,노진구] [50,통통이]

[10,도라에몽] [20,비실이] [30,이슬이] [40,노진구] [50,통통이]

<operator>

```
map<int, string> mm;
mm[1] = "도라에몽";
mm[10] = "노진구";
mm[50] = "통통이";
mm[40] = "비실이";
//cout << "출력" << endl;
map<int, string>::iterator it;
for (it = mm.begin(); it != mm.end(); it++)
    cout << "[" << it->first << ", " << it->second << " ] ";

//변경후 순회
mm[1] = "짱구";
```

[1,도라에몽] [10,노진구] [40,비실이] [50,통통이]

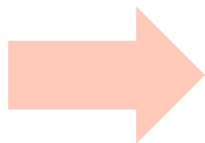
[1,짱구] [10,노진구] [40,비실이] [50,통통이]

map 정의: key와 value가 한번에 저장되는 컨테이너

<erase>

```
//삭제하기
for (it = m.begin(); it != m.end(); it++)
    cout << "[" << it->first << "," << it->second << "]" ";

for (it = m.begin(); it != m.end(); it++) {
    if (it->second.compare("도라에몽"))
        m.erase(it);
}
```



Microsoft Visual C++ Runtime Library



Debug Assertion Failed!

Program: C:\Users\AA\source\repos\ÇÇ±Ü±\Debug\ÇÇ±Ü±.exe
File: c:\program files (x86)\microsoft visual studio\2017\community\vc\tools\msvc\14.16.27023\include\xtree
Line: 236

Expression: cannot increment value-initialized map/set iterator

For information on how your program can cause an assertion failure, see the Visual C++ documentation on asserts.

(Press Retry to debug the application)

종단(A)

다시 시도(R)

무시(I)

Erase(it) 실행 후에 반복자 it의 값은 엉뚱한 포인터를 가리켜서 무한루프나 오작동 생길 수 있음

map 정의: key와 value가 한번에 저장되는 컨테이너

<erase>

```
//삭제하기
for (it = m.begin(); it != m.end(); it++)
    cout << "[" << it->first << "," << it->second << "]" ";

for (it = m.begin(); it != m.end(); it++) {
    if (it->second.compare("도라에몽"))
        m.erase(it++);
    else
        ++it;
}
```

it++를 실행하면 erase() 함수로 들어가기 전에 it의 값은 증가 되어있지만 erase()에 전달되는 값은 증가시키기 전의 값이다.

multimap 정의: key와 value가 한번에 저장되는 컨테이너

```
#include <map>
```

```
multimap<[Data Type1], Data Type2>> [변수이름]
```

map과 구별되는 가장 큰 특징

Key 값이 **중복**된다

Operator[]를 사용해서 원소
pair<key,value> 추가 수정이 불가능하다

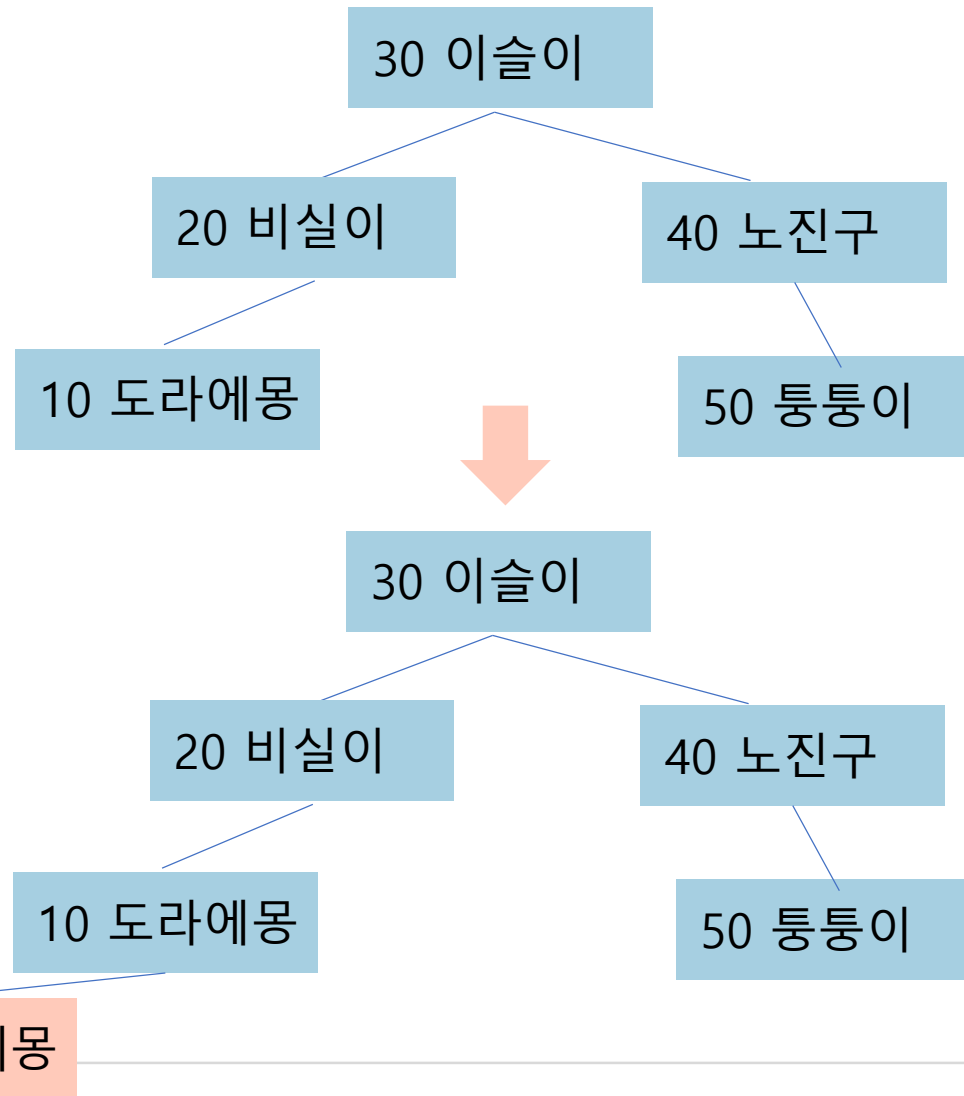
multimap

정의: key와 value가 한번에 저장되는 컨테이너

<insert>

```
multimap<int, string> m;  
//삽입하기  
m.insert(pair<int, string>(10, "도라에몽"));  
m.insert(pair<int, string>(40, "노진구"));  
m.insert(pair<int, string>(50, "통통이"));  
m.insert(pair<int, string>(20, "비실이"));  
m.insert(pair<int, string>(30, "이슬이"));  
m.insert(pair<int, string>(10, "도라에몽"));  
map<int, string>::iterator it;  
for (it = m.begin(); it != m.end(); it++) {  
    cout << "[" << it->first << ", " << it->second << "] ";  
    cout << endl;  
}
```

```
[10,도라에몽]  
[10,도라에몽]  
[20,비실이]  
[30,이슬이]  
[40,노진구]  
[50,통통이]
```



multimap

정의: key와 value가 한번에 저장되는 컨테이너

<operator>

```
multimap<int, string> mm;  
mm[1] = "도라에몽";  
mm[10] = "노진구";  
mm[50] = "통통이";  
mm[40] = "비실이";
```

Map과 달리 operator을 아예
제공하지 않음

map 정의: key와 value가 한번에 저장되는 컨테이너

문자 or 숫자+ 횟수 백준 11652 카드

백준 1157 단어 공부

* Map은 key, value로 이루어져 있기 때문에
넣은 숫자나 문자가 이미 존재하는지 따로 검사해 줄 필요가 없다

map 정의: key와 value가 한번에 저장되는 컨테이너

백준 1157 단어 공부

문제

알파벳 대소문자로 된 단어가 주어지면, 이 단어에서 가장 많이 사용된 알파벳이 무엇인지 알아내는 프로그램을 작성하시오. 단, 대문자와 소문자를 구분하지 않는다.

백준 1302 베스트셀러

김형택은 탐문고의 직원이다. 김형택은 계산대에서 계산을 하는 직원이다. 김형택은 그날 근무가 끝난 후에, 오늘 판매한 책의 제목을 보면서 가장 많이 팔린 책의 제목을 칠판에 써놓는 일도 같이 하고 있다.

오늘 하루 동안 팔린 책의 제목이 입력으로 들어왔을 때, 가장 많이 팔린 책의 제목을 출력하는 프로그램을 작성하시오.

map 정의: key와 value가 한번에 저장되는 컨테이너

백준 1157 단어 공부

문제

알파벳 대소문자로 된 단어가 주어지면, 이 단어에서 가장 많이 사용된 알파벳이 무엇인지 알아내는 프로그램을 작성하시오. 단, 대문자와 소문자를 구분하지 않는다.

출력

첫째 줄에 이 단어에서 가장 많이 사용된 알파벳을 대문자로 출력한다. 단, 가장 많이 사용된 알파벳이 여러 개 존재하는 경우에는 ?를 출력한다.

예제 입력 1 복사

```
Mississippi
```

예제 출력 1 복사

```
?
```

```
Cin>>str;
For(int i=0;i<str.size();i++)
Map<char,int> m;
It=m.find(str[i]);
If(it!=m.end()) //존재하면
    m[it->first]+=1;
Else
    m[str[i]]=1; //없으면
```

map 정의: key와 value가 한번에 저장되는 컨테이너

백준 1302 베스트셀러

김형택은 탐문고의 직원이다. 김형택은 계산대에서 계산을 하는 직원이다. 김형택은 그날 근무가 끝난 후에, 오늘 판매한 책의 제목을 보면서 가장 많이 팔린 책의 제목을 칠판에 써놓는 일도 같이 하고 있다.

오늘 하루 동안 팔린 책의 제목이 입력으로 들어왔을 때, 가장 많이 팔린 책의 제목을 출력하는 프로그램을 작성하시오.

첫째 줄에 가장 많이 팔린 책의 제목을 출력한다. 만약 가장 많이 팔린 책이 여러 개일 경우에는 사전 순으로 가장 앞서는 제목을 출력한다.

예제 입력 1 복사

```
5
top
top
top
top
kimtop
```

예제 출력 1 복사

```
top
```

```
Cin>>n;
Map<string,int> m
String str;
While(n--){
    cin>>str;
    m[str]++;
}
```

