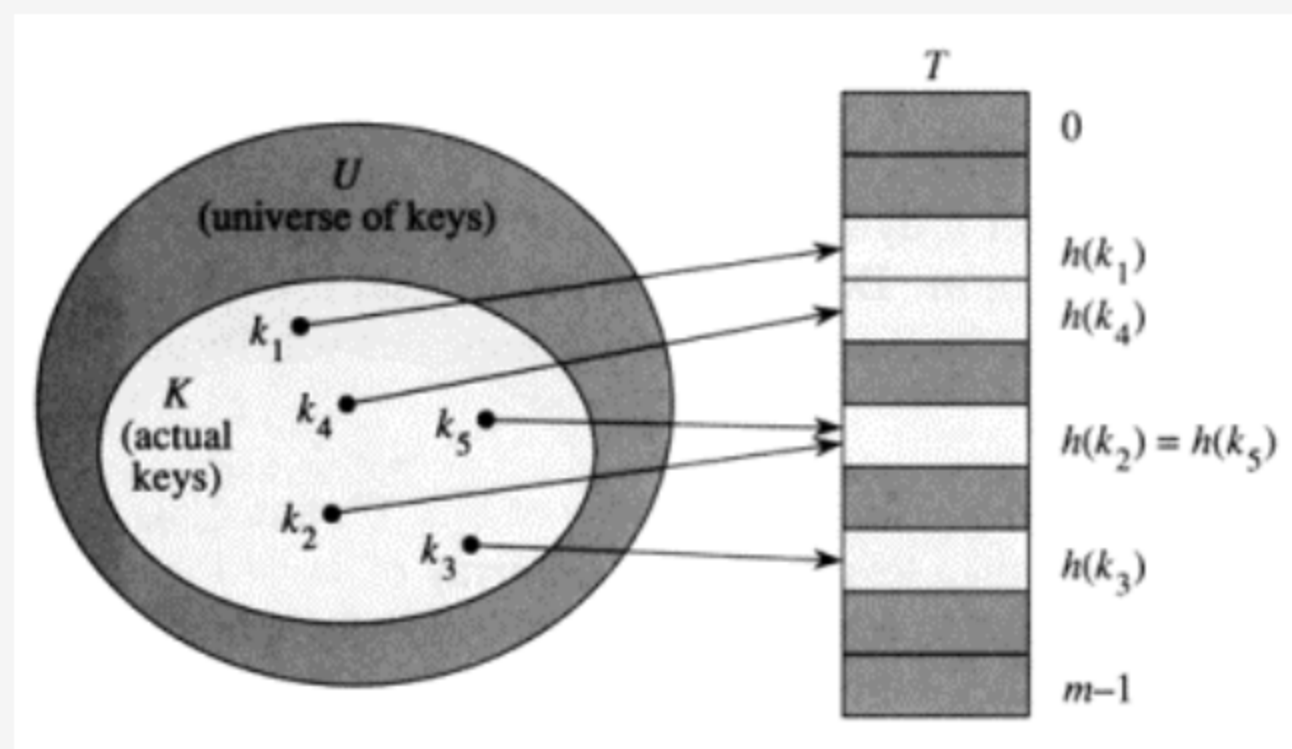


06. 해시

해시란?

임의의 길이의 데이터를 **고정된 길이의 데이터**로 매핑하는 행위 - (Wikipedia : hash function)

특정 배열의 인덱스 또는 value 값을 "데이터 값"을 이용해 찾을 수 있다.



해시 - Direct Addressing Table

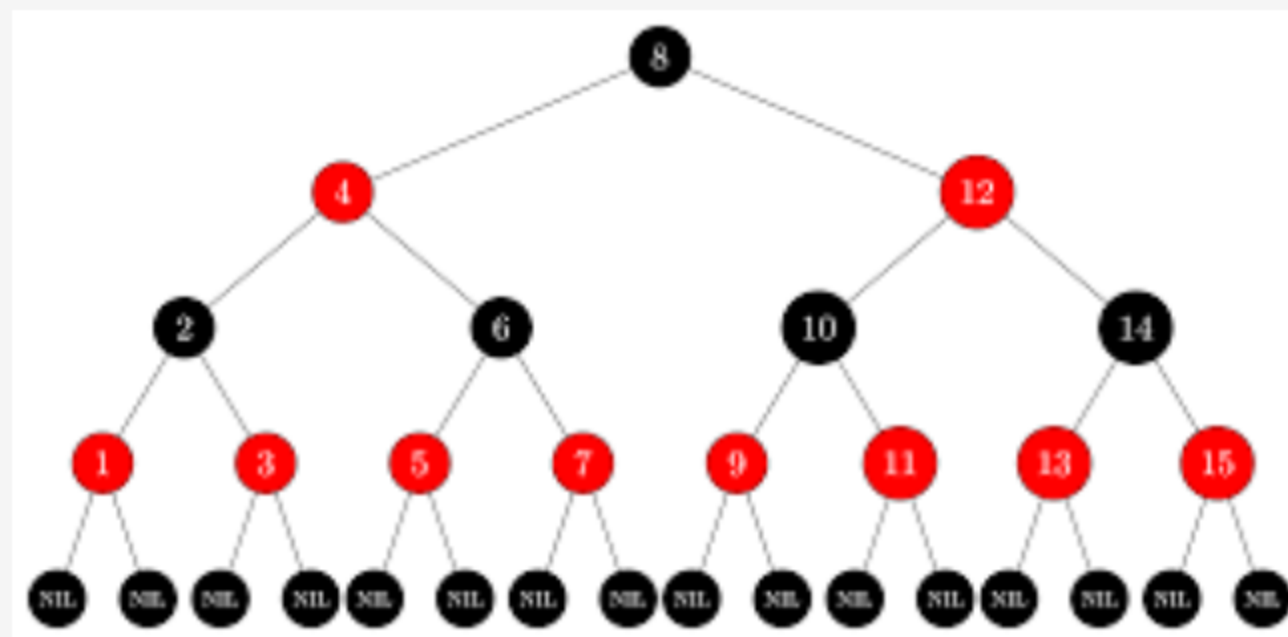
Map (C++ STL)

<key, value> 쌍을 저장한다. Python의 Dictionary 와 유사

기본적으로 균형 잡힌 이진트리 구조 (red-black tree 구현)

-> 원소가 삽입 되면서 자동으로 정렬된다!

-> sort 없이 iterator를 이용해 정렬된 상태로 순회 가능!



Map (C++ STL)

key는 고유한 값으로 중복이 불가능! (중복은 `std::multimap` 이용)

key, value 값은 사용자 정의에 따라서 어떤 자료형도 들어갈 수 있다!

Map 사용법 - 선언, 삽입, 순회

```
int main()
{
    map<string, int>mp;

    mp["hi"] = 1;
    mp["koala"] = 2;
    mp["bye"] = 3;

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

    return 0;
}
```

Map 사용법 - 검색

```
int main()
{
    map<string, int>mp;

    mp["hi"] = 1;
    mp["koala"] = 2;
    mp["bye"] = 3;

    cout << mp.count("hi") << "\n"; // 1
    cout << mp.count("bye") << "\n"; // 1
    cout << mp.count("??") << "\n"; // 0

    cout << mp["hi"] << "\n"; // 1
    cout << mp["koala"] << "\n"; // 2
    cout << mp["??"] << "\n"; // 0

    return 0;
}
```

나는야 포켓몬 마스터 이다솜

BOJ 1620번

재미있는 문제

포켓몬의 이름이 쪽 주어지고, 주어진 질문에 대한 답을 출력하는 문제



나는야 포켓몬 마스터 이다솜

BOJ 1620번

두 가지 질문

1. 이름이 주어지면, 순서 출력
2. 순서가 주어지면, 이름 출력

나는야 포켓몬 마스터 이다솜

BOJ 1620번

해시를 안쓰면?

1. 이름과 순서를 pair로 저장
2. 각 질문에 따라 pair를 순회하며 찾고, 출력

-> 개수 N , 질문 M 일때 시간복잡도 $O(NM)$ -> 시간 초과!

나는야 포켓몬 마스터 이다솜

BOJ 1620번

해시를 쓰면?

1. <이름, 순서>, <순서, 이름> 각각 map에 저장 (map 2개)

2. 각 질문에 따라 이미 매핑된 결과 출력

-> 각 질문마다 $O(1)$, 총 시간복잡도 $O(M)$

패션왕 신해빈

BOJ 9375번

의상의 종류와 이름이 무작위로 주어졌을 때, 해빈이가 입을 수 있는 데일리 룩의 가짓수

패션왕 신해빈

BOJ 9375번

풀이

1. 의상의 종류가 중요, 의상의 이름은 굳이 알 필요 없다!
2. 의상의 종류 당 몇 개 있는지 세야 한다! -> map 사용
3. 경우의 수 계산

패션왕 신해빈

BOJ 9375번

경우의 수

- 알몸이 아니면, 어떤 경우도 가능

정답 : (전체 경우의 수) - 알몸



무한 수열

BOJ 1351번

무한 수열 A 는 다음과 같다.

$$A_0 = 1$$

$$A_i = A[i/P] + A[i/Q] \quad (i \geq 1)$$

N , P 와 Q 가 주어질 때, A_N 을 구하는 프로그램을 작성하시오.

무한 수열

BOJ 1351번

풀이

1. 우선 N제한이 10^{12} 이므로 배열로 표현 불가
2. 우선 예제 과정 그림으로 그려보기

무한 수열

BOJ 1351번

풀이

문제에서 P, Q가 최소 2 이므로 트리의 높이도 최대 $\log N$

Top-Down dp로 풀이 가능!

단, 중복 제거를 위한 메모이제이션 필수!

무한 수열

BOJ 1351번

소스 코드 ->

```
#include <bits/stdc++.h>

using namespace std;

typedef long long ll;

ll n, p, q;
map<ll, ll>dp_map;

ll go(ll n)
{
    //이미 값이 들어 있다면 -> 리턴
    if (dp_map[n] > 0) return dp_map[n];
    //memoization
    return dp_map[n] = go(n / p) + go(n / q);
}

int main()
{
    ios_base::sync_with_stdio(0); cin.tie(0);
    //A0 = 1
    dp_map[0] = 1;
    cin >> n >> p >> q;
    cout << go(n) << "\n";
    return 0;
}
```