lv1달리기\_경주

-vector에서 find함수 존재?

vector 헤더파일에는 즉 vector 클래스에는 find함수가 정의되어 있지 않다. 그래서 algorithm 헤더파일에서 find함수를 이용하면 된다.// find와 find\_if가 존재한다.

find함수는 반복자를 반환한다. 따라서 auto it 로 받으면 된다. vector<int>::iterator it 이렇게 받을 수도 있음

-distance : 반복자를 반환하는 함수들 로부터 인덱스를 구하기 위해서 사용하는 함수

distance(it1,it2)는 it2-it1을 구하는 함수인데 이것에 it1에 vec.begin()을 넣으면, it2의 인덱스 값이 나오게 된다. distance는 정수 반환형을 가진 함수이다. 그런데 음수가 될수도 있고 하니까 auto로 받는 편이 좋겠다.

////달리기 경주2

-map삽입의 2가지 방법

m.insert(pair<int, int>(2, 20)); // pair객체를 생성하여 삽입

m.insert(make\_pair(3, 30)); //make\_pair를 해줄 때는 pair처럼 자료형을 써줄 필요가 없다.

-map의 iterator는 양방향 반복자라서 iterator-1등이 허용되지 않는다. 그리하여 prev()나 이런 다른 함수들을 사용해야만 한다.

auto it = myMap.find(2); // key = 2 찾기 auto prevIt = std::prev(it); // 바로 앞의 반복자

-map의 원소에 접근 할때는 반복자를 이용하는 편이 수월하다.

for (auto it = myMap.begin(); it != myMap.end(); ++it) {

std::cout << "Key: " << it->first << " -> Value: " << it->second << std::endl;

어라?? 그러면 양방향 반복자는 연산이 되는구나!! 단지 산술연산과 같이 it-2 it+1만 되지 않을 뿐이구나!!