

[프로그래머스/Lev2] 타겟 넘버 (DFS)

내 쪼대로 공부/Algorithm 2019.08.05 14:34

인턴도 끝났겠다,

새로운 마음으로 다시 시작하는 코딩테스트 연습.

프로그래머스에서 DFS/BFS 부터 공부를 시작해보려고 한다.

첫 문제는 타겟 넘버. 난이도 쉬움. ~~하지만 헛갈려서 쉽게 풀지 못했음~~

Q. 타겟넘버

n개의 음이 아닌 정수가 있습니다.

이 수를 적절히 더하거나 빼서 타겟 넘버를 만들려고 합니다.

예를 들어 [1, 1, 1, 1, 1]로 숫자 3을 만들려면 다음 다섯 방법을 쓸 수 있습니다.

$$-1+1+1+1+1 = 3$$

$$+1-1+1+1+1 = 3$$

$$+1+1-1+1+1 = 3$$

$$+1+1+1-1+1 = 3$$

$$+1+1+1+1-1 = 3$$

사용할 수 있는 숫자가 담긴 배열 numbers, 타겟 넘버 target이 매개변수로 주어질 때 숫자를 적절히 더하고 빼서 타겟 넘버를 만드는 방법의 수를 return 하도록 solution 함수를 작성해주세요.

<제한사항>

주어지는 숫자의 개수는 2개 이상 20개 이하입니다.
각 숫자는 1 이상 50 이하인 자연수입니다.
타겟 넘버는 1 이상 1000 이하인 자연수입니다.

A. 구현 코드

```
#include <string>
#include <vector>

using namespace std;
int ans=0;
void dfs(vector<int> numbers, int target, int sum, int idx){
    if(idx >= numbers.size()){
        if(sum==target) ans++;
        return ;
    }

    dfs(numbers, target, sum+numbers[idx], idx+1);
    dfs(numbers, target, sum-numbers[idx], idx+1);
}

int solution(vector<int> numbers, int target) {
    dfs(numbers, target, 0,0);

    return ans;
}
```

스택 또는 재귀로 풀 수 있고, 나는 재귀를 사용하여 풀었다.

1. numbers 의 인덱스를 0 부터 탐색한다.

2. 각각 인덱스의 값을 더하고 빼보면서 끝까지 탐색
3. 인덱스가 numbers의 끝까지 탐색을 완료하면 최종 sum 값과 target 값이 같은지를 확인
4. 같다면 카운트를 올려주고 아니면 return하여 다시 탐색 시작