

그리디 알고리즘 (탐욕법)

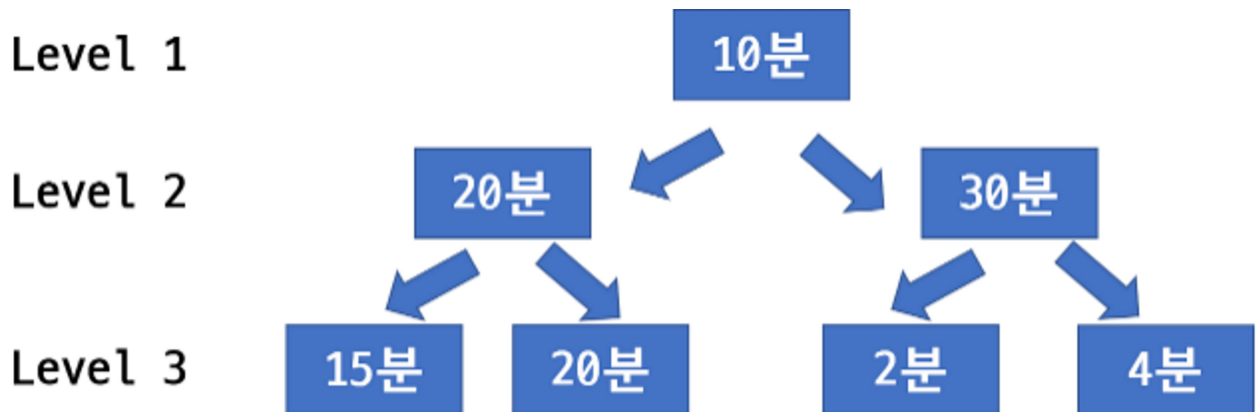
2022년 5월 13일 금요일 오후 3:37

눈 앞의 가장 큰 이익을 추구하는 기법

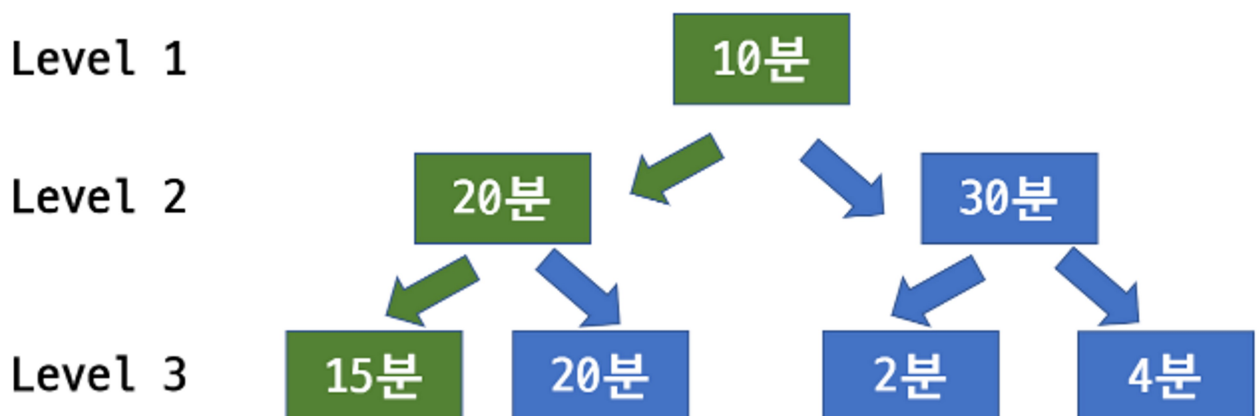
여러 경우 중 하나를 결정해야 할 때마다 그 순간 최적이라고 선택하는 방식

눈 앞의 이익만 쫓는 것은 최적의 선택이 아닐 수도 있다.

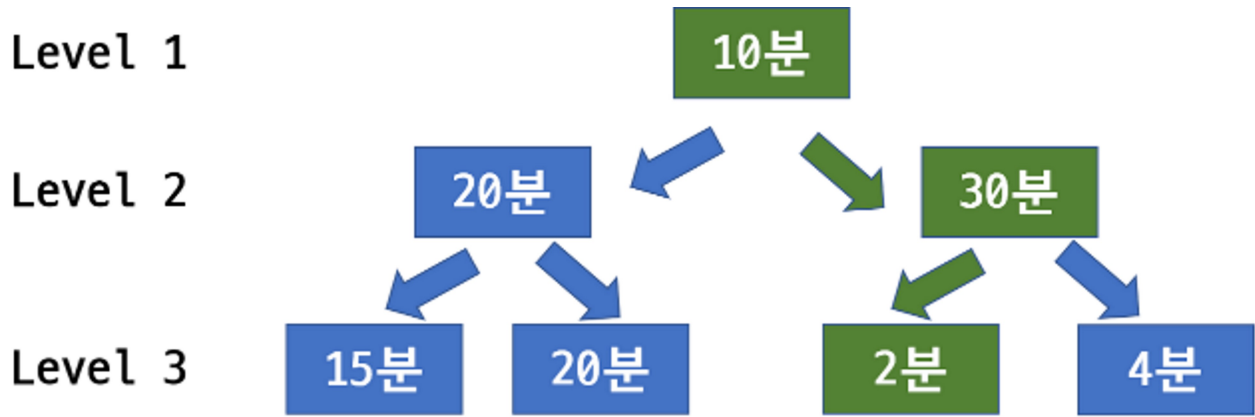
각 그림의 적힌 숫자는 문제를 푸는데 걸리는 시간이고 가장 빠른 시간안에 문제를 해결하려 한다.



그리디 알고리즘은 눈 앞의 이익을 먼저 찾기 때문에 아래 그림과 같은 상황이 발생한다. (총 45분)



하지만 최소 시간은 아래 그림과 같다 (42분)



이 알고리즘은 단순하지만 속도가 빠르다는 장점이 있다.

그리디를 사용하려면 이 방법이 최선이다 라는 근거가 있을 때만 사용해야 한다.

1. 일반적으로 최대한 적은, 최대한 많은 이라는 문구가 문제에 들어가는 경우가 많다. 최대/최소의 경우의 수를 구할것을 요구하는 문제들이다.
2. 그리디를 사용할 수 있는 조건이 주어진다. (주로 문제를 읽고 조건을 찾아야한다.)
- 3.정렬을 한 뒤 그것을 이용해 푸는 문제가 많다. (위 동전문제의 경우에도 동전을 내림차순으로 정렬한 뒤 하나 하나 사용해 나가야 할 것이다.)

/*

상근이는 요즘 설탕공장에서 설탕을 배달하고 있다. 상근이는 지금 사탕가게에 설탕을 정확하게 N킬로그램을 배달해야 한다.

설탕공장에서 만드는 설탕은 봉지에 담겨져 있다. 봉지는 3킬로그램 봉지와 5킬로그램 봉지가 있다.

상근이는 귀찮기 때문에, 최대한 적은 봉지를 들고 가려고 한다. 예를 들어, 18킬로그램 설탕을 배달해야 할 때,

3킬로그램 봉지 6개를 가져가도 되지만, 5킬로그램 3개와 3킬로그램 1개를 배달하면, 더 적은 개수의 봉지를 배달할 수 있다.

상근이가 설탕을 정확하게 N킬로그램 배달해야 할 때, 봉지 몇 개를 가져가면 되는지 그 수를 구하는 프로그램을 작성하시오.

N은 ($3 \leq N \leq 5000$)

상근이가 배달하는 봉지의 최소 개수를 출력한다. 만약, 정확하게 N킬로그램을 만들 수 없다면 -1을 출력한다.

*/

```
int solution(int n)
{
```

```
    int A = n / 5; // 5kg 사용 최대 개수
```

```

int m = 0;
int res = 0;
while (A >= 0)
{
    m = 0;
    res = 0;

    if (A > 0)
        m = n - (5 * A); // 5kg 최대 사용 개수에서 남은 개수
    else
        m = n; // 5kg 사용 개수가 없으면 최대 개수 넣어줌

    res = m / 3; // 3kg 최대 사용 개수
    m = m % 3; // 3kg 최대 사용 후 남은 개수

    if (m == 0) // 3kg 최대 사용 후 남은 개수가 없으면 retrun
        return A + res;
    A--; // 5,3kg 최대 사용 후 남은 개수가 있으면 5kg 최대 사용 개수를 하나씩 줄임
}
if (m != 0)
    return -1;

return 0;
}
int main()
{

    int ss = solution(11);

    return 0;
}

```