

기타리스트

<https://www.acmicpc.net/problem/1495>

- 첫 볼륨 : S
- 연주해야 하는 곡의 개수 N 개이고 가능한 볼륨의 범위 $0 \sim M$ 일 때 i 번 곡을 연주하기 전에 볼륨을 $v[i]$ 만큼 바꿔야 한다.

$$\begin{aligned} s &\rightarrow s + v[1] \rightarrow s + v[1] + v[2] \dots \\ &\rightarrow s + v[1] - v[2] \dots \end{aligned}$$

$$\begin{aligned} &\rightarrow s - v[1] \rightarrow s - v[1] + v[2] \dots \\ &\rightarrow s - v[1] - v[2] \dots \end{aligned}$$

👉 곡의 개수가 N 개일 때 2^N 개 이므로 완전 탐색은 불가능하다.
 $N \times M$ 다이나믹을 구성해 본다.



기타리스트

<https://www.acmicpc.net/problem/1495>

- $dp[i][j]$ = i 번 곡을 볼륨 j 로 연주할 수 있는가

$N = 3, S = 5, M = 10$

$A[1] = 5, A[2] = 3, A[3] = 7$

	0	1	2	3	4	5	6	7	8	9	10
0											
1											
2											
3											

	0	1	2	3	4	5	6	7	8	9	10
0						1					
1											
2											
3											

	0	1	2	3	4	5	6	7	8	9	10
0						1					
1	1										1
2											
3											

$A[1] = 5$

기타리스트

<https://www.acmicpc.net/problem/1495>

- $dp[i][j]$ = i 번 곡을 볼륨 j 로 연주할 수 있는가

$N = 3, S = 5, M = 10$

$A[1] = 5, A[2] = 3, A[3] = 7$

	0	1	2	3	4	5	6	7	8	9	10
0						1					
1	1										1
2				1				1			
3											

$A[2] = 3$

	0	1	2	3	4	5	6	7	8	9	10
0						1					
1	1										1
2				1				1			
3	1										1

$A[3] = 7$

- $dp[i][j] += dp[i-1][j - A[i]]$
 $+= dp[i-1][j + A[i]]$

기타리스트

<https://www.acmicpc.net/problem/1495>

```
int solve(int i, int j) {
    if (i == 0 && j == S) return true;
    if (i == 0) return false;

    if (dp[i][j] != -1) return dp[i][j];
    dp[i][j] = 0;
    if (j - A[i] >= 0) dp[i][j] += solve(i - 1, j - A[i]);
    if (j + A[i] <= M) dp[i][j] += solve(i - 1, j + A[i]);

    return dp[i][j];
}
```

```
dp[0][S] = true;
for (int i = 0; i <= N - 1; i++) {
    for (int j = 0; j <= M; j++) {
        if (dp[i][j] == false) {
            continue;
        }
        if (j - A[i + 1] >= 0) {
            dp[i + 1][j - A[i + 1]] = true;
        }
        if (j + A[i + 1] <= M) {
            dp[i + 1][j + A[i + 1]] = true;
        }
    }
}
```