



# 건강식<sup>1)</sup>

[문제] 준비된 식재료에서 가성비까지 고려된 건강식 (healthy diet)을 짜야한다. 각 재료의 단위 무게마다 기본 영양분 {단백질, 탄수화물, 지방, 비타민<sup>2)</sup>}이 포함되어 있는데, 여러분은 선택한 재료의 성분의 합이 반드시 지정된 최소 권장량을 확보하도록 해야 한다. 준비된 6개의 재료가 제시된 아래 표-1의 예제로 설명해보자.

4개 성분 최소 권장량이 (100, 70, 90, 10) 이라고 가정하자. 여러분은 준비된 6가지 재료 중에서 잘 선택하여 4개의 영양분 (단백질, 지방, 탄수화물, 비타민)의 총합이 최소 기준치인 (100, 70, 90, 10) 이상 되도록 해야 한다.

가장 간단한 방법은 넉넉하게 모든 재료를 선택하는 것이다. 그런데 이렇게 준비하면 조리과정도 복잡하고 무엇보다도 구입 비용이 커지는 단점이 생긴다. 따라서 우리는 최소 권장량 조건을 만족하면서도 선택한 재료 비용은 최소(minimum cost)가 되도록 선택해야 한다. 예를 들어 재료 {1, 3, 5}를 선택하면 영양분은 (100, 145, 130, 10)으로 조건을 만족하지만, 총 구입 가격은 270이 된다. 대신 {2, 3, 4}를 선택하면 최소 권장기준을 (110, 130, 90, 10)으로 만족시키면서도 비용은 180이 되어 앞의 가격 270짜리 재료 집합 {1, 3, 5} 보다는 더 나은 선택이 된다. 여러분에게는  $n$ 개의 식재료와 최소 권장량 기준값 ( $P_m, F_m, S_m, V_m$ )이 주어진다. 여러분은 이 최소 권장량을 만족하면서도 가장 작은 비용의 재료 집합을 찾아야 한다.

표-1. 준비된 재료와 각 재료의 영양소 성분표

재료	단백질	지방	탄수화물	비타민	가격
1	30	55	10	8	100
2	60	10	10	2	70
3	10	80	50	0	50
4	40	30	30	8	60
5	60	10	70	2	120
6	20	70	50	4	40

[입출력] 입력파일 **stdin**의 첫 줄에는 식재료의 개수  $k$  ( $5 \leq k \leq 20$ )가 주어진다. 다음 줄에는

1) healthy diet라고 하며 지중해식 식단, 그리스식( Greek diet), 오키나와식 식단 등이 전형적인 예이다.

2) 각종 비타민뿐만 아니라 mineral 성분 등을 모두 포함한 것으로 생각하면 된다.

우리가 만족시켜야 할 4개 영양성분의 최소 권장량이 4개의 정수  $mp, mf, ms, mv$ 가 주어진다. 이 값은 1이상 25000 이하의 정수이다. 이어지는  $k$ 개의 각 줄에는  $i$ 번째 식재료의 영양분과 가격이 5개의 정수  $p_i, f_i, s_i, v_i, c_i$ 가 주어진다.  $c_i$ 는  $i$ -번째 재료의 가격이다. 각 값은 0 이상 500 이하의 정수이다. 여러분은 최소 권장량 조건을 만족하는 최소비용 재료 집합을 찾아서 그 index를 오름차순으로 한 줄에 출력해야 한다.

만일 조건을 만족하는 최소 비용의 집합이 하나 이상이라면 영양분의 총합이 더 높은 조합을 선택해야 한다.<sup>3)</sup> 만일 이 총합까지 같다면 선택한 식재료 집합 index의 사전식 순서가 더 빠른 것을 출력한다. 즉 **{2,5,12}**과 **{2,6,9}**, 이 둘 모두 영양성분의 합과 비용까지 같다면 사전식 순서로 더 빠른 **{2,5,12}**를 유일한 정답으로 출력해야 한다. 만일 모든 식재료를 선택해도 기본 조건을 만족하지 못할 때는 숫자 zero(0)을 첫 줄에 출력하여 이 상황을 보고해야 한다.

stdin	stdout
6 100 70 90 10 // 최소성분 30 55 10 8 100 // $d_1$ 60 10 10 2 70 10 80 50 0 50 40 30 30 8 60 60 10 70 2 120 20 70 50 4 40 // $d_6$	2 4 6 // 선택재료 $\{i\}$
16 212 131 184 159 // 최소성분 120 31 94 54 146 // $d_1$ 320 75 65 6 264 88 142 59 80 24 10 21 5 1 3 66 30 75 71 20 1 113 129 62 108 170 57 244 270 138 39 156 84 119 108 288 208 66 179 51 249 159 19 318 112 // $d_{10}$ 85 353 242 240 403 199 14 295 352 155 6 6 1 8 3 93 99 110 124 85 100 112 54 46 30 34 144 19 119 1 // $d_{16}$	3 5 15 // 오름차순

[제한] 프로그램 이름은 **diet.{c, cpp, py}**, 수행시간은 각 test case당 **10초**를 초과할 수 없다. 사용 가능한 token은 500개이다. 만점자 중 NESPA에 기록된 수행시간 기준으로 가장 빠른 5명에게 +5의 보너스 가점이 주어진다. Python 코드 중에서 가장 빠른 1인에게도 +5의 가점을 준다.

3) 당연히 같은 값이라면 영양분의 총합이 더 많은 쪽을 선택해야겠죠.