

그리디 알고리즘(Greedy Algorithm)

탐욕 알고리즘이라고도 하며, 말 그대로 선택의 순간마다 당장 눈앞에 보이는 최고의 상황만을 쫓아 최종적인 해답에 도달하는 방식이다.

문제 :

일렬로 놓여 있는 숫자 카드에서 왼쪽 맨 끝과 오른쪽 맨 끝 카드 중 하나를 가져가는 방식으로 4개의 카드를 가져갔을 때 가져간 카드의 숫자 합의 최댓값은?

[2, 3, 7, 1, 2, 1, 5]

답 : $5 + 2 + 3 + 7 = 17$

버스

학생들이 버스에 타고 여행을 가려고 합니다.

버스는 승객을 태울 수 있는 무게 제한이 있습니다.

매개변수 `weight`에 각 학생들의 몸무게 정보가 주어지고, `limit`에 버스가 태울 수 있는 총 승객의 무게가 주어지면 버스에 탈 수 있는 최대인원수를 구하여 반환하는 프로그램을 작성하세요.

입출력 예:

weight	limit	answer
[300, 100, 230, 120, 90, 150, 60]	700	5
[50, 90, 70, 120, 300, 200, 150, 180, 190]	1000	7
[70, 90, 100, 80, 60, 75, 73, 85, 120, 110, 200]	800	9
[50, 90, 100, 130, 140, 120, 130, 120, 150, 160, 140, 170]	1000	8
[100, 110, 50, 50, 60, 70, 50, 55, 57]	350	6

제한사항:

- `weight`의 길이는 100,000을 넘지 않습니다.
- $50 \leq \text{weight}[i] \leq 300$
- $100 \leq \text{limit} \leq 100,000,000$

최대 사과와 개수

여러 종류의 사과박스가 있습니다.

각 박스의 종류에 따라 박스에 담겨있는 사과의 개수가 다릅니다.

트럭에 박스를 싣으려고 합니다. 트럭에 박스를 싣을 수 있는 최대 개수 제한이 있습니다.

매개변수 box에 각 박스 종류의 정보가 주어지고, limit에 트럭의 싣을 수 있는 박스의 최대 개수가 주어지면 트럭에 싣을 수 있는 사과의 최대 개수를 반환하는 프로그램을 작성하세요.

입출력 예:

box	limit	answer
[[2, 20], [2, 10], [3, 15], [2, 30]]	5	115
[[1, 50], [2, 20], [3, 30], [2, 31], [5, 25]]	10	302
[[3, 40], [5, 20], [5, 70], [1, 80], [5, 30], [3, 35]]	15	745
[[2, 70], [5, 100], [3, 90], [1, 95]]	8	775
[[80, 20], [50, 10], [70, 15], [70, 30], [80, 70], [90, 88], [70, 75]]	500	23920

제한사항:

- box의 길이는 100,000을 넘지 않습니다.
- box[i][0]은 i 종류 박스의 개수, box[i][1]은 i 종류의 박스 한 개에 들어 있는 사과의 개수입니다. 서로 다른 종류의 박스라도 담아 있는 사과의 개수는 같을 수 있습니다.
- $1 \leq \text{box}[i][0] \leq 100$, $1 \leq \text{box}[i][1] \leq 100$
- $1 \leq \text{limit} \leq 10,000,000$

선긋기

한 번의 선긋기는 수직선상의 한 점에서 다른 한 점까지 선을 긋는 것입니다.

선을 그을 때는 이미 선이 있는 위치에 겹쳐서 그을 수도 있습니다.

여러번 그은 곳과 한 번 그은 곳의 차이는 없습니다.

수직선은 0번 지점부터 m번 지점까지의 길이를 갖고 있습니다.

매개변수 nums에 각각의 선긋기 정보가 주어지면 0번 지점부터 m번 지점까지 연속적인 선이 그어지도록 하기 위한 선긋기 최소횟수를 반환하는 프로그램을 작성하세요.

모든 입력은 0번 지점부터 m번지점까지 연속적인 선이 그어집니다.

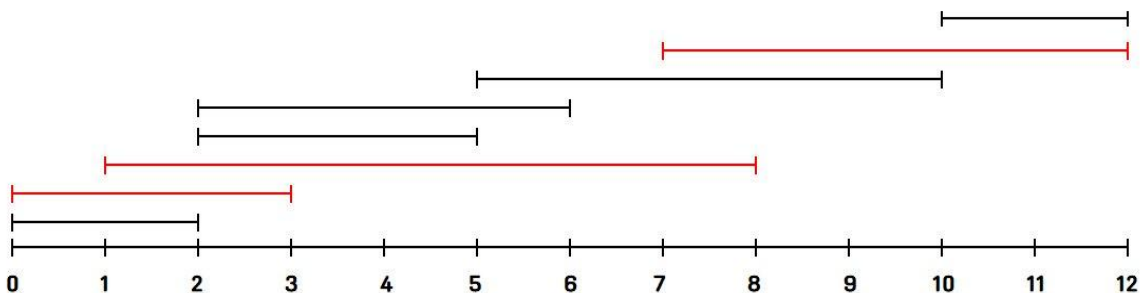
입출력 예:

m	nums	answer
12	[[5, 10], [1, 8], [0, 2], [0, 3], [2, 5], [2, 6], [10, 12], [7, 12]]	3
15	[[1, 10], [0, 8], [0, 7], [0, 10], [12, 5], [0, 12], [8, 15], [5, 14]]	2
20	[[3, 7], [5, 8], [15, 20], [0, 5], [7, 14], [3, 10], [0, 8], [13, 18], [5, 9]]	4
30	[[5, 7], [3, 9], [2, 7], [0, 1], [0, 2], [0, 3], [19, 30], [8, 15], [7, 12], [13, 20]]	5
25	[[10, 15], [15, 20], [5, 15], [3, 16], [0, 5], [0, 3], [12, 25]]	3

제한사항:

- $3 \leq m \leq 10,000$
- nums의 길이는 100,000을 넘지 않습니다.
- $\text{nums}[i][0]$ 은 i번째 선긋기의 시작 점, $\text{nums}[i][1]$ 은 i번째 선긋기의 끝점입니다.
- $0 \leq \text{nums}[i][0] < \text{nums}[i][1] \leq 10,000$

예제 1번 설명:



카드 점수

N개의 카드가 일렬로 놓여져 있습니다. 각 카드에는 숫자가 적혀있습니다.

현수는 카드가 일렬로 놓여진 줄의 양 끝 즉 왼쪽 맨 끝카드와 오른쪽 맨 끝 카드 둘 중 하나를 가져갈 수 있습니다. 현수는 양 끝에서 가져가는 방식으로 k개의 카드를 가져갈 수 있습니다. 그리고 가져간 카드에 적혀진 숫자의 총합이 현수가 얻는 점수입니다.

일렬로 놓여진 각 카드의 숫자가 매개변수 nums에 주어지고, 현수가 가져갈 수 있는 카드의 개수 k가 주어지면 현수가 얻을 수 있는 최대점수를 반환하는 프로그램을 작성하세요.

입출력 예:

nums	k	answer
[2, 3, 7, 1, 2, 1, 5]	4	17
[1, 2, 3, 5, 6, 7, 1, 3, 9]	5	26
[1, 30, 3, 5, 6, 7]	3	38
[1, 2, 15, 3, 6, 7, 8, 9]	5	35
[12, 5, 6, 12, 34, 35, 13, 3, 7, 8, 9]	7	117

제한사항:

- nums의 길이는 300,000을 넘지 않습니다.
- nums의 원소는 100을 넘지 않는 자연수입니다..
- $2 \leq k < \text{nums의 길이}$

입력예제 1번 설명 :

왼쪽에서 2, 3, 7, 오른쪽에서 5 이렇게 4개를 가져가면 최대가 됩니다. $2+3+7+5=17$ 입니다.