

문제. 막내

$N \times N$ 크기의 집에 네 명의 형제가 있다. 막내는 형들을 좋아해서, 아침에 일어나면 첫째, 둘째, 셋째 형에게 차례대로 아침 안부를 묻는다. 초기에 네 형제는 모두 서로 다른 칸에 존재한다.

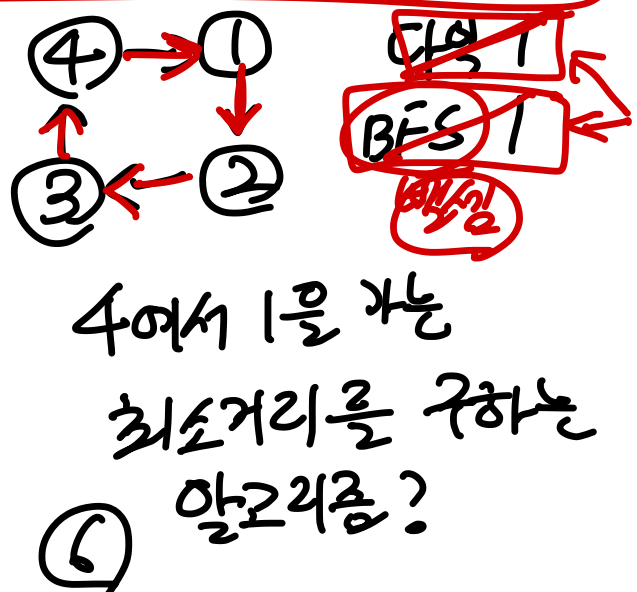
구체적으로 막내는 초기에 자신이 있는 위치에서 첫째, 둘째, 셋째의 위치를 차례대로 방문한 뒤에 원래의 위치로 돌아온다. 막내는 현재 위치에서 1초마다 인접한 상, 하, 좌, 우 위치로 이동할 수 있으며, 이동하는 과정에서는 한 칸에 여러 명이 존재할 수 있다 또한, 특정한 형에게 안부를 묻기 위해서는 해당 형이 있는 동일한 격자 위치까지 도달해야 한다.

막내가 모든 형들에게 차례대로 도달하여 안부를 물어 본 뒤에 다시 원래의 위치로 돌아오기까지 걸리는 최소 시간을 계산하는 프로그램을 작성하여라.

예를 들어 $N = 6$ 인 상황을 확인해 보자. 격자의 값 중에서 0은 아무 것도 없는 빈 칸을 의미하며, 1은 첫째가 있는 칸, 2는 둘째가 있는 칸, 3은 셋째가 있는 칸, 4는 막내가 있는 칸을 의미한다.

	1	2	3	(형, 열)	1
1	0	0	0	0	0
2	0	4	0	0	0
3	0	(2,3)	0	2	0
4	0	0	0	0	0
5	0	0	0	0	1
6	0	3	0	0	0

(5,2) + (5,2)



각 칸을 (행, 열)의 형태로 위치를 나타내고, 행과 열의 번호는 0부터 시작한다고 가정하자. 현재 상황에서는 막내가 $(1, 1) \rightarrow (4, 4) \rightarrow (2, 3) \rightarrow (5, 1) \rightarrow (1, 1)$ 순서대로 이동하는 것이 최선이므로, 총 $18(6 + 3 + 5 + 4)$ 초의 시간이 소요된다. 따라서 본 예시에서의 정답은 18이다.

입력 조건

$4 \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 4$

이동 순서 고정됨

이동 순서를 알아서
구하려고 하는 것 → 최소를 구하라

2를 → 다 해봄

정답의 수가 4 이상

지도를 못가는데 칸이 $\frac{0}{2}$

지도를 못가는데 칸이 $\frac{0}{2}$

→ 거리를 구하는 방법: BFS

↓ ↓ ↓
 $4 \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 4$
 $4 \rightarrow 1 \rightarrow 3 \rightarrow 2 \rightarrow 4$
 $4 \rightarrow 2 \rightarrow 1 \rightarrow 3 \rightarrow 4$
 $4 \rightarrow 2 \rightarrow 3 \rightarrow 1 \rightarrow 4$
 $4 \rightarrow 3 \rightarrow 1 \rightarrow 2 \rightarrow 4$
 $4 \rightarrow 3 \rightarrow 2 \rightarrow 1 \rightarrow 4$

가장 먼저 집의 크기 N 이 주어진다. N 은 3 이상 100 이하의 자연수다.

이어서 행과 열의 길이가 각각 N 인 2차원 배열 형태의 집 정보 배열 *house*가 주어진다. 격자의 값 중에서 0은 아무 것도 없는 빈 칸을 의미하며, 1은 첫째가 있는 칸, 2는 둘째가 있는 칸, 3은 셋째가 있는 칸, 4는 막내가 있는 칸을 의미한다.

출력 조건

막내가 모든 형들에게 차례대로 도달하여 안부를 물어 본 뒤에 다시 원래의 위치로 돌아오기까지 걸리는 최소 시간을 반환하여라.

입출력 예시

N	<i>house</i>	정답
6	[[0, 0, 0, 0, 0, 0], [0, 4, 0, 0, 0, 0], [0, 0, 0, 2, 0, 0], [0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 1, 0], [0, 3, 0, 0, 0, 0]]	18
4	[[0, 1, 0, 0], [0, 2, 0, 0], [0, 0, 0, 0], [0, 4, 0, 3]]	10

해설 1. 막내

본 문제는 기초 프로그래밍 문법을 활용하여 해결할 수 있다. 문제에서 주어지는 N 의 값이 최대 100이므로, 최대 10,000개의 각 칸에 대하여 확인하여 첫째, 둘째, 셋째, 막내의 위치를 (행, 열)의 형태로 기록한다. 이후에 두 명의 거리를 계산할 때는 **행의 차이와 열의 차이의 합**을 계산하면 된다. 결과적으로 본 문제는 **기초 문법, 시뮬레이션 및 완전 탐색 유형**에 속한다.

문제. 세금

지난 해 한 국가에서는 N 명이 세금을 납부했다. 세금을 관리하는 직원은 지난 해 사람들이 세금을 낸 정보를 수집하는 과정에서 K 번의 정보 수집을 진행하고자 한다. 각 정보 수집에서는 X 와 Y 두 가지 변수가 사용되는데, 이는 납부한 세금의 금액이 X 이상 Y 이하인 사람의 수를 계산하는 작업을 의미한다.

각 K 번의 정보 수집에 대하여, 납부한 세금의 금액이 X 이상 Y 이하인 사람의 수를 차례대로 계산하는 프로그램을 작성하여라.

예를 들어 $N = 5$ 명이 납부한 세금의 금액이 다음의 표와 같다고 해보자.

번호	①	②	③	4	5
금액	7	2	3	5	2

이후에 $K = 3$ 번의 정보 수집에 대한 정보가 다음과 같다고 가정하자. 정보 수집을 위해 사용되는 변수는 $[X, Y]$ 배열 형태로 주어진다.

번호	1	2	3
수집 내용	<u>$[1, 100]$</u>	<u>$[3, 5]$</u>	<u>$[2, 2]$</u>

현재 예시에서 각 정보 수집에 대한 정답 결과는 다음과 같다. 현재 예시에서 정보 수집 결과는 차례대로 5명, 2명, 2명이다.

번호	1	2	3
수집 결과	5	2	2

입력 조건

가장 먼저 세금을 낸 사람의 수 N 과 정보 수집 횟수 K 가 주어진다. 이때 N 과 K 는 모두 4 이상 100,000 이하의 자연수다.

사람 N 명, 정보 수량 K 번

$$4 \leq N, K \leq (1000)$$

$$\text{하루씩 하기 : } \underline{O(N)}$$

$$\text{이분 탐색 : } O(\lg N)$$

$$\rightarrow \underline{K\text{번 하면 : } O(NK)} \rightarrow O(K \lg N)$$

$$1000 \times 1000 = 1000000$$

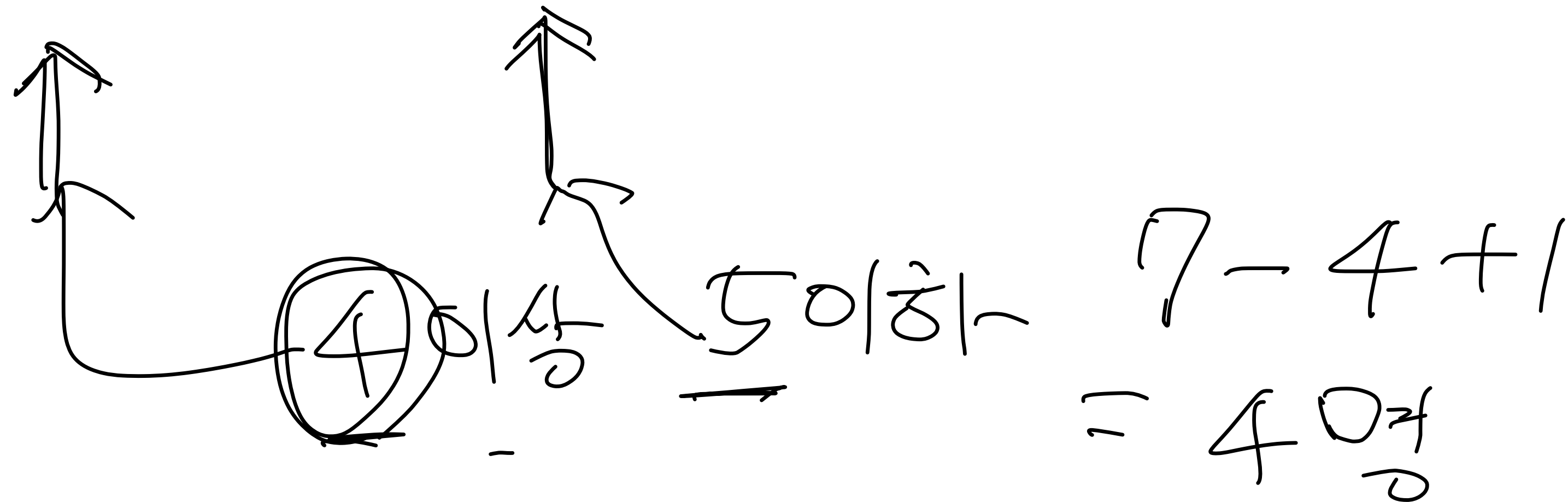
$$N \lg N + K \lg N$$

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

정렬된 배열

수의 위치는

몇몇 알고리즘?



4 이상 7 이하 $7 - 4 + 1$

이진 탐색

Binary Search

②

$\lg N$

상한 (upper bound) : 큰 수 중 최대 수
 하한 (lower bound) : 크거나 같을 수 중의 최대 수

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

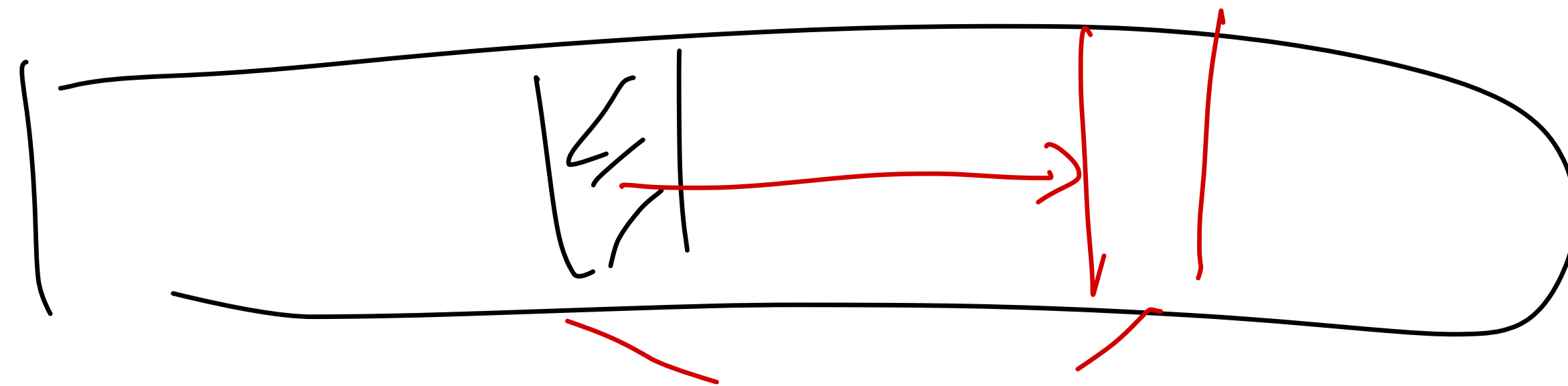
4이상 5이하 : 5의 상한 4의 하한

2이상 7이하 : 7의 상한 2의 하한
 $8 - 4 = 4$
 $8 - 2 = 6$

Array, Binary Search()

↓
return

중요한 것만



이어서 N 명의 사람들이 낸 세금에 대한 정보가 담긴 배열 arr 가 주어진다. 이때, 각 사람이 납부한 세금의 금액은 1 이상 1,000,000,000 이하의 자연수다.

이어서 K 회의 정보 수집 작업에 대한 정보가 담긴 배열 $queries$ 가 주어진다. 구체적으로 행의 크기가 N 이고, 열의 크기가 2인 형태의 2차원 배열이다. 각 정보 수집 작업에서의 X 와 Y 는 모두 1 이상 1,000,000,000 이하의 자연수로, 이때 X 는 Y 보다 작거나 같다.

출력 조건

K 번의 정보 수집에 대하여, 납부한 세금의 금액이 X 이상 Y 이하인 사람의 수를 차례대로 계산하여 1차원 배열 형태로 반환한다.

입출력 예시

N	K	arr	$queries$	정답
5	3	[7, 2, 3, 5, 2]	[[1, 100], [3, 5], [2, 2]]	[5, 2, 2]
6	4	[9, 8, 1, 5, 5, 5]	[[5, 8], [3, 10], [1, 2], [5, 9]]	[4, 5, 1, 5]

해설 2. 세금

본 문제는 전형적인 **정렬 및 이진 탐색 유형**의 문제다. 최대 100,000개의 정보 수집 작업이 수행될 수 있다는 점에서 선형 탐색으로는 시간 제한을 통과할 수 없다. 따라서 이진 탐색을 수행하기 위한 사전 조건으로, 먼저 N 명이 납부한 세금 값에 대하여 오름차순 정렬을 수행한다. 이후에 매 정보 수집 작업에 대하여 이진 탐색을 활용하면 값의 범위가 $[X, Y]$ 에 속하는 원소의 개수를 구하기 위해 **로그(log) 시간**이 소요된다. 결과적으로 K 번의 작업에 대하여 매번 이진 탐색을 수행하여 문제를 해결할 수 있다. 따라서 정렬을 위해 시간 복잡도 $O(N \cdot \log N)$ 가 요구되며, 전체 쿼리에 대하여 시간 복잡도 $O(K \cdot \log N)$ 가 요구되므로, 최종적인 시간 복잡도는 $O(N \log N + K \log N)$ 이다.

- Python3 정답 코드 예시

```
from bisect import bisect_left, bisect_right
```

문제. 협동력 점수

K 가지
는 방법의 수: $\frac{1}{K} \frac{2}{K} \frac{3}{K} \dots \frac{N}{K}$

N 명의 학생을 K 개의 그룹으로 나누고자 한다. 각 학생은 협동력 점수를 가지고 있으며, 단 하나의 그룹에만 속할 수 있다. 모든 학생은 반드시 각자 특정한 하나의 그룹에 배치된다.

모든 학생에 대한 그룹 배치가 끝나고 나면, 각 그룹의 전투력을 계산할 수 있다. 이때 특정한 그룹의 전투력은 그룹에 포함된 각 학생에 대하여 (해당 그룹에 속한 구성원 수 \times 해당 학생의 협동력 점수)을 계산하여 모두 더한 값이다. 어떤 학생의 협동력 점수는 음수(-)일 수 있으며, 결과적으로 특정한 그룹의 전투력은 음수가 될 수도 있다.

이때, N 명의 학생으로 K 개의 그룹을 구성하는 모든 경우를 고려했을 때 가장 전투력이 높은 그룹과 전투력이 낮은 그룹의 (전투력 차이)의 최솟값을 구하여라. 단, 모든 K 개의 그룹에 최소한 1명의 학생은 배치되어야 한다.

예를 들어 $N = 4$ 명의 학생들의 협동력 점수가 다음의 표와 같으며, 그룹의 수 $K = 2$ 라고 하자.

학생 번호	1	2	3	4
협동력 점수	-1	0	3	-2

가장 먼저, 다음과 같이 두 그룹을 결성했다고 가정하자.

그룹 번호	1	2
학생 번호 구성	[1, 2]	[3, 4]

이때 전투력은 다음과 같이 계산되므로, 두 그룹의 전투력 차이는 4이다

- (1) 1번 그룹의 전투력: $-2 = 2 \times (-1) + 2 \times 0$
- (2) 2번 그룹의 전투력: $2 = 2 \times 3 + 2 \times (-2)$

혹은 다음과 같이 두 그룹을 결성했다고 가정하자.

4

$$\begin{aligned} 2 \times -1 &= -2 \\ 2 \times 0 &= 0 \\ \hline 2 \times 3 &= 6 \\ 2 \times -2 &= -4 \end{aligned} \quad \begin{aligned} &-2 \\ &0 \\ &6 \\ &-4 \end{aligned} \quad \begin{aligned} &-2 \\ &2 \end{aligned}$$

그룹 번호	1	2
학생 번호 구성	[2]	[1, 3, 4]

이때 전투력은 다음과 같이 계산되므로, 두 그룹의 전투력 차이는 0이다.

(1) 1번 그룹의 전투력: $0 = 1 * 0$

(2) 2번 그룹의 전투력: $0 = 3 * (-1) + 3 * 3 + 3 * (-2)$

따라서, 현재 입력 예시에서의 최적의 해는 0이다.

$\eta = 823543$

입력 조건

가장 먼저 학생의 수 N 이 자연수로 주어진다. N 은 2 이상 7 이하의 자연수다.

이어서 그룹의 수 K 가 자연수로 주어진다. K 는 2 이상 N 이하의 자연수다.

이후에 각 학생의 협동력 점수 정보가 담긴 배열 arr 가 주어진다. 각 학생들의 협동력 점수는 -10 이상 10 이하의 정수이다.

출력 조건

N 명의 학생으로 K 개의 그룹을 구성하는 모든 경우를 고려했을 때, 가장 전투력이 높은 그룹과 전투력이 낮은 그룹의 전투력 차이의 최솟값을 반환한다.

입출력 예시

N	K	arr	정답
4	2	[-1, 0, 3, -2]	0
5	4	[2, 1, 1, 2, 3]	2

해설 3. 협동력 점수