

# 백준 13164- 행복 유치원

<input checked="" type="checkbox"/> Check	<input type="checkbox"/>
# 문제번호	13164
≡ 유형	그리디
📅 Date	@2022년 4월 2일

13164번: 행복 유치원

행복 유치원 원장인 태양이는 어느 날  $N$ 명의 원생들을 키 순서대로 일렬로 줄 세우고, 총  $K$ 개의 조로 나누려고 한다. 각 조에는 원생이 적어도 한 명 있어야 하며, 같은 조에 속한 원생들은 서로 인접해 있어야 한다.

<https://www.acmicpc.net/problem/13164>

BAEKJOON  
ONLINE JUDGE

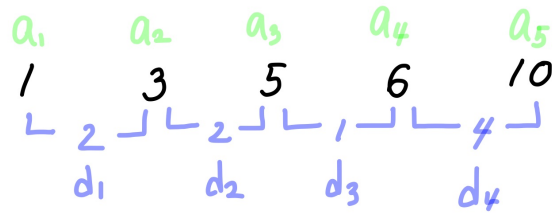
## 1. 문제 요약

- $N$  명의 원생들을 키 순서대로 줄을 세운다. (오름차순)
- $K$ 개의 조로 나누어 조마다 단체 티셔츠를 맞춘다.
  - 각 조 원생은 최소 한 명 이상은 있어야 한다.
  - 티셔츠의 비용 = 각 조에서 가장 큰 원생의 키 - 가장 작은 원생의 키
- 단, 티셔츠 만드는 비용의 합 최소로 할 것

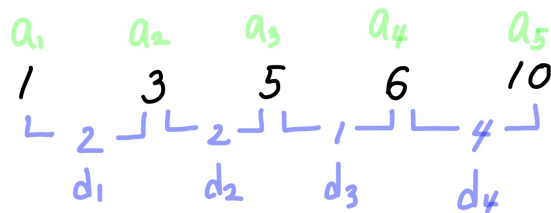
## 2. 제한 사항

- 원생의 수:  $1 \leq N \leq 300,000$
- 조의 수:  $1 \leq K \leq N$
- 원생의 키는  $10^9$ 를 넘지 않는 자연수

## 3. 아이디어



- $a_5 - a_3 = d_3 + d_4$
- $a_4 - a_2 = d_2 + d_3$
- $a_5 - a_2 = d_2 + d_3 + d_4$



- Q1. 1개의 조만 만든다면?  $d_1 + d_2 + d_3 + d_4$   $5 - 1$
- Q2. 2개의 조만 만든다면?  $d_1 + d_2 + d_3$   $5 - 2$
- Q3. 3개의 조만 만든다면?  $d_2 + d_3$   $5 - 3$
- $\vdots$   
 $N - K$

- 위 그림에서 보면 **K개의 조**를 만들기 위해서는 **N-K개의 차**를 더해야 한다는 것을 알 수 있다.
- 문제에서 요구하는 것은 **최소의 티셔츠 비용**이므로 **N-K개의 차**를 더한 값이 **최소**이면 된다.

⇒ N개의 원소들의 차를 구하고, 이 차를 오름차순으로 정렬하여 N-K개 선택하기로 한다.

## 4. 코드

```
import java.util.Arrays;
import java.util.Scanner;

/*
 * 22. 04. 02. 토
 * 백준 13164 - 행복 유치원
 *
 */

public class Main_13164_Greedy {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        int N = sc.nextInt(); // 원생의 수
        int K = sc.nextInt(); // 만들 조의 수

        /**
         * 구해야 할 것 : 가장 키가 큰 원생과 가장 키가 작은 원생의 키 차이
         */

        int[] heights = new int[N]; // 원생의 키
        int[] diffs = new int[N - 1]; // 각 원생 사이의 키 차이
        for (int i = 0; i < N; i++) {
            heights[i] = sc.nextInt();
        }
        // ===== input end =====

        for (int i = 1; i < N; i++) { // O(2N) = O(N)
            diffs[i - 1] = heights[i] - heights[i - 1];
        }
        Arrays.sort(diffs);

        int result = 0;
        for (int i = 0; i < N - K; i++) { // O(N-K)
            result += diffs[i];
        }

        System.out.println(result);

        sc.close();
    }
}
```

- 시간 복잡도 =  $O(N)$

- 각 원소의 차를 구하는 복잡도 ( $O(2N)$ )

- 첫 번째 원소와 두 번째 원소를 제외하면, 모두 2번 정도 방문한다.

- 각 차를 구한 뒤, 차를  $N-K$  개 더한다. ( $O(N-K)$ )

$\Rightarrow O(2N) + O(N-K) \asymp O(N)$