

2021-1

11주차 문제풀이  
Sangmyung ICPC Team

LEE JIWOO

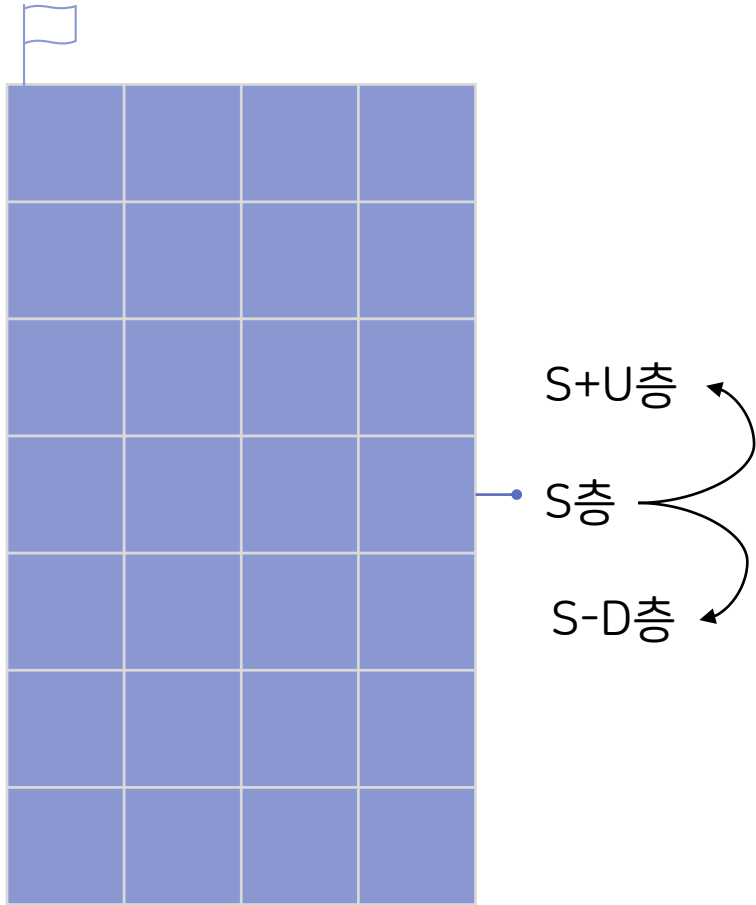
# StartLink





S층에서 G층으로 가기 위해 눌러야 하는 버튼의 수의 최소값

## Step1 - N층에 가기 위한 최소 버튼 Push 횟수는?



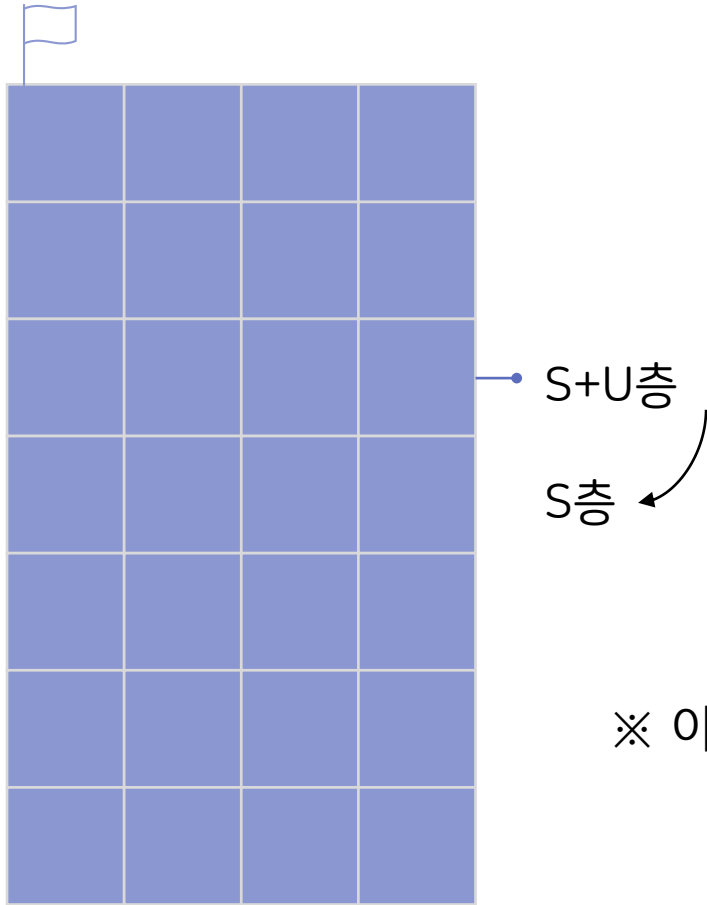
\*U == 1, D == 1일 때

$$\begin{aligned} \min\_push[S+U] &= 1 \text{ (번)} \\ \min\_push[S] &= 0 \text{ (번)} \\ \min\_push[S-D] &= 1 \text{ (번)} \end{aligned}$$

Curved arrows with '+1' connect the middle equation to the top and bottom equations.

$$\begin{aligned} \gg \quad \min\_push[N+U] &= \min\_push[N] + 1 \\ \min\_push[N-D] &= \min\_push[N] + 1 \end{aligned}$$

## Step2 – 이미 방문한 적 있는 층의 최소 버튼 Push 횟수는?



$\text{min\_push}[S+U] = 1 \text{ (번)}$

$\text{min\_push}[S] = 2 \text{ (번)} \dots ???$

※ 이미 방문한 층은 이미 해당 층에 대한 최소 버튼 Push 횟수가 구해짐

≫ 해당 층에 방문했는지 확인할 수 있어야 함  
 $\text{min\_push}[N] == -1$ : 방문X  
 $\text{min\_push}[N] \neq -1$ : 방문O

\*U == 1, D == 1일 때

1. 현재 N층에서 갈 수 있는 모든 층을 탐색
2. 이미 방문한 층이라면 Pass
3. 아니라면  $\text{min\_push}[N] + 1$  대입

→ BFS를 이용하자!

### Step3 - 구현

#### 입력

```
from sys import stdin

f, s, g, u, d = map(int, stdin.readline().split())
```

min\_push[] 초기화 - s층: 0, 나머지 층: -1

```
min_push = [-1 for _ in range(f+1)]
min_push[s] = 0
```

#### 큐 생성 및 s(시작 층) 삽입

```
queue = list()
queue.append(s)

next_floor = [u, -d]
```

다음으로 갈 수 있는 층을 구하기 위한 next\_floor

### Step3 - 구현

큐에 요소가 있는 동안 반복

```
while queue:
    curr_floor = queue.pop(0)

    if curr_floor == g: 목적지 층(G)에 달하면 반복문 탈출
                        (b/c 목적지 층에 대한 min_push값이 이전 단계에서 구해졌음)
        break

    for next in next_floor: 현재 층에서 갈 수 있는 다음 층 탐색
        if (1 <= curr_floor+next <= f) and (min_push[curr_floor+next] == -1):
            min_push[curr_floor+next] = min_push[curr_floor]+1
            queue.append(curr_floor+next) 건물 층 수를 벗어나는지 확인 후, 방문 여부 확인

print(min_push[g]) if min_push[g] != -1 else print("use the stairs")
```

# Thanks

