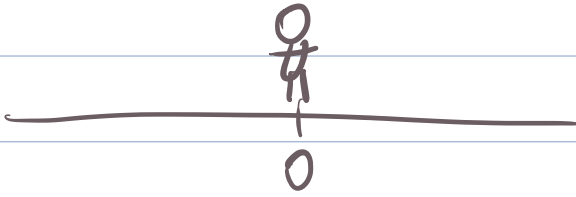


도서관

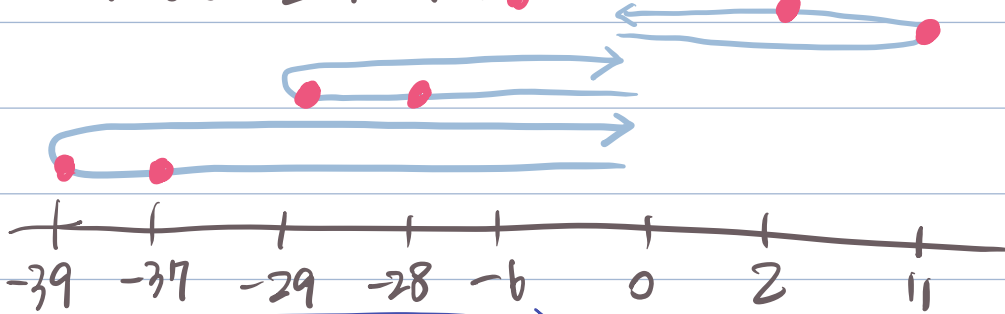


[N 책갯수

M 행번에 최대 M권

7

2



← 가장 앞 행번은 후 정렬

0 2 6 11 28 29 37 39

↓
계산연산부터

$$39 + 29 + 11 + 6 = 85$$

$$\begin{array}{r} \text{ } \\ \hline 40 \\ \hline 45 \end{array}$$

$$39 + 29 + 11 + 6$$

x2

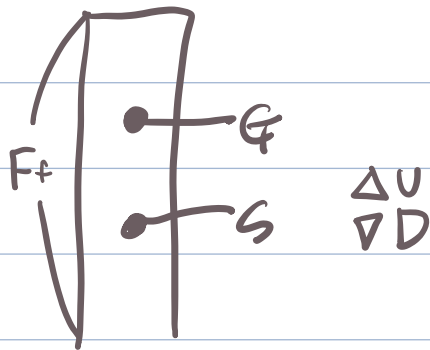
$$40 + 39 = 79$$

$$\begin{array}{r} 79 \\ \times 2 \\ \hline 158 + 6 \\ \hline = 164 \end{array}$$

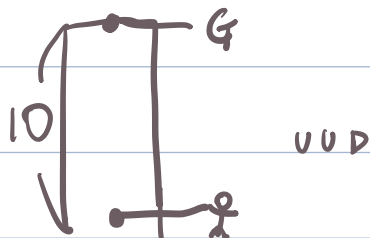
$$(11 + 37 + 28) \times 2 + 39$$

$$\begin{array}{r} 65 \\ \times 11 \\ \hline 76 \\ \times 2 \\ \hline 152 \\ \hline 39 \\ \hline 191 \end{array}$$

스타팅링크



1) 10^F 1^S 10^G 2^U 1^D, visited[]



S → G 로 가는 최단경로 ⇒ BFS
visited

처음 S. 현재 C. 다음 $\begin{cases} C+U \\ C-D \end{cases} \leq F$
(방문 안)

bfs(f, s, g, u, d)

bfs(int f, s, g, u, d)

Queue<Integer> q = new LinkedList<Integer>();

q.add(s) ; 처음 위치

v[s] = 1 ; 방문처리 X

while (!q.isEmpty()) {

int c = q.poll(); 현재 위치.

if ① $c \neq g$. 현재 위치가 가야 할 곳이면 stop

else ② 방문

if ($c+u \leq f$ && $c-d \geq 1$) {

visited[c+u] = !visited[c]

~~q~~ q.add(c+u)

q.add(c-d)

}

if (v[g] == 0)

syso.("use the stairs");

&& visited[c+u] == 0

&& visited[c-d] == 0