

개강 기념 ICTS 미니 대회

# 문제 1. 징검다리 건너기

- 다이나믹 프로그래밍

- 처음 푼 사람:
- 가장 빠른 풀이를 한 사람:

# 문제 1. 징검다리 건너기

- 매우 큰 점프 유무, 현재 밟고 있는 돌을 기준으로 2차원 배열 만든다.
  - 2차원 배열의 마지막 원소 중 최솟값을 고르면 된다
- 
- 별해:  $N$ 이 작기 때문에 재귀로 모든 경우의 수를 확인하는 것도 가능

# 문제 2. 사과 나무

- 그리디
- 수학

- 처음 푼 사람:
- 가장 빠른 풀이를 한 사람:

## 문제 2. 사과 나무

- 나무 하나를 1만큼 성장시키는 물뿌리개를 A, 다른 물뿌리개를 B라 하자
- 우리는 A,B를 동시에 사용해야 하므로 주어진 입력 값처럼 나무를 기를 수 있는지 판별하려면, A와 B를 같은 횟수로 사용했다는 것을 확인 하면 됨.
- 한 나무의 크기 N이 홀수인 경우, A를 1번, B를  $N // 2$  번 사용하여 만들 수 있다는 것을 확인 가능하다.
- 또한, B를 한 번 사용한 것은 A를 2번 사용한 효과를 가졌다는 것을 확인 할 수 있다.
- 각 나무를 확인 하여 A를 사용한 총 횟수 a, B를 사용한 총 횟수 b를 구하면 다음과 같은 사실을 확인 할 수 있다.
- $a > b$  인 경우, 어떠한 경우의 수로도 입력 값처럼 나무를 키울 수 없다는 것을 확인 가능
- $a < b$  인 경우,  $b - i = a + 2*i$  를 확인해보자.

# 문제 3. 숫자놀이

- 문자열
- 정렬
- 구현

- 처음 푼 사람:
- 가장 빠른 풀이를 한 사람:

# 문제 3. 숫자놀이

- 숫자를 문자열로, 문자열을 숫자로 바꿔주는 함수 (또는 자료구조) 를 활용.
- 숫자를 문자열로 만든 후
- 문자열들을 정렬
- 정렬된 결과를 숫자로 출력

# 문제 4. 너무 시시했다

- 수학
- 비트 마스킹
- 에드 혹

- 처음 푼 사람:
- 가장 빠른 풀이를 한 사람:



## 문제 4. 너무 시시했다

- $A + B = A \wedge B + ((A \& B) \ll 1)$ 임을 발견해야함...

올림( $((A \& B) \ll 1)$ )	1	0	0	0	1	0
A	0	1	0	1	0	1
B	0	1	1	0	0	1
$A + B$	1	0	1	1	1	0
$A \wedge B$	0	0	1	1	0	0

- 즉  $x - y$ 를 하여  $A \& B$ 를 구하고,  $A \wedge B$ 와 비교하며 가능한 경우의 수를 카운팅하면 된다.
- 힌트)  $A \& B == 1$ 일 때 경우의 수는 1가지 이지만  $A \wedge B == 1$ 일 때 경우의 수는 2가지이다.

# 문제 5. 가희와 프로세스 1

- 우선순위 큐

- 처음 पून 사람:
- 가장 빠른 풀이를 한 사람:

# 문제 5. 가희와 프로세스 1

- (우선순위,  $-1 \times \text{인덱스}$ ) 을 기준으로하는 우선순위 큐를 생성.
- 1초 동안 프로세스를 실행하고 나머지 프로세스들의 우선 순위가 1 상승한데, 이는 반대로 말하면 지금 실행 중인 프로세스의 우선순위가 1 떨어진다고 생각할 수 있음.

# 문제 6. 가희와 프로세스 2

- 정렬
- 이분탐색 비스무리
- 에드 혹

- 처음 푼 사람:
- 가장 빠른 풀이를 한 사람:

## 문제 6. 가희와 프로세스 2

- 1초 동안 프로세스를 실행하고 나머지 프로세스들의 우선 순위가 1 상승한다는데, 이는 반대로 말하면 지금 실행 중인 프로세스의 우선순위가 1 떨어진다고 생각할 수 있음.
- 즉, 해당 프로세스가 1초씩 실행 되므로, 프로세스가  $n$ 번 실행되면, 우선순위도  $n$ 번 내려감.
  - 우선순위가  $n \rightarrow n-1 \rightarrow n-2 \rightarrow \dots \rightarrow n-t+1$ 로 바뀜. ( $n-t$ 일 때면 프로세스는 완전히 실행되었기에 더 이상 큐에 안 들어감)
- 프로세스는 우선순위가 가장 높은 프로세스를 먼저 시행하기에, 우선순위가  $n-1$ 인 프로세스를 실행할 때, 우선순위가  $n$ 인 프로세스 모두를 1번씩 실행해야됨.
- 즉, 우리는  $t$ 초일 때, 어떤 우선순위의 프로세스를 실행하는지 알아낼 수 있다!
- [해답 링크](#)