

간식 분배 - 풀이

작성자: 박영우

부분문제 1

모든 학생은 정확히 하나의 간식을 좋아하므로, 학생이 방에 들어갔을 때 최대 하나의 간식을 가져간다.

만약 여러 학생이 공통으로 좋아하는 간식이 있다면, 그중 나중에 방에 들어가는 학생은 아무 간식도 가져갈 수 없다. 따라서 이 경우에는 목표를 달성할 수 없다.

그렇지 않다면 모든 학생이 서로 다른 간식을 좋아하므로, 학생들을 어떤 순서로 방에 들여보내더라도 모든 학생이 정확히 하나의 간식을 가져간다.

부분문제 2

이 부분문제에서는 목표를 달성하는 것이 가능할 경우, $1, 2, \dots, N$ 번 학생들을 이 순서대로 방에 들여보내도 목표를 달성할 수 있다.

따라서 방 안에 남은 간식을 관리하면서, 각 학생이 순서대로 방에 들어갈 때 가져가는 간식의 개수를 세면 된다.

부분문제 3

이 부분문제에서 학생들이 방에 들어가는 경우의 수는 최대 $N! \leq 120$ 가지이다. 따라서 각각의 경우에 대해 부분문제 2의 알고리즘을 사용하여 모든 학생이 정확히 하나의 간식을 가져가는지 판정하면 된다.

부분문제 4

$\{1, 2, \dots, N\}$ 의 모든 부분집합 S 에 대해, $dp[S]$ 를 다음과 같이 정의한다.

- 만약 S 에 속한 학생들의 순서를 잘 정하여, 그 순서대로 방에 들여보냈을 때 S 에 속한 각 학생이 정확히 하나의 간식을 가져가도록 할 수 있다면 $dp[S] = 1$ 이다.
- 그렇지 않다면 $dp[S] = 0$ 이다.

모든 S 에 대하여, $dp[S] = 1$ 인 조건은 다음과 같다.

- S 가 비어 있거나,
- S 에서 i 를 제거한 집합 T 가 있어서 $dp[T] = 1$ 이고, i 번 학생이 좋아하는 간식 중 T 에 속한 학생들이 좋아하는 간식이 아닌 것은 정확히 1개이다.

따라서, S 의 모든 부분집합 T 에 대해 $dp[T]$ 를 구했다면 $dp[S]$ 는 $\mathcal{O}(N)$ 에 구할 수 있다.

비트필드를 이용한 다이내믹 프로그래밍을 사용하면 $\mathcal{O}(N2^N)$ 에 풀 수 있다.

부분문제 5

현재 일부 학생들이 이미 방에 들어가 정확히 하나의 간식을 가져갔고, 남아 있는 간식 중 정확히 하나만 좋아하는 학생 i 가 있다고 하자.

현재 상태에서 남은 학생들의 순서를 정하여 모든 학생들이 정확히 하나의 간식을 가져가도록 할 수 있다고 하자. 그 순서에서 i 보다 먼저 들어오는 학생은 i 가 가져갈 유일한 간식을 좋아할 수 없다. 따라서 i 를 그 학생들보다 앞으로 옮겨도 다른 학생들이 가져가는 간식에 영향을 주지 않는다. 그러므로, 만약 현재 상태에서 목표를 달성하는 것이 가능하다면 i 를 바로 들여보내도 답이 존재한다.

따라서 다음과 같은 방법으로 문제를 풀 수 있다.

- 현재 남은 간식 중 정확히 하나를 좋아하는 학생이 없다면 불가능하다고 판정한다.
- 그렇지 않은 경우, 정확히 하나의 간식을 좋아하는 학생을 찾고, 그 학생을 방에 들여보낸다. 만약 그러한 학생이 여러 명일 경우, 어떤 학생을 먼저 들여보내더라도 상관없다.
- 위 과정을 모든 학생이 간식을 가져갈 때까지 반복한다.

부분문제 5의 경우, $N \leq 300$ 이다. 다음과 같이 구현할 수 있다.

- 학생과 간식의 관계를 $N \times N$ 배열로 저장한다. i 번 학생이 j 번 간식을 좋아한다면 i 행 j 열에 1을 저장한다.
- 아직 방에 들어가지 않은 모든 학생의 행을 확인하여, 현재 남은 선호 간식이 정확히 하나인 학생을 찾는다. 그러한 행이 존재하지 않는다면 불가능하다고 판정한다.
- 찾은 행을 i 번 행이라 하고, i 번 학생이 좋아하는 간식 중 현재 유일하게 남아있는 것을 j 번 간식이라 하자. 배열에서 j 열을 모두 지운다.
- 위 과정을 모든 학생이 방에 들어갈 때까지 총 N 번 반복한다.

한 단계에 $\mathcal{O}(N^2)$ 번의 연산이 사용되고, 총 N 번 반복해야 하므로 전체 시간 복잡도는 $\mathcal{O}(N^3)$ 이다.

부분문제 6

부분문제 5의 알고리즘을 최적화할 수 있다. 각 학생 i 에 대해 i 번 학생이 좋아하는 간식 중 현재 방에 남은 것의 개수를 cnt_i 라 하고, 이 배열을 관리한다. 다음과 같이 해결할 수 있다.

- 방에 들어갈 학생을 구하는 것은 아직 방에 들어가지 않은 학생 중 $cnt_i = 1$ 인 학생 i 를 구하면 된다. 이때 i 번 학생이 방에 들어가면 정확히 하나의 간식 j 를 가져가게 된다.
- 이 과정에서 j 번 간식을 좋아하는 학생 k 에 대해, cnt_k 가 1 감소한다. j 번 간식을 좋아하는 모든 학생 k 를 찾아 cnt_k 를 갱신한다.

두 과정 모두 $\mathcal{O}(N)$ 에 처리할 수 있고, 총 N 번 반복하므로 전체 시간 복잡도는 $\mathcal{O}(N^2)$ 이다.

부분문제 7

부분문제 6의 알고리즘에서 추가적인 최적화를 할 수 있다.

- $cnt_i \leq 1$ 인 학생들을 큐(queue) 등의 자료구조를 사용해 저장한다. 어떤 학생이 간식을 가져가서 학생 i 의 cnt_i 가 1 감소했다 하자. 이때 $cnt_i \leq 1$ 인지 검사하고, 만약 그렇다면 i 를 큐에 추가하면 된다.
- j 번 간식을 좋아하는 학생들을 구하는 것은, 처음에 간식 j ($1 \leq j \leq N$) 각각에 대해 j 번 간식을 좋아하는 학생들의 배열을 `std::vector` 등의 동적 배열을 만들어 사용하면 된다. 각 간식은 최대 한 번 제거되므로, 각 배열도 최대 한 번 확인된다.

전체 시간복잡도는 $\mathcal{O}(N + \sum C_i)$ 이다.