

JavaScript

다이나믹 프로그래밍

다이나믹 프로그래밍 문제 풀이

다이나믹 프로그래밍 이해하기 | 코딩 테스트에서 자주 등장하는 다이나믹 프로그래밍 이해하기

강사 나동빈

JavaScript

다이나믹 프로그래밍

다이나믹 프로그래밍 문제 풀이

JavaScript 동적 계획법 **혼자 힘으로 풀어보기** 다이나믹 프로그래밍

JavaScript
동적 계획법
다이나믹
프로그래밍

문제 제목: 파도반 수열

문제 난이도: ★★☆☆☆

문제 유형: 다이나믹 프로그래밍

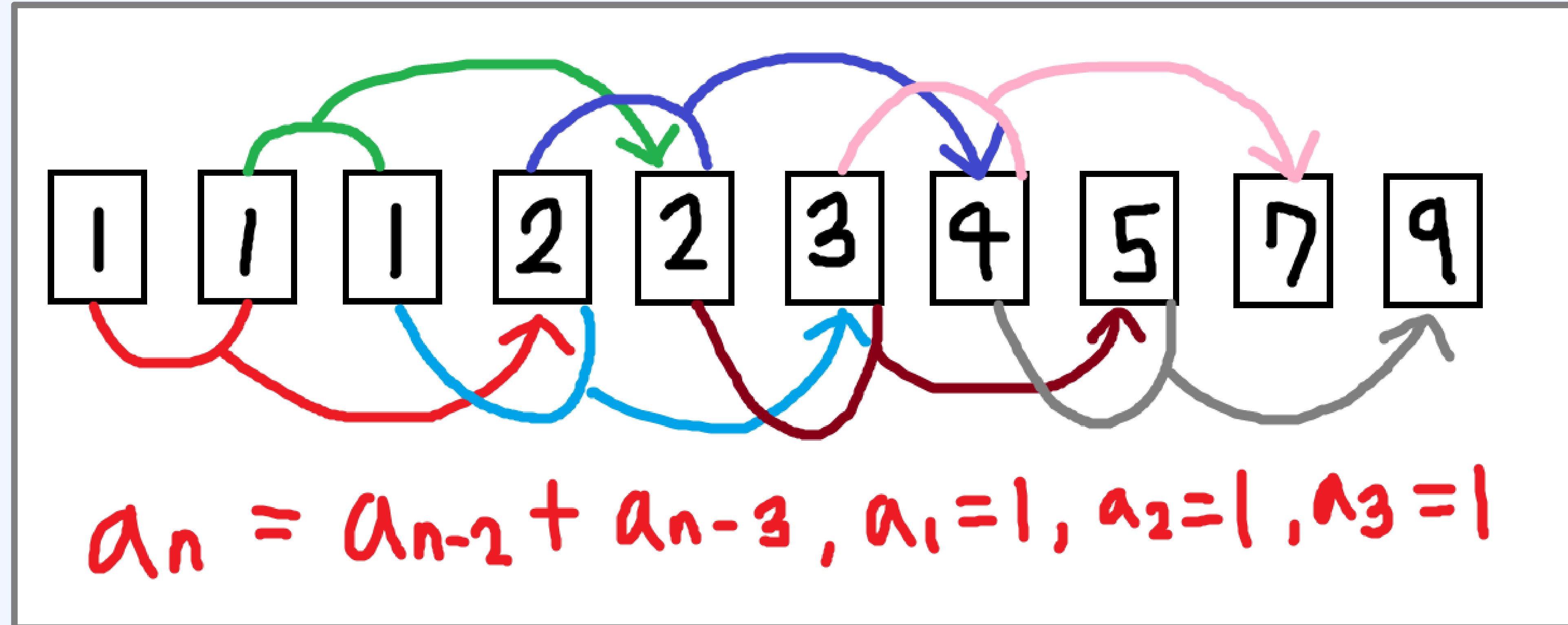
추천 풀이 시간: 30분

JavaScript 동적 계획법 문제 해결 아이디어

다이나믹 프로그래밍

JavaScript
동적 계획법
다이나믹
프로그래밍

- 점화식을 도출하기 위해 몇 가지 예시를 고려해 볼 수 있다.



JavaScript 동적 계획법

정답 코드 예시

다이나믹 프로그래밍

JavaScript
동적 계획법
다이나믹
프로그래밍

```
// readline 모듈보다는 fs를 이용해 파일 전체를 읽기
let file = require('fs').readFileSync('/dev/stdin');
let input = file.toString().split('\n');

// 앞서 계산된 결과를 저장하기 위한 DP 테이블 초기화
d = new Array(101).fill(0);
d[1] = 1;
d[2] = 1;
d[3] = 1;

// 다이나믹 프로그래밍(Dynamic Programming) 진행 (보텀업)
for (let i = 4; i <= 100; i++) {
    d[i] = d[i - 2] + d[i - 3];
}

let testCases = Number(input[0]); // 테스트 케이스의 개수
for (let t = 1; t <= testCases; t++) {
    let n = Number(input[t]); // N 입력받기
    console.log(d[n]);
}
```

JavaScript 동적 계획법 다이나믹 프로그래밍

혼자 힘으로 풀어보기

JavaScript
동적 계획법
다이나믹
프로그래밍

문제 제목: 정수 삼각형

문제 난이도: ★★☆☆☆

문제 유형: 다이나믹 프로그래밍

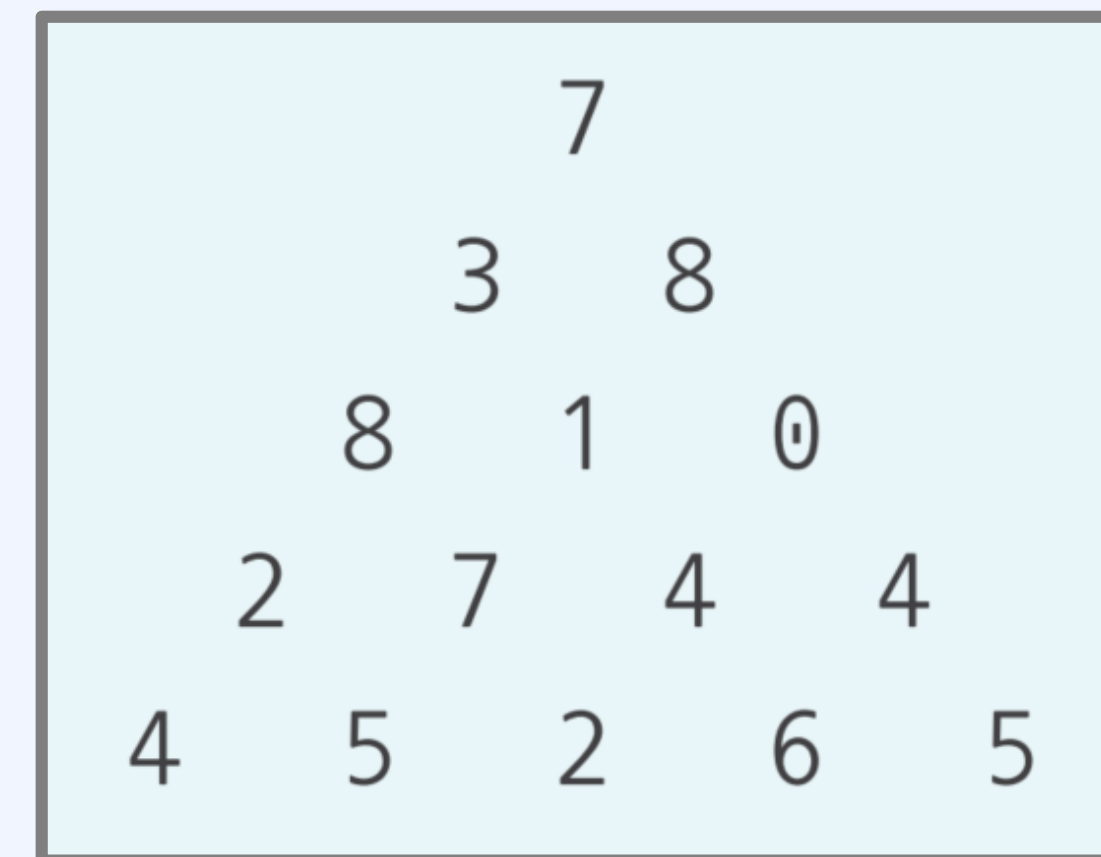
추천 풀이 시간: 30분

JavaScript 동적 계획법 문제 해결 아이디어

다이나믹 프로그래밍

JavaScript
동적 계획법
다이나믹
프로그래밍

- 아래 그림은 크기가 5인 정수 삼각형의 한 모습이다.
- 맨 위층부터 시작해서 아래에 있는 수 중 하나를 선택하여 아래층으로 내려올 때, 이제까지 선택된 수의 합이 최대가 되는 경로를 구하는 프로그램을 작성해야 한다.
- 아래층에 있는 수는 현재 층에서 선택된 수의 대각선 왼쪽 또는 대각선 오른쪽에 있는 것 중에서만 선택할 수 있다.
- 삼각형의 크기는 1 이상 500 이하이다.
- 삼각형을 이루고 있는 각 수는 모두 정수이다.
- 각 정수의 값의 범위는 0 이상 9999 이하이다.



JavaScript 동적 계획법 문제 해결 아이디어

다이나믹 프로그래밍

JavaScript
동적 계획법
다이나믹
프로그래밍

- 특정한 위치에 도달하기 위해서는 다음의 2가지 위치에서만 내려올 수 있다.
- ① 왼쪽 위 혹은 ② 바로 위

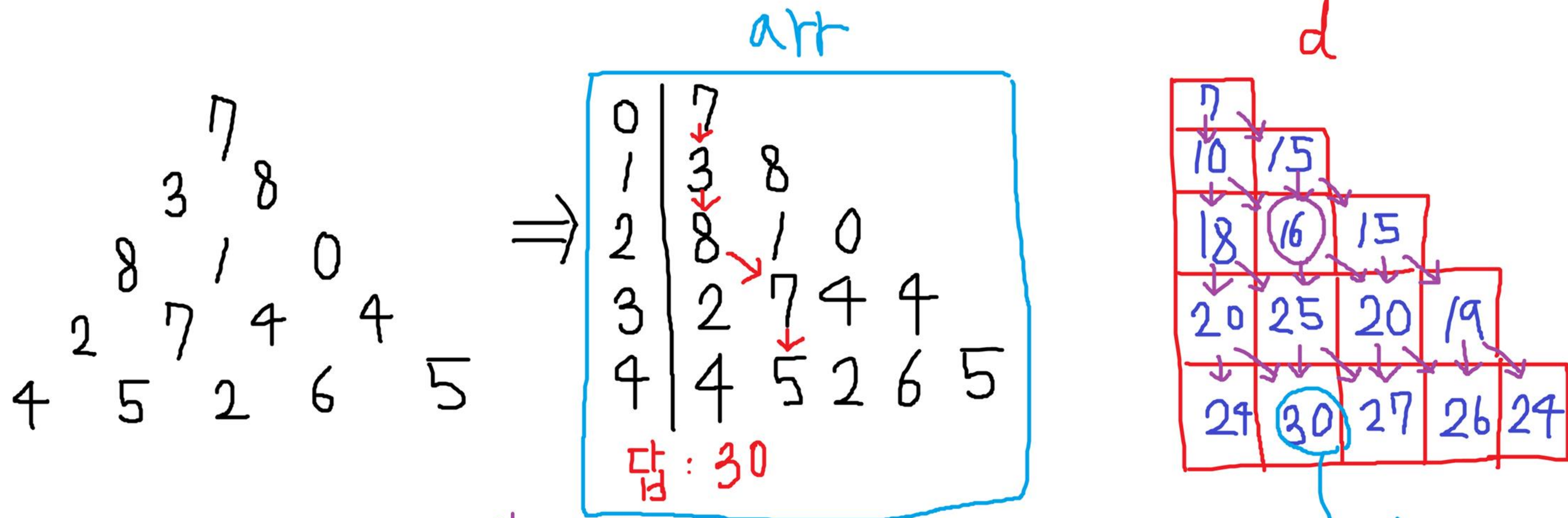
$$dp[i][j] = arr[i][j] + \max(dp[i-1][j-1], dp[i-1][j])$$

- 단, DP 테이블에 접근해야 할 때마다 리스트의 범위를 벗어나지 않는지 체크할 필요가 있다.
- 편의상 하나의 배열만 사용해 DP 테이블에 초기 데이터를 담아서 점화식에 따라 갱신할 수 있다.

JavaScript 동적 계획법 문제 해결 아이디어

다이나믹 프로그래밍

JavaScript
동적 계획법
다이나믹
프로그래밍



$$d_{ij} = \max(\overset{\text{왼쪽위}}{d_{i-1,j-1}}, \overset{\text{위}}{d_{i-1,j}}) + \text{arr}_{ij} \quad (d_{ij} \text{ 는 해당 위치까지 내려왔을 때 최댓값})$$

JavaScript 동적 계획법

정답 코드 예시

다이나믹 프로그래밍

JavaScript
동적 계획법
다이나믹
프로그래밍

```
let file = require('fs').readFileSync('/dev/stdin');
let input = file.toString().split('\n');

let n = Number(input[0]);
let dp = []; // 다이나믹 프로그래밍을 위한 DP 테이블 초기화

for (let i = 1; i <= n; i++) {
  let data = input[i].split(' ').map(Number);
  dp.push(data);
}

// 다이나믹 프로그래밍으로 2번째 줄부터 내려가면서 확인
for (let i = 1; i < n; i++) {
  for (let j = 0; j <= i; j++) {
    // 왼쪽 위에서 내려오는 경우
    let upLeft = 0;
    if (j !== 0) upLeft = dp[i - 1][j - 1];
    // 바로 위에서 내려오는 경우
    let up = 0;
    if (j !== i) up = dp[i - 1][j];
    // 최대 합을 저장
    dp[i][j] = dp[i][j] + Math.max(upLeft, up);
  }
}

console.log(Math.max(... dp[n - 1]));
```