Dynamic Programming II

이종서(leejseo)

Dynamic Programming II

- DP로 최단 경로를 구하는 방법
- LIS를 O(N log N) 시간에 구하는 방법
- DP를 이용한 경우의 수 세기

Floyd-Warshall Algorithm

- All Pair Shortest Path Problem
  - N개의 정점으로 구성된 가중치 있는 방향 그래프가 있다. 임의의 두 정점 사이의 최단 경로를 모두 구하여라.

Floyd-Warshall Algorithm

- 부분 문제 정의
  - D(k, i, j) = 1~k번 정점만 “중간 정점”으로 경유하면서 i와 j를 경유하는 최단 경로의 길이
  - D(0, i, j) = cost(i, j) if i->j 간선 존재

Floyd-Warshall Algorithm

- 부분 문제 정의
  - D(k, i, j) = 1~k번 정점만 “중간 정점”으로 경유하면서 i와 j를 경유하는 최단 경로의 길이
  - D(0, i, j) = cost(i, j) if i->j 간선 존재

- 점화식
  - D(k, i, j) = min(D(k-1, i, j), D(k-1, i, k) + D(k-1, k, j))

Floyd-Warshall Algorithm

- 연습문제: 공간 복잡도를 O(N^2)으로 하면서 구현하는 방법을 생각해보자.

LIS in O(N log N)

- D[i]: 마지막 항을 i로 하는 가장 긴 증가하는 부분 수열의 길이

LIS in O(N log N)

- D[i]: 마지막 항을 i로 하는 가장 긴 증가하는 부분 수열의 길이
- X[k]: D[i] = k인 A_i 들 중 가장 작은 수

LIS in O(N log N)

- D[i]: 마지막 항을 i로 하는 가장 긴 증가하는 부분 수열의 길이
- X[k]: D[i] = k인 A_i 들 중 가장 작은 수

- 그러면 배열을 어떻게 업데이트 해줄까?

LIS in O(N log N)

- D[i]: 마지막 항을 i로 하는 가장 긴 증가하는 부분 수열의 길이
- X[k]: D[i] = k인 A_i 들 중 가장 작은 수

- 그러면 배열을 어떻게 업데이트 해줄까?

- 관찰:
  - A_i의 직전 항을 생각해보자. A_i의 직전 항은 A_i보다 작다.
  - X 배열은 단조증가(i < j이면 X[i] <= X[j])

LIS in O(N log N)

- D[i]: 마지막 항을 i로 하는 가장 긴 증가하는 부분 수열의 길이
- X[k]: D[i] = k인 A_i 들 중 가장 작은 수

- 그러면 배열을 어떻게 업데이트 해줄까?

- D[i] := (X[k] < A_i인 최대의 k) + 1 = (X[k] >= A_i인 최소의 k)
- X[D[i]] := A_i

경우의 수 세기 – 백준 1328 고층 빌딩

- D(N, L, R) := N개의 건물이 있을 때 왼쪽에서 L개, 오른쪽에서 R개가 보이는 경우의 수

- D(N, L, R) = D(N-1, L-1, R) + D(N-1, L, R-1) + (N-2) * D(N-1, L, R)

경우의 수 세기 – 백준 1328 고층 빌딩

- D(N, L, R) := N개의 건물이 있을 때 왼쪽에서 L개, 오른쪽에서 R개가 보이는 경우의 수

경우의 수 세기 – 백준 1328 고층 빌딩

- D(N, L, R) := N개의 건물이 있을 때 왼쪽에서 L개, 오른쪽에서 R개가 보이는 경우의 수

- D(N, L, R) = D(N-1, L-1, R) + D(N-1, L, R-1) + (N-2) * D(N-1, L, R)