ZOAC 2024 풀이

Official Solutions

by

한양대학교 ERICA 알고리즘 학회 0&1

2024년 11월 9일

문제		의도한 난이도	출제자
A	당구 좀 치자 제발	**Easy**	김동우gi1dong2
B	정민이의 수열 제조법	**Easy**	김동우gi1dong2
C	동까뚱뽑 게임	**Medium**	김동우gi1dong2
D	랜덤 넘버 추측하기	**Medium**	오익준putdata
E	파괴왕 뚱뽑	**Medium**	김동우gi1dong2
F	골드바흐흑흙의 추측	**Hard**	권혁준khj20006
G	ZOAC 7	**Medium**	권혁준khj20006
H	GLCCDM	**Easy**	권혁준khj20006
I	익웜 바이러스	**Challenging**	오익준putdata

A. 당구 좀 치자 제발

`implementation`
출제진 의도 – **Easy**

- 제출 29번, 정답 12팀 (정답률 41.38%)
- 처음 푼 팀: **segfault**, 2분
- 출제자: 김동우`gildong2`

A. 당구 좀 치자 제발



- 동우의 현재 분노를 x라고 합시다. x의 초기값은 0입니다.
- 그 날의 날씨를 확인하여 비가 온다면 x의 값을 1만큼 증가시키고, 오지 않는다면 1만큼 감소시킵니다. 이후 변경된 x값을 결과값에 더합니다.
- x의 값을 변경하고 결과값에 더하는 과정을 1일 차부터 N일 차까지 진행한 후, 최종 결과값을 출력하면 됩니다.
- 총 시간복잡도 $\mathcal{O}(N)$에 해결할 수 있습니다.

B. 정민이의 수열 제조법

`sieve_of_eratosthenes`, `prefix_sum`
출제진 의도 – **Easy**

- ✓ 제출 23번, 정답 3팀 (정답률 13.04%)
- ✓ 처음 푼 팀: **segfault**, 9분
- ✓ 출제자: 김동우`gi1dong2`

B. 정민이의 수열 제조법



- 1 부터 N 까지의 정수 중에서 지문에 주어진 작업으로 만들 수 없는 수는 1 과 소수들입니다.
- 에라토스테네스의 체를 사용하여 2 부터 N 까지의 소수를 확인한 후, 1 과 소수의 개수를 담은 누적합 배열을 만듭니다.
- 만들어진 누적합 배열을 이용하여 익준이의 질문에 대한 답을 출력하면 됩니다.
- 총 시간복잡도 $\mathcal{O}\left(N \log \log N + M\right)$ 에 해결할 수 있습니다.

C. 동까뚱뽑 게임

`dfs`

출제진 의도 – **Medium**

- ✓ 제출 6번, 정답 3팀 (정답률 50.00%)
- ✓ 처음 푼 팀: **Proof by AC**, 38분
- ✓ 출제자: 김동우`gi1dong2`

C. 동까뚱빱 게임



- 동우와 혁준이가 각각 이기는 경우에 대해 생각해야 합니다.
- 동우가 이기기 위해서는 혁준이보다 점수가 높아야 합니다.
- 턴을 번갈아 가며 진행하기 때문에 혁준이가 이기는 경우는 동점일 경우뿐입니다.
- 이러한 경우를 바탕으로 승리 조건을 설정해야 합니다.

C. 동까뚱빱 게임



- 만약 i번 정점에서 게임을 시작할 때 혁준이가 이긴다면, i번 정점의 부모 정점에서 시작하면 동우가 이깁니다.
- 말단 정점에서 시작할 경우 이동할 수 있는 정점이 없어 동점으로 게임이 끝나므로 혁준이가 이깁니다.
- 따라서, 말단 정점부터 시작하여 현재 정점이 혁준이가 이기는 정점이면, 부모 정점은 동우가 이기는 정점으로 설정할 수 있습니다. 이것은 깊이 우선 탐색으로 해결할 수 있습니다.
- 총 시간복잡도 $\mathcal{O}(N)$에 해결할 수 있습니다.

D. 랜덤 넘버 추측하기

`segment_tree`
출제진 의도 – **Medium**

- ✓ 제출 18번, 정답 2팀 (정답률 11.11%)
- ✓ 처음 푼 팀: **segfault**, 18분
- ✓ 출제자: 오익준`putdata`

D. 랜덤 넘버 추측하기

- 회원의 번호가 a 이고 가중치가 p_a 이면, 리스트 W에서 이 회원의 응모권 순서는 $\sum_{i=1}^{a-1} p_i + 1$ 부터 $\sum_{i=1}^{a} p_i$ 에 해당합니다.
- 각 회원들의 응모권 순서를 빠르게 얻기 위해 누적합을 생각할 수 있습니다. 그런데 리스트 W는 당첨자를 뽑을 때마다 응모권을 제거하기 때문에 상태가 변화하므로 누적합 배열 또한 달라집니다.

D. 랜덤 넘버 추측하기

- 응모권을 제거한다는 것은 당첨된 회원의 가중치로 0으로 바꾼다는 것과 같습니다.
- 따라서 a번 회원이 뽑히게 될 시점의 응모권의 순서는 $\sum_{i=1}^{a-1} \begin{cases} 0 & p_i = 0 \\ p_i & p_i \neq 0 \end{cases} + 1$ 부터 $\sum_{i=1}^{a} \begin{cases} 0 & p_i = 0 \\ p_i & p_i \neq 0 \end{cases}$ 까지입니다.
- 값 업데이트와 구간 합을 빠르게 수행하기 위해 세그먼트 트리 혹은 펜윅 트리 자료 구조를 생각할 수 있습니다.
- 해당 자료 구조를 사용하면 총 시간 복잡도 $\mathcal{O}(N \log N)$ 만에 해결할 수 있습니다.

E. 파괴왕 뚱빱

`dijkstra`
출제진 의도 – **Medium**

- ✓ 제출 2번, 정답 2팀 (정답률 100.00%)
- ✓ 처음 푼 팀: **Proof by AC**, 60분
- ✓ 출제자: 김동우`gi1dong2`

E. 파괴왕 뚱뽑



- 어떤 칸에 도착하기 위해 소모된 힘이 주어진 힘보다 크다면, 그 칸에 도착할 수 없음을 의미합니다.
- 따라서 상하좌우로 이동할 때 힘이 부족하여 이동하지 못하는 경우는 고려하지 않고, 각 칸에 도착하기 위한 최소한의 힘을 $(1, 1)$에서 다익스트라를 시작하여 구하면 됩니다.
- 이제 다익스트라가 어떻게 진행되는지 살펴보겠습니다.

E. 파괴왕 뚱뽑



- 순간이동 혹은 상하좌우 이동을 통해 각 칸에 도착했을 때 소모된 힘을 저장하기 위해 $dist$ 배열을 3 차원으로 설정합니다.
- (x, y) 칸에 있는 기둥의 정보를 $board[x][y]$, (x, y) 칸을 tel 만큼의 순간이동을 통해 도착했을 때 들었던 힘을 $dist[tel][x][y]$ 라고 합시다.
- 현재 칸이 (x, y) 이고 다음 이동할 칸이 (nx, ny) 일 때, 이 칸의 $dist$ 값은 다음과 같이 설정됩니다.

  상하좌우 이동 $\rightarrow dist[tel][nx][ny] = \min(dist[tel][nx][ny], dist[tel][x][y] + board[nx][ny])$

  순간이동 $\rightarrow dist[tel + 1][nx][ny] = \min(dist[tel + 1][nx][ny], dist[tel][x][y])$

- 순간이동이 최대 T 번 가능하다는 것을 주의하며 다익스트라를 진행하면 됩니다.

E. 파괴왕 뚱뽑



- (x, y)를 도착하기 위해 필요한 최소 힘은 $dist[t][x][y](0 \leq t \leq T)$의 최솟값 입니다.
- 이를 바탕으로 각 쿼리에서 주어진 힘이 해당 칸에 도착하는 데 필요한 최소 힘 이상인지 확인하면 됩니다.
- 총 시간복잡도 $\mathcal{O}(NMT \log NMT + Q)$에 해결할 수 있습니다.

F. 골드바흐흑흙의 추측

`number_theory`, `knapsack`, `meet_in_the_middle`
출제진 의도 – **Hard**

- ✓ 제출 3번, 정답 1팀 (정답률 33.33%)
- ✓ 처음 푼 팀: **Proof by AC**, 80분
- ✓ 출제자: 권혁준 `khj20006`



- 구간 $[A, B]$ 내에 존재하는 소수의 개수를 C, 합을 S 라고 정의하겠습니다.
- 만약 $C \leq 40$ 이라면, 중간에서 만나기를 적용하여 $\mathcal{O}\left((C/2) \times 2^{(C/2)}\right)$ 에 해결할 수 있습니다.
- $C > 40$ 일 때는 어떻게 해결할 수 있을까요?

- ✓ 구간의 길이가 300인 모든 가능한 (A, B) 쌍 중, $C > 40$일 때 $C \times S$의 최댓값은 2×10^7을 넘지 않습니다.
- ✓ 따라서, 배낭 문제 해결법을 적용하여 $\mathcal{O}(C \times S)$에 해결할 수 있습니다!
- ✓ 위의 $C \times S$ 상한에 대한 증명은 로컬 환경에서 에라토스테네스의 체와 슬라이딩 윈도우를 적용하여 쉽게 확인해볼 수 있습니다.



G. ZOAC 7

dp

출제진 의도 – **Medium**

- 제출 1번, 정답 1팀 (정답률 100.00%)
- 처음 푼 팀: **Proof by AC**, 90분
- 출제자: 권혁준[khj20006]



- ✓ 격자 상의 i 행 j 열을 앞으로는 (i, j) 와 같이 표기하겠습니다.
- ✓ (i, j) 에 적힌 문자를 A_{ij} 라 하고, 문자 k 를 수집하는 경로를 생각해 봅시다. $(k \in \{\mathsf{Z}, \mathsf{O}, \mathsf{A}, \mathsf{C}\})$
- ✓ 순간이동을 고려하지 않는다면, 다음과 같은 dp 식을 정의하여 문제를 해결할 수 있습니다.

$dp[i][j] =$ $(1, 1)$에서 출발하여 (i, j)까지 이동했을 때, 문자 k의 최대 수집 개수

$$= \max(dp[i-1][j], dp[i][j-1]) + \begin{cases} 1 & (A_{ij} = k) \\ 0 & (A_{ij} \neq k) \end{cases}$$

- ✓ 답은 $dp[N][M]$ 이 됩니다.



- ✓ 순간이동을 고려하는 경우는 어떻게 해결할까요?
- ✓ (i, j) 에서 순간이동을 할 때 최대 수집 경로는 반드시
  $(i+1, 1) \to (N, M)$ 혹은 $(1, j+1) \to (N, M)$ 경로에 포함될 수밖에 없습니다.
- ✓ 따라서 항상 $(i+1, 1)$ 혹은 $(1, j+1)$ 로 순간이동하는 것이 최선의 선택입니다.



- ✓ 새로운 dp 식을 정의합시다.

$dp2[i][j] =$ (i,j)에서 출발하여 (N,M)까지 이동했을 때, 문자 k의 최대 수집 개수

$$= \max(dp2[i+1][j], dp2[i][j+1]) + \begin{cases} 1 & (A_{ij} = k) \\ 0 & (A_{ij} \neq k) \end{cases}$$

- ✓ (i,j)에서 순간이동하는 경우, 최대 수집 개수는 $dp[i][j] + \max(dp2[i+1][1], dp2[1][j+1])$ 이 됩니다.
- ✓ 모든 가능한 (i,j) 쌍에 대한 최대 수집 개수 중 최댓값을 구해주면 됩니다.
- ✓ $dp, dp2$를 계산하는 데 $\mathcal{O}(NM)$, 최댓값을 확인하는 데 $\mathcal{O}(NM)$ 이므로 총 시간복잡도 $\mathcal{O}(NM)$으로 해결할 수 있습니다.

H. GLCCDM

`math, bruteforcing`
출제진 의도 – **Easy**

- 제출 4번, 정답 2팀 (정답률 50.00%)
- 처음 푼 팀: **segfault**, 35분
- 출제자: 권혁준[khj20006]

H. GLCCDM



- 먼저, $d_1 < \cdots < d_K$ 를 만족하도록 출력한다고 가정합시다.
- 출력해야 할 수들이 어떤 성질을 지녀야 하는지 관찰해 봅시다.
  - 모든 수는 A 의 배수여야 합니다.
  - 모든 수는 B 의 약수여야 합니다.
- $d_1 = A, d_K = B$ 로 고정해두고, 위의 성질을 모두 만족하도록 구간 (A, B) 내에서 서로 다른 정수 $K-2$ 개를 구할 수 있다면 수열은 항상 존재합니다.
- 어떤 수 x 를 A 로 나눈 나머지가 0 이고, B 를 x 로 나눈 나머지가 0 이면 위의 성질을 만족하고, 시간복잡도는 수 하나당 $\mathcal{O}(1)$ 입니다.
- 따라서 총 시간복잡도 $\mathcal{O}(B)$ 에 해결할 수 있습니다.

- ✓ 만약 $B \leq 10^{18}$ 이라면 이 문제를 풀 수 있을까요?

I. 익웜 바이러스

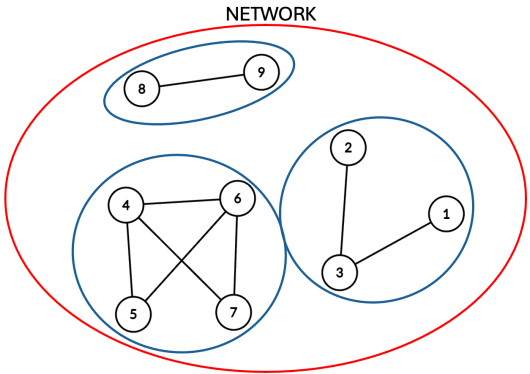
`minimum_spanning_tree`, `union_find`, `greedy`
출제진 의도 – **Challenging**

- ✓ 제출 0번, 정답 0팀 (정답률 -.-%)
- ✓ 처음 푼 팀: -, -분
- ✓ 출제자: 오익준`putdata`



- 먼저 익준(가상 노드)이가 PC를 직접 해킹하여 바이러스를 주입하는 행위의 가상 간선이 있다고 생각해봅시다.
- 이러한 가상 간선과 네트워크의 간선이 포함된 그래프에 대해 MST(최소 스패닝 트리)를 구하면 최적의 총 전송 시간을 구할 수 있습니다.
- 이 때, 사용된 가상 간선이 K 보다 작거나 같으면 그 값이 답이 됩니다.
- 이제 K 보다 클 경우에 대해 생각해봅시다.

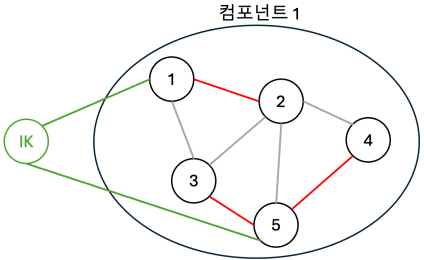
- PC들이 서로 이어져 있지 않은 경우가 있기 때문에, 그림과 같이 네트워크 안에 여러 컴포넌트가 존재할 수 있습니다.



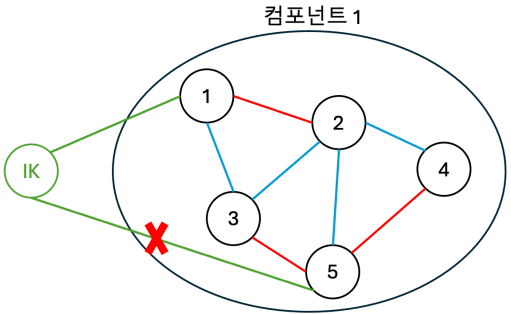
I. 익웜 바이러스



- 네트워크에 바이러스가 퍼질 수 없는 경우는 컴포넌트의 개수가 K 보다 클 때 뿐입니다. 즉, 모든 컴포넌트가 가상 노드와 적어도 하나의 가상 간선을 통해 연결되어 있어야 합니다.
- 한 컴포넌트가 가상 노드와 여러 개의 가상 간선을 통해 연결 되어 있다면, 컴포넌트 안에서 연결된 가상 간선의 개수 만큼의 서브 컴포넌트가 생기게 됩니다.
- 예를 들어, 아래의 그림과 같이 MST가 완료된 시점의 한 컴포넌트에 연결된 가상 간선의 개수가 2개라고 해봅시다.



- ✓ 이 컴포넌트의 가상 간선 개수를 하나 줄일 경우 모든 PC가 해킹 되어야 하기 때문에, 가상 노드와의 연결이 끊어진 서브 컴포넌트와 연결해 줄 간선을 하나 찾아 새롭게 연결해 주어야 합니다.





- 이러한 특성을 토대로, 현재 스패닝 트리에 대해서 ’연결된 가상 간선의 가중치 - 서브 컴포넌트와 연결될 네트워크 간선의 가중치’가 최소가 되는 간선 하나를 찾아 가상 간선을 하나 끊고 네트워크 간선을 하나 연결한 새로운 스패닝 트리를 만듭니다.
- 이를 연결된 가상 간선의 개수가 K 가 될 때까지 반복합니다.
- 이런 과정을 잘 구현해주면 됩니다.
- * 참고 - 처음에 MST를 구축하지 않고 모두 가상 간선에 연결한 후 시작해도 무방합니다.