SciOI 2022 Editorial

Official Solutions

by

SciCom 2022

2022년 12월 30일

문제		의도한 난이도	출제자
A	카드 뽑기	**Easy**	qwerasdfzxcl
B	최대 점수	**Easy**	qwerasdfzxcl
C	점수 내기	**Medium**	math_rabbit_1028
D	직육면체	**Medium**	andyandy
E	앤디 공격하기	**Medium**	math_rabbit_1028
F	공격 릴레이	**Hard**	math_rabbit_1028
G	지뢰 피하기	**Hard**	math_rabbit_1028
H	영어 시간	**Hard**	79brue
I	생물 연구	**Hard**	qwerasdfzxcl
J	비밀 기지	**Challenging**	79brue

목차

A. 카드 뽑기 4
B. 최대 점수 7
C. 점수 내기 9
D. 직육면체 11
E. 앤디 공격하기 14
F. 공격 릴레이 17
G. 지뢰 피하기 20
H. 영어 시간 23
I. 생물 연구 34
J. 비밀 기지 41



A. 카드 뽑기

`math`
출제진 의도 – **Easy**

- 제출 371번, 정답 155명 (정답률 44.474%)
- 처음 푼 사람: **yup0927**, 1분
- 출제자: `qwerasdfzxcl`

- 가능한 게임 결과는 $2^N - 1$ 가지 입니다.
- 각 카드를 선택할 확률과 선택하지 않을 확률은 같으므로 각 결과가 나올 확률은 $\frac{1}{2^N - 1}$ 로 동일합니다.
- 따라서, 구하고자 하는 값은 뽑은 카드에 적힌 정수가 모두 다른 경우의 수와 동일합니다.

A. 카드 뽑기

- cnt_i 를 i 가 적힌 카드의 개수라고 정의합시다.
- i 가 적힌 카드는 1개 이하로 뽑아야 하므로, $cnt_i + 1$ 가지의 경우의 수가 나옵니다.
- 공집합은 허용되지 않으므로 $(2^N - 1)p = \prod_{i=1}^{N} (cnt_i + 1) - 1$ 입니다.
- 시간복잡도는 $O(N)$ 입니다.



B. 최대 점수

`greedy`, `two pointer`
출제진 의도 – **Easy**

- 제출 508번, 정답 61명 (정답률 12.205%)
- 처음 푼 사람: **koosaga**, 11분
- 출제자: `qwerasdfzxcl`

B. 최대 점수

- 리프가 현재까지 방문한 방의 번호를 $l, l+1, \ldots, r$ 이라고 합시다.
- 일반성을 잃지 않고, l 번 방의 왼쪽으로 이동한다고 합시다.
- 리프가 이동을 시작한 시점보다 점수가 높아질 때까지 이동을 해야 이득입니다.
- 만약 이러한 지점에 도달하기 전에 게임오버가 일어난다면, 현재 상태에서 왼쪽으로 이동할 수 없습니다.
- 따라서, l 의 왼쪽과 r 의 오른쪽에 대해 점수가 현재보다 처음으로 높아지는 지점을 각각 구한 후, 이동할 수 있는 방향으로 이동하는 것을 반복하면 됩니다.
- 투 포인터를 이용하여 구현하면 $O(N)$ 에 문제를 해결할 수 있습니다.



C. 점수 내기

dp

출제진 의도 – **Medium**

- 제출 125번, 정답 23명 (정답률 19.200%)
- 처음 푼 사람: **jk410**, 26분
- 출제자: `math_rabbit_1028`

C. 점수 내기



- $DP[i]$ 를 점수가 i 인 사람이 내는 금액이라고 합시다.
- 점수가 i 인 사람은 점수가 $i+1$ 점 이상인 사람을 보고 돈을 지불해야 합니다.
- 점수가 i 점인 사람이 점수가 $i+x+1$ 점 이상인 사람에게 지불해야 하는 돈은 $DP[i+x]+y \times$ (점수가 $i+x+1$ 점 이상인 사람 수) 와 같습니다.
- 점수가 $i+1$ 점 이상 $i+x$ 점 이하인 사람에게 지불해야 하는 돈은 누적 합을 이용해 구할 수 있습니다.
- 시간 복잡도 $O(N+\max(X_i))$



D. 직육면체

`math, ad_hoc`
출제진 의도 – **Medium**

- 제출 185번, 정답 126명 (정답률 68.649%)
- 처음 푼 사람: **wsb1017**, 4분
- 출제자: andyandy

Observation

- 직육면체의 부피인 $a \times b \times c$가 조각 부피, $1 \times p \times p = p^2$ 의 배수가 되어야 함을 알 수 있습니다.
- a, b, c 중 2개 이상이 p의 배수이면, 채우는 방법이 존재함도 관찰할 수 있습니다.
- 이제 이 경우밖에 없음을 증명합시다.

Claim

- a, b, c 중 p의 배수가 2개 이상일 때, 그리고 이 때만 채우는 방법이 존재한다.

Proof

- p의 배수가 2개 이상일 때 채울 수 있음은 자명합니다.
- 각 면은 $1 \times p$와 $p \times p$ 모양들로 채워져야 하므로, 모든 면의 넓이는 p의 배수입니다. 모서리 길이들 중 p의 배수가 하나만 있다면 나머지 두 모서리로 이루어진 직사각형은 채워질 수 없습니다. 따라서 완벽하게 채워질 수 없습니다.



E. 앤디 공격하기

knapsack
출제진 의도 – **Meidum**

- 제출 18번, 정답 7명 (정답률 38.889%)
- 처음 푼 사람: **angello10**, 61분
- 출제자: `math_rabbit_1028`

- 각 사람별로 몇 칸을 옮겨야 앤디를 볼 수 있는지를 먼저 구합니다.
- 이 값을 A_i 라고 합시다.
- 이미 볼 수 있다면 $A_i = 0$ 이고, 아니라면 방향과 위치에 따라 A_i 를 구해야 합니다.
- 구현의 편의성을 위해 방향을 고정하고 그 방향에 맞게 사람의 위치를 회전/대칭이동 시키는 것이 좋습니다.
- 또한, A_i 칸 옮겼을 때 앤디와의 최대 거리를 B_i 라고 합시다. 이 값도 구할 수 있습니다.

E. 앤디 공격하기



- 이제 i번 사람을 A_i만큼 옮겼을 때 B_i만큼의 공격량을 확보할 수 있습니다.
- 또한, 1만큼 추가적으로 옮길 때마다 1만큼의 추가 공격량을 확보할 수 있습니다.
- 이 상황은 배낭 문제와 거의 동치이고, 따라서 Knapsack Problem을 풀면 $O(Nk)$에 문제를 해결할 수 있습니다.
- 이때, 한 명도 앤디를 바라볼 수 없는 상황에서 추가 공격량만으로 총 공격량 k를 채우는 경우가 나오지 않도록 처리해야 합니다.



F. 공격 릴레이

`segment_tree`, `sweeping`
출제진 의도 – **Hard**

- 제출 0번, 정답 0명 (정답률 0.000%)
- 처음 푼 사람: **없음**
- 출제자: `math_rabbit_1028`



- 술래가 정해지면 그 다음 턴의 술래도 정해집니다.
- 따라서 술래와 그 다음 턴의 술래의 관계는 함수가 되고, 이를 Functional Graph로 해석할 수 있습니다.
- 결론적으로, 술래를 계속 나열하면 어느 순간 사이클에 들어가게 되며, 이때부터는 해당 사이클에 있는 술래들만 반복적으로 나타납니다.
- 여기까지 알아낸 상황에서, 다음의 두 가지만 계산하면 문제를 풀 수 있습니다.
  - 현재 술래가 i번일 때, 얻는 공격량은 얼마인가?
  - 현재 술래가 i번일 때, 다음 술래는 누구인가?

F. 공격 릴레이

- 결론부터 말하자면, 각각은 세그먼트 트리 스위핑으로 풀 수 있습니다.
- 각각의 경우 스위핑은 총 8번씩을 해야 하며, 술래의 위치를 기준으로 방향을 45도씩 나눠서 해야 합니다.
- 이때 45도 영역을 스위핑으로 처리하기 위해 좌표 변환을 해야 하는데, 일차변환을 통해 x 축은 그대로, $x = y$ 를 y 축으로 옮길 수 있게 됩니다.
- 공격량을 구할 때는 개수와 합을 관리하는 세그먼트 트리가 필요합니다.
- 또 공격량 처리가 끝난 뒤, 술래가 원점에 있을 때 $x = y$ 나 $x + y = 0$ 위의 다른 학생들이 여러 번 세어지지 않도록 조심해야 합니다.
- 술래를 구할 때는 최솟값을 관리하는 세그먼트 트리로 충분합니다.
- 이 모든 과정을 처리하면 $O(N \log N)$ 에 문제를 풀 수 있습니다.



G. 지뢰 피하기

`dijkstra`
출제진 의도 – **Hard**

- 제출 35번, 정답 11명 (정답률 31.429%)
- 처음 푼 사람: **eunlin**, 47분
- 출제자: `math_rabbit_1028`

- 각 선물을 얻고 나서 다음 선물을 얻기 전에 출구 칸을 무조건 거쳐야 한다고 생각해 봅시다.
- 이 경우에도 답은 변하지 않습니다. 어차피 해당 선물을 얻고 언젠가는 출구로 가야 하기 때문입니다.
- 따라서 각 선물별로, 이 선물이 최대 몇 번째 선물까지 될 수 있는지를 구하면 됩니다.
- i번 선물이 A_i 번째 선물까지 될 수 있다는 것은, 이 선물을 A_i 번째로 가져가면 그 뒤에 출구로 갈 수 있지만, A_{i+1} 번째 선물로 가져가면 출구로 갈 수 없다는 뜻입니다.

- A_i 는 출입구 칸에서 선물 i 가 있는 모든 경로에 대해, 지뢰에 쓰인 수의 최솟값의 최댓값 (지뢰가 없으면 ∞) 로 정의됩니다.
- 이 값은 다익스트라 알고리즘을 이용해 찾을 수 있습니다. (일반적인 형태는 아니고, 조금 변형해야 합니다.)
- A_i 를 모두 구하고 나면, A_i 가 작은 선물부터 그리디하게 뽑아 선물의 최대 개수를 알 수 있습니다.
- 시간 복잡도는 $O(NM \log NM)$ 입니다.



H. 영어 시간

DP

출제진 의도 – **Hard**

- 제출 11번, 정답 2명 (정답률 18.182%)
- 처음 푼 사람: **ainta**, 182분
- 출제자: `79brue`

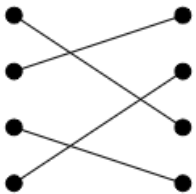


Definition

- 양 끝점의 높이 차이가 1 이하인 선분을 **중요한 간선**이라고 합시다.
- 그렇지 않은 간선을 **일반 간선**이라고 합시다.

Observation

- 중요한 간선 두 개가 교차할 경우, 두 간선에서 끝점을 하나씩 골라 x 좌표가 같고 y 좌표 차이가 1 이 되게 할 수 있습니다.
- 예를 들어, 아래 그림의 두 중요한 직선은 각각 $(2, 2)$와 $(2, 3)$을 끝점으로 가집니다.



Proof

- 귀류법을 이용합시다.
- 두 중요한 직선이 각각 $(1, a)$와 $(2, x)$, $(1, b)$와 $(2, y)$를 끝점으로 가집니다고 합시다.
- 일반성을 잃지 않고 $a < b$를 가정합시다. 이때 두 선분이 교차하므로 $y < x$입니다.
- 귀류 조건에 따라 $b - a > 1$이고 $x - y > 1$입니다.
- 따라서 $c \in (a, b)$이고 $z \in (y, x)$인 두 정수 c와 z가 존재합니다.
- $(1, c)$와 $(2, z)$가 어디로 연결되어야 하는지를 살펴봅시다.

H. 영어 시간

- $(1, c)$가 $(2, c')$과 연결되어 있다고 합시다.
- 마찬가지로 $(2, z)$가 $(1, z')$과 연결되어 있다고 합시다.
- $y < c' < x$이면 사이클이 생기므로 모순입니다.
- 마찬가지로 $a < z' < b$여도 사이클이 생기므로 모순입니다.
- 따라서 $c' < y$이거나 $c' > x$인데, 일반성을 잃지 않고 $c' < y$라고 합시다.
- 이때 $z' < a$이거나 $z' > b$인데, $z' < a$면 사이클이 생기므로 모순입니다.
- 따라서 $z' > b$여야 합니다.

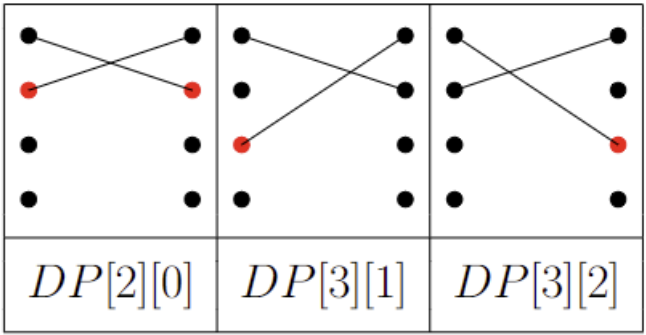
- 이 시점에서 (a, b) 내의 모든 c에 대해, $(1, c)$가 연결되는 점 $(2, c')$에 대해 $c' < y$여야 사이클이 생기지 않습니다.
- 마찬가지로 (y, x) 내의 모든 z에 대해, $(2, z)$가 연결되는 점 $(1, z')$에 대해 $z' > b$여야 사이클이 생기지 않습니다.
- 이때 사이 간선들 때문에 두 중요한 간선 위쪽의 점들과 아래쪽의 점들은 서로 연결될 수 없게 됩니다.
- 다시 말해, $a' < a$이고, $x' > x$인 적당한 a', x'에 대해, $(1, a')$와 $(2, x')$가 연결되면 사이클이 생기므로 서로 연결될 수 없습니다.
- 이는 반대 방향, 즉 $b' > b$, $y' < y$인 방향에 대해서도 성립합니다.

- 이 말은, $a' < a$인 a'에 대해 $(1, a')$와 $(2, a'')$가 연결된다면 $a'' < x$라는 뜻이 됩니다.
- 마찬가지로 $a' > a$인 a'에 대해 $(1, a')$와 $(2, a'')$가 연결된다면 $a'' > x$가 됩니다.
- 그런데, x좌표가 1인 점 중 y좌표가 a 이하이면서 아직 연결되지 않은 점의 개수는 $(a - 1)$개입니다.
- 또, x좌표가 2인 점 중 y좌표가 x 이하이면서 아직 연결되지 않은 점의 개수는 $(y - 1) - (b - a - 1) = a - b + y$개입니다.
- 이들이 서로 일대일 대응되려면 $a - 1 = a - b + y$, 즉 $b - y = 1$이어야 합니다.
- 이는 $(1, b)$와 $(2, y)$가 중요한 간선이라는 가정에 모순입니다.

Approach

- 지금까지 본 중요한 간선들의 끝점을 중심으로 한 DP 풀이를 만들어 봅시다.
- $DP[i][d]$의 정의는 아래와 같습니다.
  - $d = 0$인 경우: $(1, i)$와 $(2, i)$가 모두 간선의 끝점이며, y좌표가 $i + 1$ 이상인 점은 하나도 연결되지 않은 경우의 수
  - $d = 1$인 경우: $(1, i)$가 $(2, j)\ (j < i - 1)$와 연결되어 중요한 간선을 이루고 있으며, $(1, x)\ (1 \le x \le i - 2)$는 모두 연결되어 있고, $(2, y)\ (1 \le y < i - 1)$ 역시 모두 연결되어 있고, 명시되지 않은 나머지 점은 모두 연결되지 않은 상황의 경우의 수
  - $d = 2$인 경우: 위 경우의 좌우반전입니다.

- 각 경우를 그림으로 표현하면 다음과 같습니다.





- DP에서 가능한 전이는 대칭을 제외하고 다음과 같습니다. 상태 d를 기준으로 설명합니다.
- $0 \to 0$: 한 칸짜리 연결성분 하나를 만들거나, 두 칸짜리 연결성분 하나를 만듭니다.
- $0 \to 1$: $DP[i-1][0]$에서 시작한다면, $(2, i)$에서 $(1, i+2)$나 더 큰 y 좌표의 점과 연결해야 합니다. 이때 $(1, y)$와 연결이 가능할 조건은
  - $(2, i)$와 $(1, y)$의 연결이 가능하고
  - $i \leq j <\leq y-2$인 모든 정수 j에 대해 $(1, j)$과 $(2, j+1)$의 연결이 가능해야 합니다.
- $1 \to 0$: 간선 하나의 가능성 여부만 확인하면 됩니다.
- $1 \to 2$: $0 \to 1$과 비슷한 방법으로 하면 되지만, 디테일이 조금 다릅니다.
- $1 \to 1$: 불가능합니다.



- 쉬운 전처리를 통해 앞선 과정을 $O(N)$에 진행할 수 있습니다.
- 여기까지 설명했으면 구현이 어렵지 않겠지만, 몇 가지 힌트를 남깁니다.
- 두 정점이 연결 가능한지를 판별할 때에는 선분이 이미 존재하거나, 양쪽 점 모두 이어지지 않았는지를 확인합니다.
- $(1, i)$와 $(2, i+1)$이 연결되었는지를 저장하는 배열을 만들고, 이것의 누적합을 관리합니다.
- 세그먼트 트리를 활용해서 푸는 것도 가능하지만, 누적 합으로 모두 대체할 수 있습니다.



I. 생물 연구

`dp_tree`
출제진 의도 – **Hard**

- 제출 9번, 정답 5명 (정답률 55.556%)
- 처음 푼 사람: **arnold518**, 168분
- 출제자: `qwerasdfzxcl`

$O(N^2)$ 풀이

- $dp_{v,i}$를 점 v의 서브트리에 속한 점 중 v와 거리가 i인 점의 개수라고 정의합시다.
- 최소 공통 조상이 점 v이면서 v와의 거리가 d인 집합의 개수는 $\prod_{P_w=v}(dp_{w,d}+1)-\sum_{P_w=v}dp_{w,d}-1$입니다.
- 이를 구현하면 $O(N^2)$입니다.

$O(N \log N)$ 풀이

- h_v를 점 v와 v의 서브트리에 속한 점 사이의 거리의 최댓값으로 정의합시다.
- $P_w = v$인 w에 대해 순서쌍 (h_w, w)가 감소하도록 나열한 수열을 $\{adj(v)_i\}$라고 합시다.
- $O(N^2)$ 풀이에서 점 v에 대한 dp배열과 집합 개수를 계산하는데 걸리는 시간은 $O\left(\sum_{i=1}^{deg(v)} h_{adj(v)_i}\right)$ 입니다.

$O(N \log N)$ 풀이

- v 로부터 거리가 $h_{adj(v)_2}$ 초과인 점들은 추가적인 계산이 필요하지 않습니다.
- 이를 이용하면 점 v에 대해 dp배열과 집합 개수를 $O\left(\sum_{i=2}^{deg(v)} h_{adj(v)_i}\right)$ 에 계산할 수 있습니다.
- 전체 시간복잡도는 $O\left(\sum_{v=1}^{N} \sum_{i=2}^{deg(v)} h_{adj(v)_i}\right)$ 입니다.

$O(N \log N)$ 풀이

- 놀랍게도, $\sum_{v=1}^{N} \sum_{i=2}^{deg(v)} h_{adj(v)_i} = N - h_{adj(1)_1}$ 이 성립합니다. 이에 대한 증명은 후술합니다.
- 따라서, 전체 시간복잡도는 $O(N \log N)$ 입니다. 정렬에 드는 시간을 제외하면 $O(N)$ 입니다.
- 점 v에 대한 정보를 계산할 때, 시간복잡도가 $O\left(\sum_{i=1}^{deg(v)} h_{adj(v)_i}\right)$ 또는 $O\left(h_{adj(v)_2} deg(v)\right)$ 가 되지 않도록 유의해야 합니다.

시간복잡도 증명

- ✓ $w = adj(v)_1$ 를 만족하는 점 v, w 에 대해, 간선 $\{v, w\}$ 를 **특별한 간선**이라고 합시다.
- ✓ 특별한 간선들은 일직선 형태의 컴포넌트인 **체인**을 이룹니다.
- ✓ 각 체인은 어떤 점 x 에서 시작하여 리프까지 연결되어 있는 형태입니다.
- ✓ C_v 를 점 v 와 동일한 체인에 속한 점 중 h_i 가 최대인 점의 번호로 정의합시다. 즉, v 가 속한 체인의 꼭대기 점입니다.

시간복잡도 증명

- $f(x) = P_{C_x}$ 로 정의합시다.
- 1 이상 N 이하인 임의의 정수 v에 대해, $f(x) = v$를 x는 정확히 $\sum_{i=2}^{deg(v)} h_{adj(v)_i}$ 개입니다.
- $f(x) = -1$인 x는 $h_{adj(1)_1}$ 개입니다.
- 따라서, $\sum_{v=1}^{N} \sum_{i=2}^{deg(v)} h_{adj(v)_i} = N - h_{adj(1)_1}$ 입니다.



J. 비밀 기지

`segment_tree`, `euler_tour_trick`, `centroid_decomposition`
출제진 의도 – **Challenging**

- 제출 4번, 정답 1명 (정답률 25.000%)
- 처음 푼 사람: **koosaga**, 57분
- 출제자: `79brue`, `blackking26`

- dfs ordering, 즉 전위 순회 순서대로 정점을 배열합니다.
- 이때 각 정점에 있는 사람 수만큼 정점 번호를 (dfs ordering 순서대로) 적습니다.
- 모든 정점에 있는 인구의 합을 S라고 합시다.
- 위에서 숫자를 적을 때 $\lceil \frac{S}{2} \rceil$ 번째에 오는 것을 x라고 합시다.
- 이때 비밀 기지 최적의 위치 중 적어도 하나는 x 또는 x의 조상 정점이라는 것이 보장됩니다.
- 증명은 귀류법으로 간단한 케이스 처리를 통해 가능합니다.



- 이제 정점 x로부터 루트 노드까지 가는 경로에서 이분 탐색을 하면서 비밀 기지의 위치를 잡습니다.
- 비밀 기지의 위치는 그 경로 위에서 서브트리 내의 인구 수가 절반 이상이 되는 가장 깊이가 깊은 노드가 되어야 합니다.
- 이후 Centroid Decomposition을 통해 거리 합을 구하면 됩니다.

- ✓ 풀이에 적힌 모든 알고리즘을 이해하고 있다면 디테일은 어렵지 않습니다.
- ✓ 트리와 쿼리 17, 18을 풀고 오는 것이 도움이 될 수 있습니다.