

ZOAC 2022 해설

rasauq1122
mksmk606
rhdqor213
hyoseok

문제	예상난이도	출제자
A ZOAC 5	Easy	rasauq1122
B 전투의 신	Easy	mksmk606
C 황금 칵테일	Easy	rhdqor213
D 이 사람 왜 이렇게 1122를 좋아함?	Easy	rasauq1122
E 색종이와 공예	Medium	rasauq1122
F 용 조련사 룰루	Medium	rasauq1122
G map, filter	Hard	hyoseok
H 가장 작은 수	Hard	rasauq1122
I 인생은 B와 D 사이의 C다.	Challenging	rasauq1122
J 사기 주사위	Medium	mksmk606
K 순열 구하기	Medium	hyoseok
L 형광펜 강민우	Challenging	mksmk606
M 이게 게임이냐?	Hard	rhdqor213

도움주신 분들

외부 검수진
@eaststar
@tony9402
@amsminn
@hellogaon

후원



A. ZOAC 5

implementation

First Solve : 권혁준 (1 min)

출제자 : rasauq1122

예상 난이도 : **Easy**

A. ZOAC 5

- 성우는 어떤 알파벳도 두 번 이상 입력하지 않는다.
- 따라서 입력된 모든 알파벳은 주어진 문자열에서 정확히 N 개만 존재한다.
- 문자열에 존재하는 알파벳 하나의 개수를 세준 후, 그 값을 출력한다.

B. 전투의 신

math

First Solve : 권혁준 (9 min)

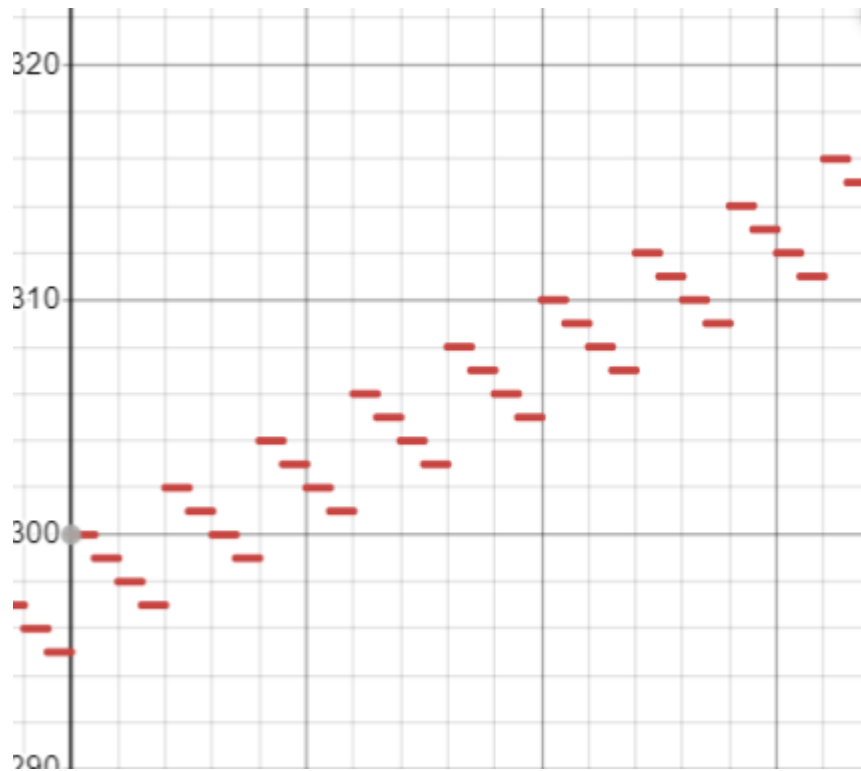
출제자 : mksmk606

예상 난이도 : **Easy**

B. 전투의 신

- 탱커를 x 명 고용한다고 가정하자.
- 남은 돈으로 최대한 딜러를 고용하는 것이 최선이므로 딜러를 $[(N - xP_A)/P_B]$ 명 고용한다.
- $xP_A \leq N$ 을 만족하는 모든 음이 아닌 정수 x 에 대하여 위의 과정을 반복하고 그 중 전투력의 합이 가장 큰 경우를 출력한다.

B. 전투의 신



- 탱커를 최대한 고용하고 남은 돈으로 딜러를 고용하거나 딜러를 최대한 고용하고 남은 돈으로 탱커를 고용하는 것 중 전투력의 합이 큰 경우를 구하는 방법은 최선이 아닐 수 있다.
- x 와 y 가 항상 정수이기 때문에 남는 돈이 발생할 수도 있기 때문이다.
- 따라서 x 에 대한 최대 전투력 그래프를 그려보면 왼쪽과 같이 불연속 점들이 나타난다.
- 차라리 한 명을 덜 고용하고 다른 쪽을 여러 명 더 고용하는 것이 최선일 수도 있다.

C. 황금 칵테일

string, sort, brute_force

First Solve : 무한루프탈출 (17 min)

출제자 : rhdqor213

예상 난이도 : Easy

C. 황금 칵테일

- 문자열 비교를 통해 이미 들어간 이름의 재료라면 있는 값에 더해주고, 아니라면 이름 목록에 추가시켜주는 방식으로 각 재료가 얼마나 들어갔는지 계산한다. 최악 $O(|S|N^2)$
 - $\text{floor}(A_i * 1.618) = A_j, i \neq j$ 를 만족하는 (i, j) 쌍이 있는지 찾는다. $O(N^2)$
- Cf) $i \neq j$ 를 체크해주지 않으면 $\text{floor}(1.618) = 1$ 이기에 1만큼 들어간 재료 하나만으로 Delicious!를 출력하게 된다.

D. 이 사람 왜 이렇게 1122를 좋아함?

implementation

First Solve : 라일락꽃향기말던이세계시골소년과링딩동링딩동디기손잡고러브다이브하며코딩테스트
에참여한이야기에대하여 (34 min)
출제자 : rasauq1122

예상 난이도 : Easy

D. 이 사람 왜 이렇게 1122를 좋아함?

- 준성이의 최애 수를 포함할 수 있는 범위의 최댓값과 최솟값만 추적하면 된다.
- up 질의응답은 최솟값을, down 질의응답은 최댓값을 변경시킬 수 있다.
- 최솟값과 최댓값이 같다면 “I got it!”을 출력하고,
최솟값이 최댓값보다 커지면 “Paradox!”을 출력한다.
- 최애 수를 찾은 이후에도 모순이 발견될 수 있다는 점에 유의하도록 하자.

E. 색종이와 공예

graph_traversal, greedy

First Solve : 경찰청쇠창살외철창살검찰청쇠창살쌍철창살 (64 min)

출제자 : rasauq1122

예상 난이도 : Medium

E. 색종이와 공예

- 두 가지 풀이가 존재한다.
- Flood Fill을 활용하여 해결하는 방법을 먼저 알아보자.

E. 색종이와 공예

- 알파벳 하나를 골라 인접한 알파벳 중 같은 알파벳에 대해서 탐색을 진행한다.

B	B	B	B	B	B	B	B	C	C
B	B	B	B	B	B	B	B	C	C
C	C	C	A	A	A	A	A	A	D
C	D	A	A	A	A	A	A	A	D
C	C	A	A	A	A	A	A	A	D
C	C	A	A	A	A	A	B	B	B
D	D	D	D	B	B	D	B	B	B
A	A	A	B	B	B	B	B	B	B

E. 색종이와 공예

- 접근한 알파벳의 개수가 빨간 선으로 감싼 직사각형의 넓이와 같은지 확인한다.

B	B	B	B	B	B	B	B	C	C
B	B	B	B	B	B	B	B	C	C
C	C	C	A	A	A	A	A	A	D
C	D	A	A	A	A	A	A	A	D
C	C	A	A	A	A	A	A	A	D
C	C	A	A	A	A	A	B	B	B
D	D	D	D	B	B	D	B	B	B
A	A	A	B	B	B	B	B	B	B

E. 색종이와 공예

- 아직 접근하지 않은 알파벳에 대해서 모두 같은 방식으로 확인한다.
- 직사각형이 아닌 색종이 조각이 하나라도 있다면 예쁜 색종이 공예품이 아니다.
- 다음은 greedy approach로 문제를 해결하는 법이다.

E. 색종이와 공예

- ㄱ, ㄴ, ㄷ, ㄹ 모양의 색종이 조각이 있는지 확인한다.
이 중 하나라도 있다면 예쁜 색종이 공예품이 아니다.

B	B	B	B	B	B	B	B	C	C
B	B	B	B	B	B	B	B	C	C
C	C	C	A	A	A	A	A	A	D
C	D	A	A	A	A	A	A	A	D
C	C	A	A	A	A	A	A	A	D
C	C	A	A	A	A	A	B	B	B
D	D	D	D	B	B	D	B	B	B
A	A	A	B	B	B	B	B	B	B

E. 색종이와 공예

- 예쁜 색종이 공예품을 귀납적으로 만드는 과정을 통해 증명할 수 있다.
- 세로길이가 1인 색종이 공예품은 항상 예쁜 색종이 공예품이다.
- 세로길이가 N 인 예쁜 색종이 공예품에 적절한 알파벳 문자열을 추가하여 세로길이가 $N + 1$ 인 예쁜 색종이 공예품을 만들 수 있다.

E. 색종이와 공예

- 예쁜 색종이 공예품의 마지막 행만 고려하여 알파벳 문자열을 추가해도 세로 길이가 $N + 1$ 인 예쁜 색종이 공예품을 만들 수 있다.

E. 색종이와 공예

- 한 색종이 조각에 대해서 예쁜 색종이 공예품을 만들기 위해 알파벳 문자열을 추가하는 경우는 다음 2가지 방법 중 하나다.

C	C	B	B	B	B	B	B	C	C
---	---	---	---	---	---	---	---	---	---

1. 색종이 조각과 동일한 알파벳 문자열을 추가한다.

C	C	B	B	B	B	B	B	C	C
		B	B	B	B	B	B		

2. 같은 알파벳을 하나도 포함하지 않은 알파벳 문자열을 추가한다.

C	C	B	B	B	B	B	B	C	C
		C	C	A	A	A	D		

E. 색종이와 공예

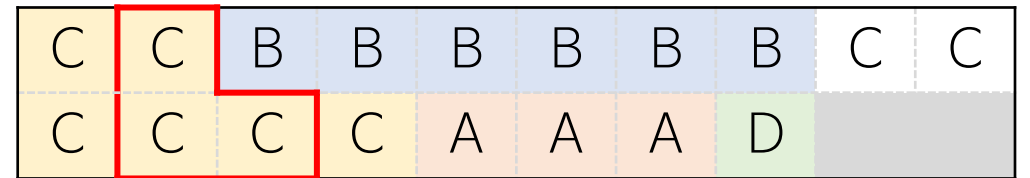
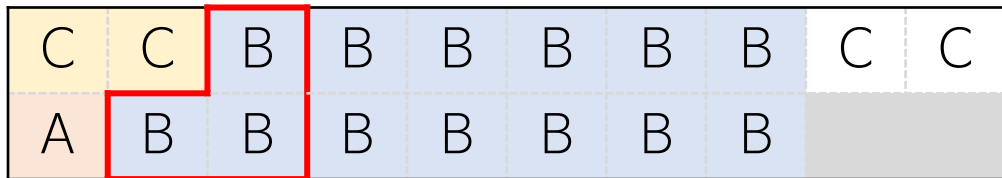
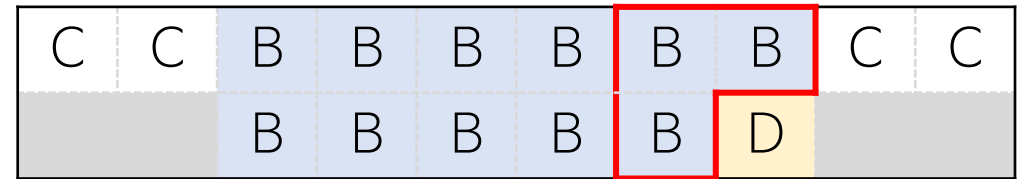
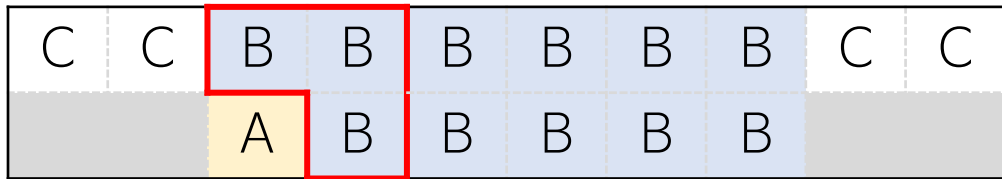
- 단, 인접한 색종이 조각에 대해 추가하는 알파벳 문자열끼리 동일한 알파벳으로 이어질 경우 예쁜 색종이 공예품이 되지 않을 수 있으므로 이에 유의한다.

C	C	B	B	B	B	B	B	C	C
A	B	B	B	B	B	B	B		

C	C	B	B	B	B	B	B	C	C
C	C	C	C	A	A	A	D		

E. 색종이와 공예

- 이 방법이 아닌 다른 방법으로 알파벳 문자열을 추가할 경우 반드시 Γ , \sqsubset , \sqsupset , \sqcap 모양 중 하나를 포함하는 색종이 조각이 생긴다.



F. 용 조련사 룰루

ad_hoc, greedy, constructive

First Solve : ??? (?? min)

출제자 : rasauq1122

예상 난이도 : **Medium**

F. 용 조련사 룰루

- 용을 각자의 힘이 내림차순이 되도록 정렬한다. 정렬한 용의 힘 수열을 B 라고 하자.
- 앞에서부터 탐색하여 인접한 두 용의 힘 차이가 처음으로 M 보다 커지는 지점을 찾는다.
- 해당 지점을 기준으로 힘이 큰 용들을 L그룹, 힘이 작은 용들을 R그룹이라 부르자.
- 수학적으로 엄밀한 정의는 다음과 같다:
 $B_i > B_{i+1} + M$ 을 만족하는 i 의 최솟값을 k 라고 하면, $B_1 \sim B_k$ 는 L그룹, $B_{k+1} \sim B_N$ 은 R그룹이라고 한다.

F. 용 조련사 룰루

- N 이 3 이하라면, 모든 용이 L그룹에 속하는 경우에만 용을 평화롭게 옮기는 방법이 존재한다.
- 그 외 경우, L그룹에 속한 용이 4 마리 이상인 경우에만 용을 평화롭게 옮기는 방법이 존재한다.

F. 용 조련사 룰루

- 첫 번째로 옮기는 용은 L그룹에 속한 용(이하 'L용')이어야 한다.
첫 번째로 옮기는 용이 R그룹에 속한 용(이하 'R용')이라면, 언젠가 L용을 옮길 때 학살이 일어난다.
- 두 번째로 옮기는 용 또한 L용이어야 한다.
R용을 옮기게 되면, 즉시 칼바람에서 학살이 일어나기 때문이다.
- 두 마리의 용을 옮긴 후, 협곡에서 학살이 일어나지 않기 위해서는 L용이 2마리 이상 남아야 한다.
1마리 이하로 남았다면 옮기는 과정에서 협곡에서 학살이 일어나기 때문이다
- 반드시 옮겨야 하는 용 2마리 + 남아야 하는 용 2마리로, L용은 최소 4마리 이상이어야 한다.

G. map, filter

implementation, binary_search

First Solve : ??? (?? min)

출제자 : hyoseok

예상 난이도 : **Hard**

G. map, filter

- map 연산으로 진행하는 곱셈과 덧셈 연산을 빠르게 적용시키기 위해 두 변수를 관리한다.
- $a \times A_i + b$ 의 형태로 처음에는 $a = 1, b = 0$ 이다.
- 덧셈 연산의 경우 $b = b + x$ 를 해준다.
- 곱셈 연산의 경우 $a = a \times x, b = b \times x$ 를 해준다.

G. map, filter

- map 연산으로 진행하는 나머지 연산의 경우는 새로운 배열을 만든다.
- x 의 범위가 최대 100이므로 길이 100의 배열에 나머지 연산한 결과의 개수를 저장한다.
- 그 후 관리하던 두 변수를 다시 초기화 하고 동일하게 진행한다.

G. map, filter

- filter 연산을 처리하는 데는 두 가지 경우가 존재한다.
- map 연산에서 나머지 연산을 진행했으면 길이 100 배열을 전부 탐색해 구한다.
- 아니면 이분 탐색을 이용해서 구한다. 처음에 배열을 정렬해두면 덧셈과 곱셈을 진행해도 배열이 정렬된 상태로 유지되기 때문에 가능하다.
- 시간 복잡도는 $O(q \log n)$ 이다.

H. 가장 작은 수

data_structures, number_theory

First Solve : ??? (?? min)

출제자 : rasauq1122

예상 난이도 : **Hard**

H. 가장 작은 수

- 어떤 수의 약수의 개수를 알기 위해서는 먼저 그 수를 소인수분해를 해야 한다.
 p_i 는 i 번째로 작은 소수라고 하면 정수 A 는 다음과 같이 소인수분해를 할 수 있다.

$$A = p_1^{a_1} \times p_2^{a_2} \times \cdots \times p_i^{a_i} \times \cdots$$

- 이때 A 의 약수의 개수는 $(a_1 + 1) \times (a_2 + 1) \times \cdots \times (a_i + 1) \times \cdots$ 이다.

H. 가장 작은 수

- A 가 2^N 개의 약수를 갖기 위해서는 수열 a_i 는 $2^{k_i} - 1$ 의 형태이어야 한다.

$$\begin{aligned} & (a_1 + 1) \times (a_2 + 1) \times \cdots \times (a_i + 1) \times \cdots \\ &= (2^{k_1} - 1 + 1) \times (2^{k_2} - 1 + 1) \times \cdots \times (2^{k_i} - 1 + 1) \times \cdots \\ &= 2^{k_1+k_2+\cdots+k_i+\cdots} \end{aligned}$$

- 이때 수열 k_i 의 합은 N 이다.

H. 가장 작은 수

- $2^0 (= 1)$ 개의 약수를 갖는 수는 1로 유일하고, 1의 수열 k 의 모든 원소는 0이다.
- 2^N 개의 약수를 갖는 수 A 에 $p_i^{2^{ki}}$ 를 곱하여 2^{N+1} 개의 약수를 갖는 수 A' 을 얻을 수 있다.
- $p_i^{2^{ki}}$ 을 최소로 하게하는 i 의 값을 c 라고 하면 2^N 개의 약수를 갖는 수 중 가장 작은 수에 $p_c^{2^{kc}}$ 를 곱하여 2^{N+1} 개의 약수를 갖는 수 중 가장 작은 수를 귀납적으로 구할 수 있다.
- 위 방법은 heap을 활용하여 쉽게 구현할 수 있다.

H. 가장 작은 수

1. min heap에 가능한 모든 $p_i^{2^{k_i}}$ 을 넣는다. 초기 수열 k 의 모든 원소는 0이다.
2. 소수의 개수는 무한하므로 min heap에 넣을 소수의 한도를 정해야 한다.
계산 과정 중 2 000 003이 곱해지면 출력하는 수는 항상 0이 된다.
2 000 003은 소수이므로 2 000 003이하의 소수에 대해서만 $p_i^{2^{k_i}}$ 를 min heap에 넣으면 된다.
3. 아래 과정을 N 번 반복하여 2^N 개의 약수를 갖는 가장 작은 수를 구할 수 있다.
 1. min heap에서 $p_c^{2^{k_c}}$ 를 구하여 저장한다.
 2. $p_c^{2^{k_c}}$ 을 min heap에서 제거한 후, $p_c^{2^{k_c+1}}$ 을 min heap에 넣는다.
 3. 위 과정을 N 번 반복하여 저장한 수들을 모두 곱한 후 2 000 003으로 나눈 나머지를 출력한다.

I. 인생은 B와 D 사이의 C다.

dynamic_programming, graph_theory, tree

First Solve : ??? (?? min)

출제자 : rasauq1122

예상 난이도 : **Challenging**

I. 인생은 B와 D 사이의 C다.

- 높이가 h 인 포화 이진 트리(이하 'PBT')의 루트 정점과 그와 연결된 간선들을 제거하면 높이가 $h - 1$ 인 PBT 2개로 나누어진다.
- 주어진 트리를 높이가 h 인 PBT 형태로 만들기 위해서는 루트의 자식 중 정점을 2개 골라 그 정점을 루트로 하는 높이가 $h - 1$ 인 PBT 형태로 만들어야 한다.
- 따라서 주어진 트리를 높이가 h 인 PBT 형태로 만들기 위해 필요한 최소 헬스장 등록 기간(이하 '비용')은 자식 중 2개를 높이가 $h - 1$ 인 PBT 형태로 만들기 위해 필요한 비용의 합과 같다.
- 위와 같은 고찰로 점화식을 세워 dynamic programming을 통해 문제를 해결할 수 있다.

I. 인생은 B와 D 사이의 C다.

- 주어진 트리를 높이가 0인 PBT 형태로 만들기 위해서는 루트가 아닌 모든 정점을 지워야 한다.
이때 비용은 루트의 모든 자식을 높이가 0인 PBT 형태로 만들 때 필요한 비용과
루트의 모든 자식을 다이어트하여 지울 때 필요한 비용의 합과 같다.
- 주어진 트리를 높이가 h ($h > 0$)인 PBT 형태로 만들 때 드는 최소 비용은
루트의 자식을 각각 높이가 $h - 1$ 인 PBT 형태로 만들 때 필요한 비용을 구한 후,
가장 작은 비용을 갖는 2개의 자식을 PBT 형태로 만든 후 다른 모든 정점은 삭제하는 비용과 같다.
- 위 방법을 재귀적으로 구현하여 주어진 트리를 높이가 h 인 PBT 형태로 만들 때 드는 최소 비용을 구할 수 있다.

I. 인생은 B와 D 사이의 C다.

- 주어진 트리를 PBT 형태로 만들기 위한 최소 비용의 최댓값을 구해보자.
- 주어진 트리에서 루트를 제외한 모든 정점을 제거하여 항상 PBT 형태로 만들 수 있다.
이때의 비용은 $(N - 1) \times d$ 와 같고 이 값은 최대 $111\ 221 \times 111\ 222 = 12\ 370\ 222\ 062$ 이다.
- 이를 통해 주어진 트리를 높이가 33보다 큰 PBT 형태로 만드는 것은 항상 손해라는 것을 알 수 있다.
높이가 33인 PBT의 정점은 $2^{34} - 1$ 개고, $2^{34} - (1 + N) > 12\ 370\ 222\ 062$ 이기 때문이다.
- 주어진 트리를 높이가 $0, 1, \dots, 32$ 인 PBT 형태로 만들 때 드는 최소 비용 중 최솟값이 문제의 정답이다.

J. 사기 주사위

brute_force

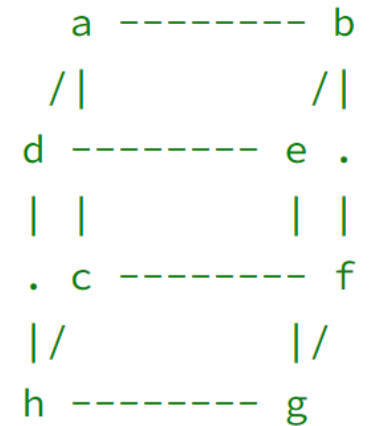
First Solve : ??? (?? min)

출제자 : mksmk606

예상 난이도 : **Medium**

J. 사기 주사위

- 정팔면체의 각 면을 정점으로 표현하고 인접한 면들을 간선으로 이으면 정육면체가 된다.
- 이때 나올 수 있는 조합을 모두 구한다. [[abcd], [baef], [daeh], [cafh], [hdcg], [edbg], [gcbg], [gehf]]
- A부터 h에 각각 1~8까지의 숫자를 대입해보고 입력으로 주어진 조합이 가능한 경우인지 확인한다.



K. 순열 구하기

ad_hoc, constructive

First Solve : ??? (?? min)

출제자 : hyoseok

예상 난이도 : **Medium**

K. 순열 구하기

순열 P:

4	2	3	1
4			
2	4		
2	3	4	
1	2	3	4

배열 B:

1	7	4	5
---	---	---	---

K. 순열 구하기

- 마지막에 xor 연산하는 배열은 무조건 1 ~ N까지 정렬된 배열이다.
- 그래서 1 ~ N까지 정렬된 배열과 xor 연산을 진행하면 배열의 마지막 값을 알 수 있다.

배열 B:

1	7	4	5
---	---	---	---

1	2	3	4
---	---	---	---

0	5	7	1
---	---	---	---

K. 순열 구하기

- 마지막 이전에는 방금 구한 값을 제외한 나머지 값들을 정렬한 배열과 xor 연산을 진행한다.
- 예시에서는 마지막 값이 1이었으므로 2, 3, 4와 xor을 해주면 된다.

배열 B:

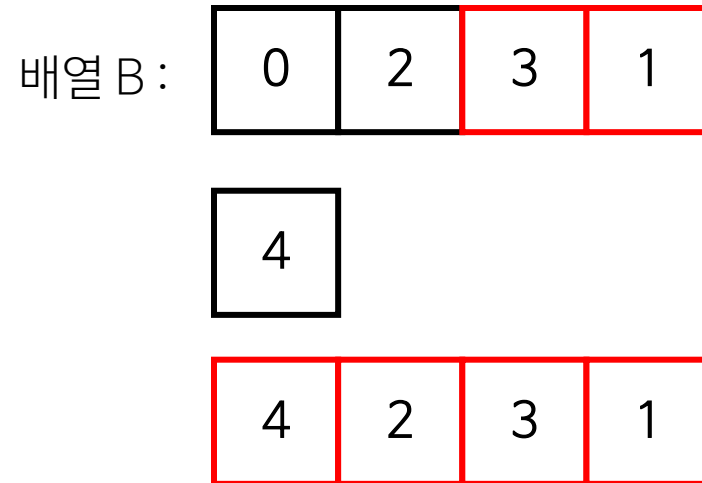
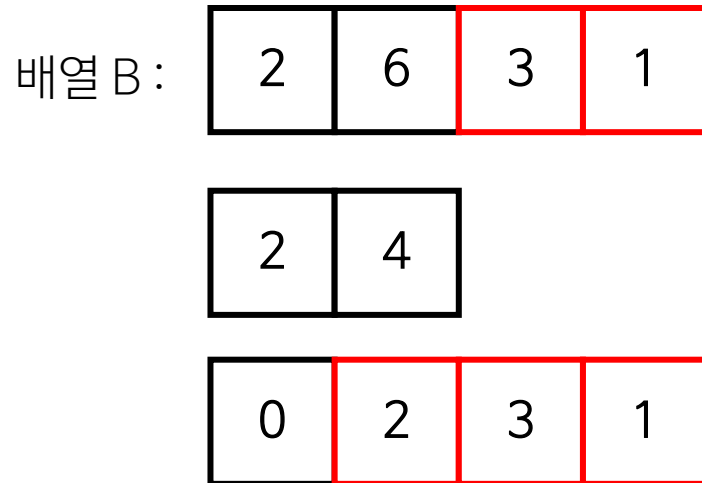
0	5	7	1
---	---	---	---

2	3	4
---	---	---

2	6	3	1
---	---	---	---

K. 순열 구하기

- 위의 과정을 N번 반복해서 원래의 순열 P를 구할 수 있다.
- 시간 복잡도는 $O(n^2)$ 이다.



L. 형광펜 강민우

mfmc

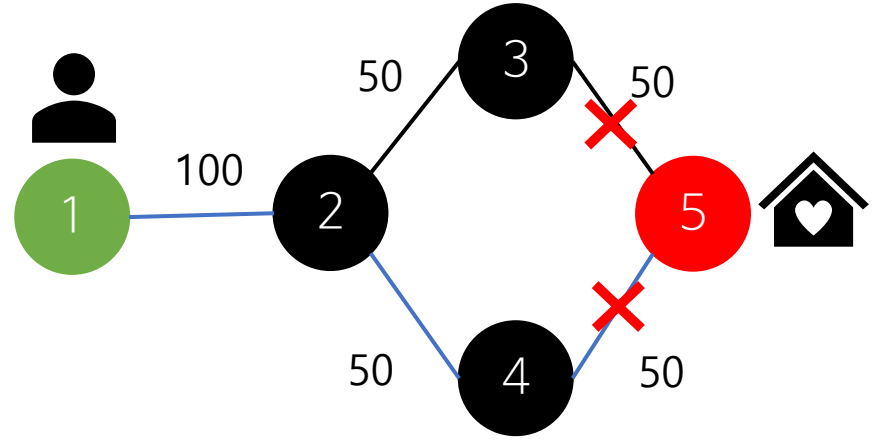
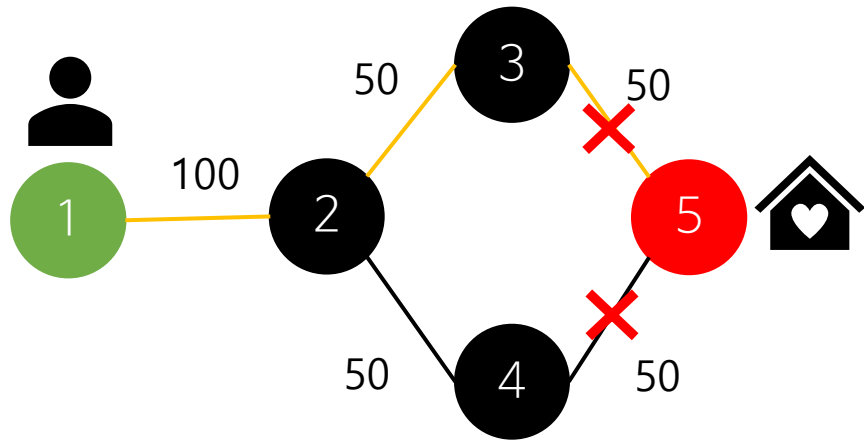
First Solve : ??? (?? min)

출제자 : mksmk606

예상 난이도 : **Challenging**

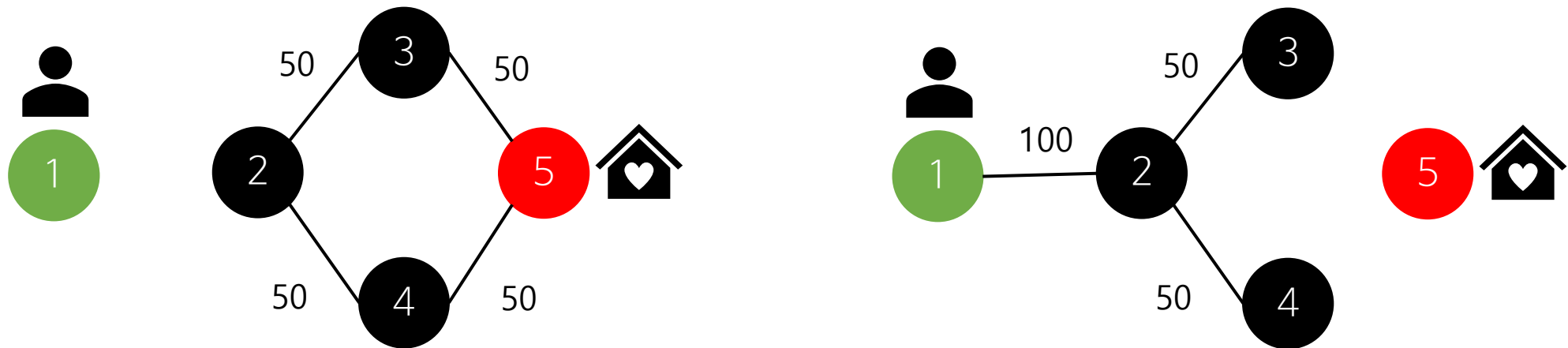
L. 형광펜 강민우

- 정훈이가 학회방에 가면서 한 번 이상은 현수막을 볼 수 있어야 한다. 즉, 현수막이 걸려있는 모든 길을 끊었을 때, 정훈이는 학회방에 갈 수 없다.



L. 형광펜 강민우

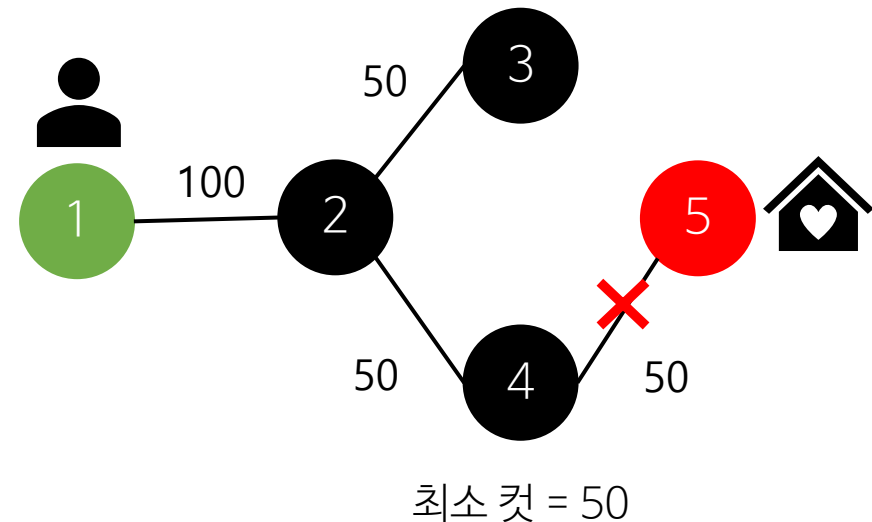
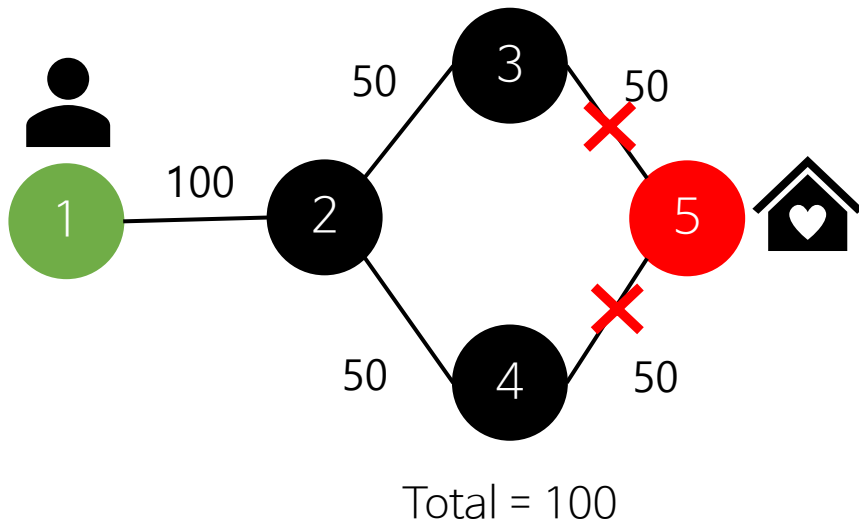
- 이때, 현수막을 설치하는 총비용은 최소가 되어야 한다. 즉, 정훈이가 있는 정점과 학회방이 있는 정점이 서로 다른 컴포넌트로 분리되면서, 끊는 간선의 비용 합이 최소(최소 컷)가 되어야 한다.



100의 비용으로 1번과 4번 정점을 분리할 수 있다.

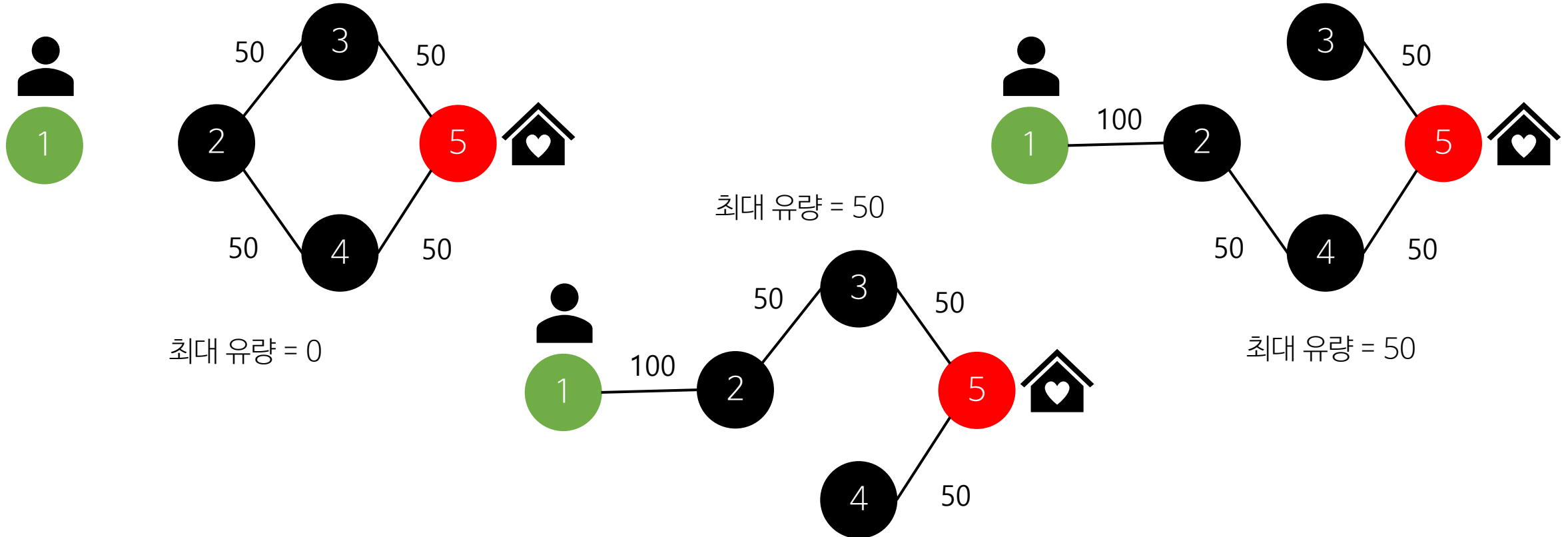
L. 형광펜 강민우

- 최대 유량 최소 컷 정리에 의해 최소 컷은 최대 유량과 같다. 따라서 각 간선의 용량을 현수막을 설치하는 비용으로 두고 최대 유량을 구한다.
- 최소 컷의 비용이 Total이라고 하자. 이때, 최소 컷에 속하는 간선 중 용량이 C인 간선 하나를 끊어주면 끊은 후의 그래프에서 최소 컷은 $Total - C$ 가 된다. 따라서 최대 유량도 C만큼 줄어든다.



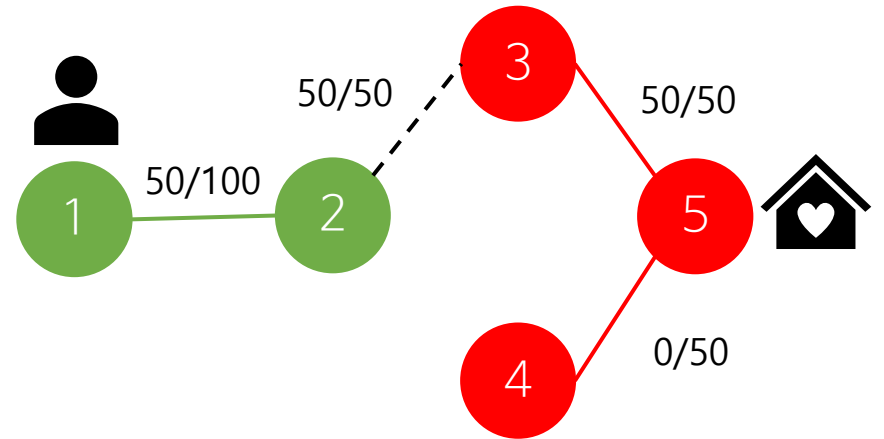
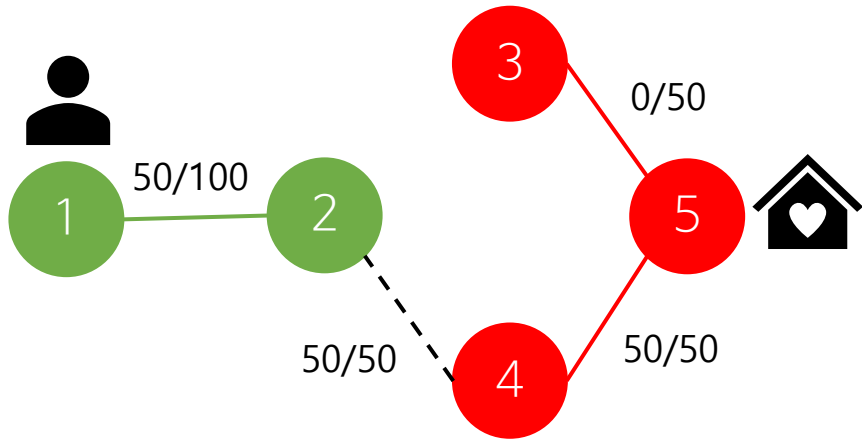
L. 형광펜 강민우

- 이를 이용하여 민우가 있는 건물과 인접한 간선을 끊었을 때 최대 유량이 그 용량만큼 줄어드는지 확인한다.



L. 형광펜 강민우

- Src에서 Sink로 유량이 더 흐르지 않는 상태에서 src에서부터 용량이 남은 간선을 탐색한다. 이 때 도달 가능한 정점들은 src와 연결된 정점들이며, 도달 불가능한 정점들은 sink와 연결된 정점들이다.
- 두 컴포넌트를 잇는 간선들이 최소 컷이다.



M. 이게 게임이냐?

Implementation, dynamic_programming, graph_traversal

First Solve : ??? (?? min)

출제자 : rhdqor213

예상 난이도 : **Hard**

M. 이게 게임이냐?

- 현재 게임의 상태는 각 더미에 해당하는 $1 \sim N$ 까지의 숫자 4개로 저장할 수 있다.
- 최초 상태 $\{0, 0, N + 1, N + 1\}$ (0은 오름차순 더미, $N + 1$ 은 내림차순 더미)
- 현재 상태에 대해서 2개의 카드를 내려놓는 순서, 각 카드를 내려놓는 더미 위치를 고려하여 행동할 수 있는 $2 \times 4 \times 4 = 32$ 번의 경우를 전부 시도해보고, 가능한 상태를 저장한다.
- 이는 N^4 의 공간에서의 DP 또는 BFS로 해결할 수 있다.