

가희와 함께 하는 7회 코딩테스트

문제	의도한 난이도	출제자
A 가희와 4시간의 벽 1	Easy	chogahui05
B 가희와 4시간의 벽 2	Easy	chogahui05
C 가희와 서울 지하철 2호선	Easy	chogahui05
D 가희와 부역명	Easy	chogahui05
E 가희와 전기 요금	Easy	chogahui05
F 가희와 클럽 오디션 3	Medium	chogahui05
G gahui and sousenkyo 7	Medium	chogahui05
H 타카하시의 카드 게임	Hard	chogahui05
I 가희와 전기 요금 2	Hard	chogahui05

문제		의도한 난이도	출제자
J	가희와 신칸센 1	Challenging	chogahui05
K	腹ペコのムト	Challenging	chogahui05
L	가희와 음악	Challenging	chogahui05

A. 가희와 4시간의 벽 1

implementation

출제진 의도 – **Easy**

- ✓ 제출 228, 정답 132명 (정답률 59.649%)
- ✓ 처음 푼 사람: **ksoosung77**, 0분
- ✓ 출제자: chogahui05

A. 가희와 4시간의 벽 1

- ✓ 항공편을 이용할 때 더 적게 걸리면 항공편을 이용해요.
- ✓ 그렇지 않으면 고속 철도를 이용합니다.
- ✓ 항공편을 이용할 때 걸리는 시간은 아래의 합으로 계산됩니다.
 - 공항 ~ 역간 이동 시간
 - 공항 a 에서 공항 b 로의 이동 시간
- ✓ 그런데, 공항 ~ 역간 이동 시간은 무시하라고 했어요. 고로, F_{ab} 와 S_{ab} 만 비교하면 됩니다.

B. 가희와 4시간의 벽 3

implementation

출제진 의도 – **Easy**

- ✓ 제출 213, 정답 113명 (정답률 53.991%)
- ✓ 처음 푼 사람: **dabbler1**, 2분
- ✓ 출제자: chogahui05

B. 가희와 4시간의 벽 2

- ✓ 공항 ~ 역간 이동 시간이 주어졌어요. 단순히 세트를 합산해서 계산했다면 틀립니다.
- ✓ 2개의 목록을 봅시다.
 - 첫 번째 조건은 너무 당연한 말을 하고 있어요.
 - 두 번째 조건은 철도를 좋아하는 사람에 대해 이야기 하고 있어요.
 - 이 사람은 고속 철도로 구간을 주파하는 데 4시간 이하가 걸리면 **항공편이 더 적게 걸리더라도** 고속철도를 이용하는 게 핵심입니다.
- ✓ 해당 부분 고려해서 조건을 잘 세우시면 정답을 받습니다.

B. 가희와 4시간의 벽 2

- ✓ 문제가 이렇게 나왔다면 어떨까요?
 - 철도를 좋아하는 사람은 역 a 에서 역 b 로 이동할 때 철도를 더 많이 이용합니다.
 - 그렇지 않으면 비행기를 더 많이 이용합니다.
 - 역 a 를 이용하는 사람은 철도를 좋아하는 사람이 더 많습니다.
 - 역 a 에서 역 b 로 이동할 때 어느 교통수단을 더 많이 이용할까요?
- ✓ 올바른 문제일까요? 아닙니다. 왜?

B. 가희와 4시간의 벽 2

- ✓ 극단적인 케이스를 생각해 봅시다.
 - 철도를 좋아하는 사람은 52명, 그렇지 않은 사람 48명이 있습니다.
 - 철도를 좋아하는 사람 중 30명은 역 b 로 갈 때 기차를 이용한다 합시다.
 - 나머지는 모두 비행기를 이용한다면 비행기를 철도보다 많이 이용합니다.

B. 가희와 4시간의 벽 2

- ✓ 또 다른 극단적인 케이스를 생각해 봅시다.
 - 철도를 좋아하는 사람은 52명, 그렇지 않은 사람 48명이 있습니다.
 - 철도를 좋아하지 않는 사람 중 30명이 비행기를 이용합니다.
 - 나머지는 철도를 이용합니다. 이 경우도 잘못된 문제에서는 가능합니다.
- ✓ 따라서 역 b 로 갈 때 철도를 더 많이 이용합니다.

B. 가희와 4시간의 벽 2

- ✓ 어? 그러면 사람들은 뭘 더 많이 이용하나요? 알 수 없습니다.
- ✓ s 를 t 보다 더 많이 이용한다는 상태가 딱 하나로 떨어지지 않기 때문입니다.
 - 10% *point* 만큼 더 많이 이용한다던지
 - 2명이 더 많이 이용한다던지
- ✓ 고로, 극단적인 케이스를 넣었을 때 서로 다른 결과가 나왔고, 잘못된 문제가 되었습니다.

B. 가희와 4시간의 벽 2

- ✓ 그래서 문제는 어떻게 나왔을까요?
 - 조건 c 를 만족하면 고속철도를 **이용한다**고 주었습니다.
 - 100% 이용한다는 것이 핵심입니다.

C. 가희와 서울 지하철 2호선

implementation, math

출제진 의도 – **Easy**

- ✓ 제출 206 번, 정답 109명 (정답률 56.796%)
- ✓ 처음 푼 사람: **dabbler1**, 4분
- ✓ 출제자: chogahui05

C. 가희와 서울 지하철 2호선

- ✓ 순환선의 역은 총 43개입니다.
- ✓ 고로 인접역간 구간도 43개입니다. *same*은 나올 수 없겠네요.
- ✓ 역 s 에서 e 로 이동한 경우 $s < e$ 인 경우와 $s > e$ 인 경우가 있습니다.

C. 가희와 서울 지하철 2호선

- ✓ $s = 216$ 이고 $e = 228$ 인 경우를 생각해 봅시다.
 - 이 경우 내선 순환은 $228 - 216$ 개의 구간을 지난 것입니다.
 - 외선 순환이 지난 구간 개수는 43 에서 12 를 빼면 되겠네요.
- ✓ $s = 201$ 이고 $e = 243$ 인 경우는 어떨까요?
 - 이 경우 내선 순환은 $243 - 201$ 개의 구간을 지난 것입니다.
 - 외선 순환이 지난 구간 개수는 43 에서 42 를 빼면 되겠네요.

C. 가희와 서울 지하철 2호선

- ✓ 반대는 어떨까요? $s = 243$ 이고, $e = 201$ 인 경우를 생각해 봅시다.
 - 이 경우 외선 순환은 $243 - 201$ 개의 구간을 지난 것입니다.
 - 내선 순환이 지난 구간 개수는 43 에서 42 를 빼면 되겠네요.
- ✓ 수학적으로 계산하는 방법 말고 일일이 인접역 시뮬레이션을 돌리는 방법도 있습니다.
- ✓ 201 번과 243 번 역만 조심하면 됩니다.

D. 가희와 부역명

implementation, string

출제진 의도 – **Easy**

- ✓ 제출 172, 정답 112명 (정답률 71.512%)
- ✓ 처음 푼 사람: **wizardrabbit**, 1분
- ✓ 출제자: chogahui05

D. 가희와 부역명

- ✓ 소괄호 안에 있는 것은 부역명입니다.
 - 여는 괄호의 위치를 lo 닫는 괄호의 위치를 $lo2$ 라고 합시다.
 - lo 는 문자열에서 *find* 함수 같은 것으로 구하면 됩니다.
 - $lo2$ 는 괄호가 나오는 경우 문자열 길이만 알면 바로 나옵니다.
 - 고로 *substring* 등으로 처리하시면 됩니다.

D. 가희와 부역명

- ✓ 이런 방법도 있습니다.
 - 여는 괄호와 닫는 괄호를 #으로 치환합니다.
 - 이제 #을 구분자로 해서 *tokenize*를 하면 됩니다.
 - 토큰이 하나면 부역명이 없는 것이고, 둘 이상이면 있는 것입니다.

E. 가희와 전기 요금1

implementation, math

출제진 의도 – **Easy**

- ✓ 제출 204, 정답 60명 (정답률 31.373%)
- ✓ 처음 푼 사람: **dabbler1**, 8분
- ✓ 출제자: chogahui05

E. 가희와 전기 요금 1

- ✓ a 와 m 을 1000과 60으로 뒤 봄시다. 두 수의 곱은 60000입니다.
- ✓ 1000W인 가전제품을 60분 사용했을 때 사용한 전력량이 $1kWh$ 입니다.
- ✓ 이에 대한 요금은 105.6원입니다.

E. 가회와 전기 요금 1

- ✓ 요금이 일정하다? 이 말은 비례식을 세울 수 있다는 의미입니다.
- ✓ a 와 m 의 곱 am 은 사용량입니다. 사용량이 60 000일 때 105.6 원이었다면
- ✓ am 만큼 이용했을 때 x 원입니다. 우리는 이 x 를 구하고 싶은 겁니다.
- ✓ $60000 : 105.6 = am : x$
- ✓ 위 비례식을 풀어서 x 를 구하면 됩니다.

F. 가희와 클럽 오디션 3

dynamic_programming, math

출제진 의도 – **Medium**

- ✓ 제출 49, 정답 31명 (정답률 65.306%)
- ✓ 처음 푼 사람: **dabbler1**, 13분
- ✓ 출제자: chogahui05

F. 가희와 클럽 오디션 3

- ✓ 먼저 lv 개의 방향키 중 c 개의 역방향 키가 등장한다고 해 봅시다.
 - lv 개 중에 역방향 키가 들어갈 c 개를 뽑습니다. 나머지 정방향 키겠네요.
 - 역방향 키를 채우는 가짓수는 8^c 입니다.
 - 정방향 키를 채우는 가짓수는 8^{lv-c} 입니다.
- ✓ 이 셋을 모두 곱하면 됩니다.

F. 가희와 클럽 오디션 3

- ✓ 역방향 키와 정방향 키의 개수가 같다는 점을 이용해서 아래와 같이 단순화 할 수도 있습니다.
 - lv 개의 자리에 8개의 패턴 중 하나를 채웁니다. 8^{lv} 가지입니다.
 - lv 개의 자리 중 역방향 키가 들어갈 c 개를 뽑습니다.
- ✓ 이제, 역방향 키가 0개인 경우부터 6개인 경우까지 모두 더해주면 됩니다.
- ✓ lv 이 5 이하인 경우 답은 4^{lv} 입니다.

G. gahui and sousenkyo 7

constructive

출제진 의도 – **Medium**

- ✓ 제출 119번, 정답 16명 (정답률 14.286%)
- ✓ 처음 푼 사람: **dabbler1**, 20분
- ✓ 출제자: chogahui05

G. gahui and sousenkyo 7

- ✓ 조건이 꽤 까다로웠을 것이라 생각합니다.
- ✓ 다음 조건을 모두 만족하는 정수 r 들만 종이에 모두 적었다는 무슨 말일까요?
- ✓ 예를 하나 들어봅시다. 아래 두 조건을 모두 만족하는 정수들을 종이에 적어봅시다.
 - 1 보다 크거나 같고 10 보다 작거나 같습니다.
 - 홀수입니다.
- ✓ 1, 3, 5, 7, 9가 적히겠네요.
- ✓ 왜 2는 종이에 적히지 않았을까요? 홀수가 아니기 때문입니다.

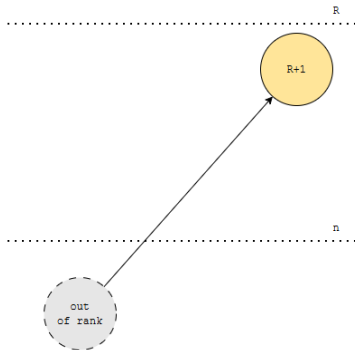
G. gahui and sousenkyo 7

- ✓ 다음 상위 r 위 안에 드는 캐릭터는 바뀌지 않았다는 무슨 말일까요?
- ✓ 아래 케이스를 생각해 봅시다.
 - 작년에 캐릭터 1, 2, 3 이 각각 1, 2, 3위를 했다고 해 봅시다.
 - 올해에 캐릭터 3, 1, 2 이 각각 1, 2, 3위를 했다고 해 봅시다.
 - ▶ 상위 1 위 안에 드는 캐릭터는 바뀌었나요? 네.
 - ▶ 상위 2 위 안에 드는 캐릭터는 바뀌었나요? 네. 2가 없기 때문입니다.
 - ▶ 대신 상위 3인은 1, 2, 3 그대로입니다.
 - ▶ 고로 상위 r 위가 바뀌지 않았다는 것은 상위 r 인의 집합이 안 바뀌었다는 것이 됩니다.

G. gahui and sousenkyo 7

- ✓ 이 문제에서는 어려운 케이스가 2개입니다.
 - 제일 큰 r 을 R 이라 할 때, $R < n$ 인 경우
 - 상위 k 인을 어떻게 만들까?
- ✓ 먼저 1 번부터 해결해 봅시다. 벽은 부숴버리라고 있는 것이라는 힌트가 있습니다.
- ✓ 그러면 권외에서 순위가 엄청나게 똥 것도 벽을 깬 것이네요?

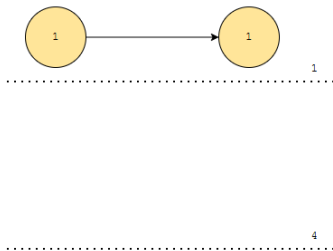
G. gahui and sousenkyo 7



- ✓ 이 경우 $R < r$ 인 경우 상위 r 인은 변했습니다.
- ✓ 권외였던 캐릭터가 $R + 1$ 위를 했기 때문입니다. 1 번째 케이스 해결

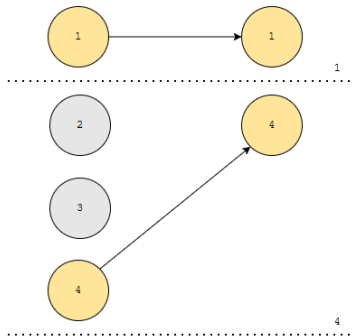
G. gahui and sousenkyo 7

- ✓ 2번째 케이스는 1은 조건을 만족하는데 2와 3은 만족하지 않고 4는 만족하는 경우입니다.
- ✓ 상위 1인, 4인은 그대로죠? 2인과 3인은 변했습니다.
- ✓ 이걸 어떻게 처리할 것인가? 1인과 2 ~ 4인을 묶을 거예요



G. gahui and sousenkyo 7

- ✓ 이제 4위였던 캐릭터를 2위로 올려봅시다.
- ✓ 그러면 상위 2인과 3인은 바뀝니다. 왜? 4위가 올라왔기 때문입니다.



G. gahui and sousenkyo 7

- ✓ 일반화를 시켜서 r 들이 오름차순 정렬이 되어 있었다고 해 봅시다.
- ✓ r_i 다음에 r_{i+1} 이 조건을 만족하는 수일 때
 - r_{i+1} 위였던 캐릭터를 $r_i + 1$ 위로 올립니다.
 - 나머지 캐릭터들은 1 위씩 강등시키면 됩니다.
 - 혹은 역으로 $r_i + 1$ 위였던 캐릭터를 r_{i+1} 위로 내리고
 - 나머지 캐릭터들을 1 위씩 올리면 됩니다.

H. 타카하시의 카드 게임

math

출제진 의도 - **Hard**

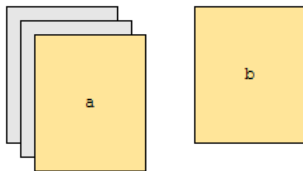
- ✓ 제출 50, 정답 19명 (정답률 38.000%)
- ✓ 처음 푼 사람: **sehujeong**, 24분
- ✓ 출제자: chogahui05

H. 타카하시의 카드 게임

- ✓ 항상 승리한다는 말이 나오면 게임 이론을 생각하시는 경우가 있습니다.
- ✓ 이 문제도 그럴까요? 게임을 간단하게 봅시다.

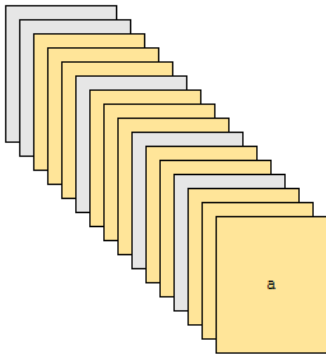
H. 타카하시의 카드 게임

- ✓ 카드를 3개 이상 뽑았다고 해 보겠습니다.
- ✓ 이 중 어느 두 개의 카드에 다른 문자가 쓰여져 있다면 어떨까요?
- ✓ 타카하시가 a 가 적혀져 있는 카드를 뽑고 b 가 적혀져 있는 카드를 뽑으면 끝납니다.



H. 타카하시의 카드 게임

- ✓ 반면에 같은 문자가 써져 있는 카드들만 넣었다면
- ✓ 타카하시가 어떻게 뽑아도 가희가 승리하게 됩니다.



H. 타카하시의 카드 게임

- ✓ 문자 c 가 k 회 나왔다고 합시다. 그러면
 - 문자 c 가 들어가 있는 카드만 하나 이상 뽑는 가지수는 $2^k - 1$ 이 됩니다.
 - 하나만 뽑는 가지수는 k 가 됩니다.
 - 두개만 뽑는 가지수는 $\binom{k}{2}$ 가 됩니다.
- ✓ 고로, u 가 작다면, u 개 이상 뽑는 가지수는 여사건으로 빠르게 구할 수 있겠네요.

H. 타카하시의 카드 게임

- ✓ 카드를 2개 뽑았다고 해 보겠습니다.
- ✓ 이 경우는 어떨까요? 타카하시가 카드를 뽑고 나서 랜덤 박스에 1개 이상의 카드가 남아야 해요.
- ✓ 고로, 타카하시는 1개밖에 못 가져갑니다.
- ✓ 가희가 2개를 넣은 경우 타카하시가 어떻게 가져가더라도 항상 가희가 승리하겠네요.



H. 타카하시의 카드 게임

- ✓ 정리해 보면 다음 경우들을 모두 더하면 됩니다.
 - 전체 카드 중 2개를 뽑는 가지수
 - 각각의 문자 c 에 대해, c 가 적혀져 있는 카드 중 3개 이상을 뽑는 가지수
 - 3은 작으므로 **여사건**을 이용하면 됩니다.

I. 가희와 전기 요금 2

dynamic_programming

출제진 의도 - **Hard**

- ✓ 제출 24, 정답 15명 (정답률 62.500%)
- ✓ 처음 푼 사람: **jeoffrey**, 29분
- ✓ 출제자: chogahui05

I. 가희와 전기 요금 2

- ✓ i 번째 컴퓨터까지 고려했을 때, 요금이 x 가 나오는 가짓수를 $dp[i][x]$ 로 정의할 수 있습니다.
- ✓ 만약 i 를 사용했을 때 요금이 c 라고 해 봅시다. 그러면 $dp[i][x]$ 는 어떻게 구할 수 있을까요?
 - $x \geq c$ 라면 $dp[i][x] = dp[i-1][x] + dp[i-1][x-c]$
 - 그렇지 않으면 $dp[i][x] = dp[i-1][x]$ 가 됩니다.
 - 그런데 이 c 가 커질 수 있는 경우가 문제입니다.

I. 가희와 전기 요금 2

- ✓ 여기서 핵심인 것은 c_1 과 c_2 가 300 000 을 안 넘는다는 것입니다.
- ✓ 고로 dp 상태의 개수는 매우 적습니다.
- ✓ 만약 컴퓨터를 사용했을 때 전기 요금이 300 000 원을 넘는다면 커팅하는 방법도 괜찮습니다.

J. 가희와 신칸센 1

implementation, unit_test, test

출제진 의도 - Challenge

- ✓ 제출 -, 정답 -명 (정답률 -%)
- ✓ 처음 푼 사람: -, -분
- ✓ 출제자: chogahui05

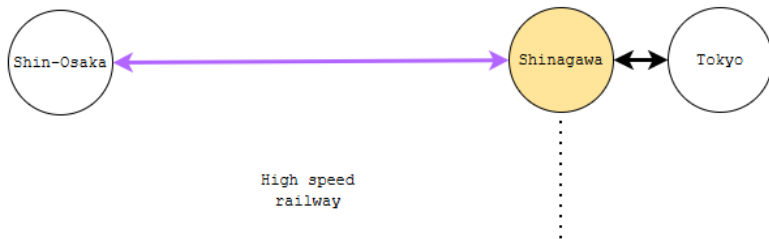
J. 가희와 신칸센 1

- ✓ 눈치 채신 분들이 있을 수도 있습니다만
- ✓ 이 문제의 원래 버전은 *Tokyo*와 *Hakata* 간의 구간이 원본이었습니다.
- ✓ *Tokyo*에서 *Shin - Osaka* 간의 구간으로 줄인 것이 이 버전입니다.

J. 가희와 신칸센 1

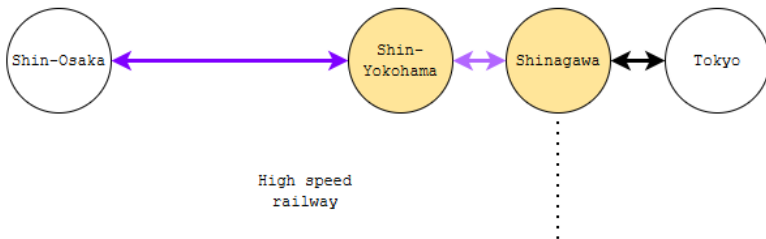
- ✓ 조건이 굉장히 많고 복잡한 데다가, 사전지식으로 절대 풀 수 없게 만들었습니다.
- ✓ 사전 지식으로 알 수 있는 것은 도카이도 신칸센 운임 전 까지이니까
- ✓ 도카이도 신칸센 운임에 대해 읽어봅시다.

J. 가희와 신칸센 1



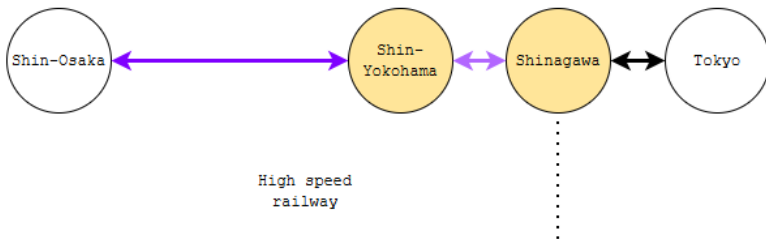
- ✓ 먼저 *Shinagawa*에서 *Tokyo*는 일반선 요금이 적용된다고 했어요.
- ✓ 그 외의 경우 고속선 요금이 적용됩니다. 이를 그림으로 나타내면 위와 같습니다.

J. 가희와 신칸센 1



- ✓ 그런데 고속선 중에서 *Shinagawa*에서 *Shin - yokohama* 간 구간은 특별합니다.
- ✓ 왜? 어느 등급이나 *kodama*의 요금을 받기 때문입니다.

J. 가희와 신칸센 1

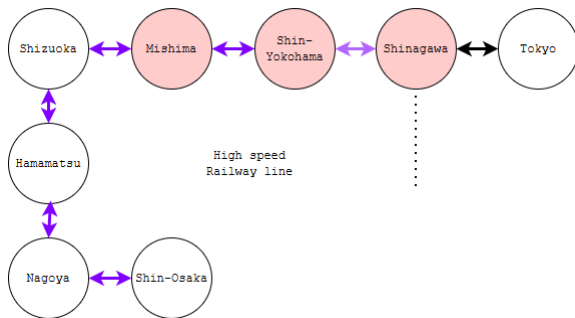


- ✓ 맨 밑에 고속선 구간만 타는 경우 33 000 원 이상이고
- ✓ 그렇지 않으면 5 000 원 이상을 만족한다는 조건도 메모해 둡시다.

J. 가희와 신칸센 1

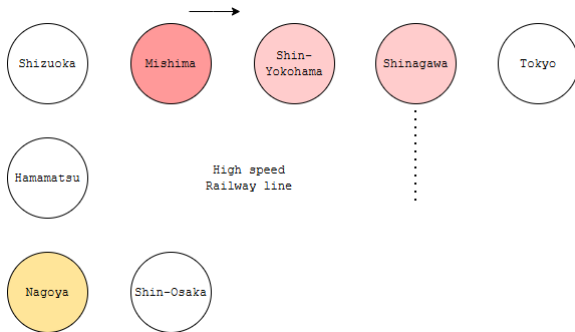
- ✓ 저 두 구간은 짧기 때문에 요금을 계산해 봅시다.
 - *Tokyo*에서 *Shinagawa* 간 요금은 1 360 원입니다.
 - *Shinagawa*에서 *Shin – yokohama* 간 요금은 6 000 원입니다.
- ✓ 고로 *Tokyo*에서 *Shin – yokohama* 간 요금은 7 360 원입니다.

J. 가희와 신칸센 1



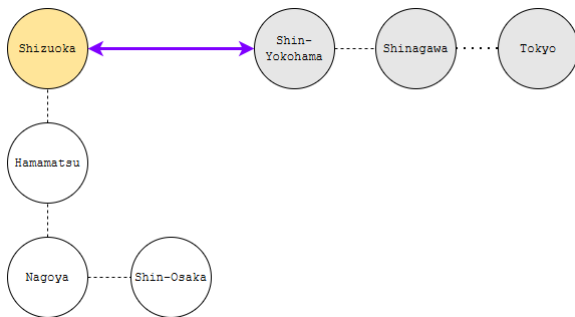
- ✓ 다음에 시발, 종착을 할 수 있는 역이 나오니까 이들도 적어 둡시다.
- ✓ 여기서 붉게 표시한 건 시발만 가능한 역입니다.

J. 가희와 신칸센 1



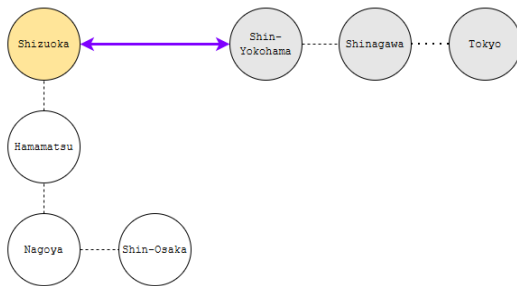
- ✓ 특이한 조건이 붙은 역들을 봅시다. *Mishima* 는 *Tokyo* 방향으로만 운행해요.
- ✓ *Nagoya* 에서 시발하거나 종착하면 *Shin - Osaka* 역에 시발하거나 종착하지 않아요.

J. 가희와 신칸센 1



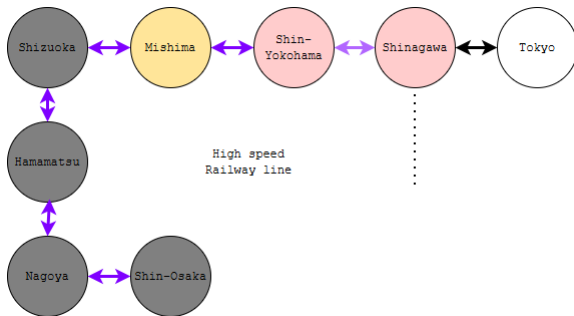
- ✓ 고로 *Tokyo* 부터 따져주면 될 것인데요.
- ✓ *Tokyo* 에서 *Shin - Yokohama* 까지는 멀지 않고, *Shizuoka* 까지 거리가 멉니다.

J. 가희와 신칸센 1



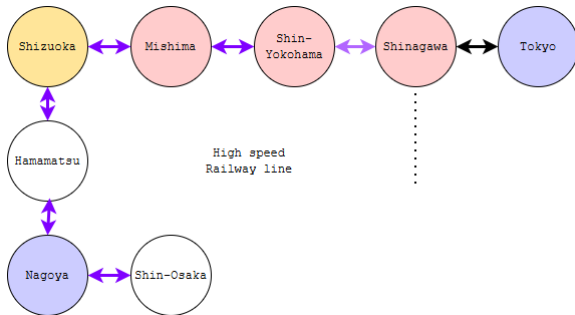
- ✓ *Shizouka*에서 *Shin - Yokohama*까지 운임 거리는 150km가 넘습니다.
- ✓ *kodama* 등급의 요금은 45 000원이 넘습니다.
- ✓ 고로 *Tokyo*역부터 *Shin - Yokohama*역은 *Shin - Osaka*방면으로 *Shizuoka*이후의 역들은 모두 종착 가능합니다.

J. 가희와 신칸센 1



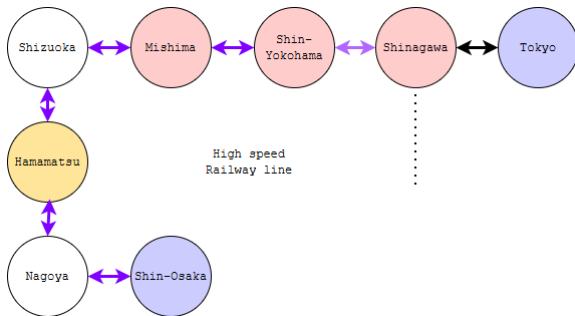
- ✓ 다음 *Mishima* 에서 시발하면 동행만 가능하다 되어 있습니다.
- ✓ *Nagoya* 기준으로 *Tokyo* 방면으로 가면 동행이라 했으니까 *Tokyo* 로만 갈 수 있네요.

J. 가희와 신칸센 1



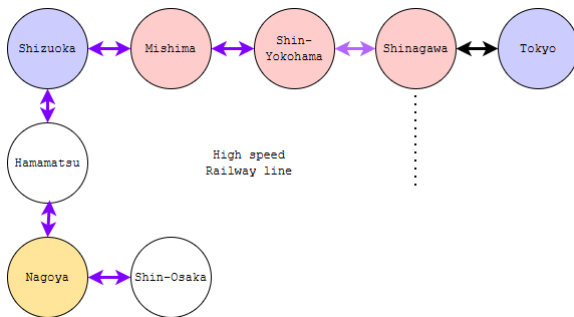
- ✓ *Shizuoka* 는 어떤가요? 일단 *Tokyo* 까지는 충분히 멀니다.
- ✓ 반대 방향으로 가 볼까요? *Hamamatsu* 는 110km 보다는 가깝습니다. 고로 제외
- ✓ *Nagoya* 는 충분히 머니까 나고야 이서지역에 있는 *Shin - Osaka* 역시 종착할 수 있겠네요.

J. 가희와 신칸센 1



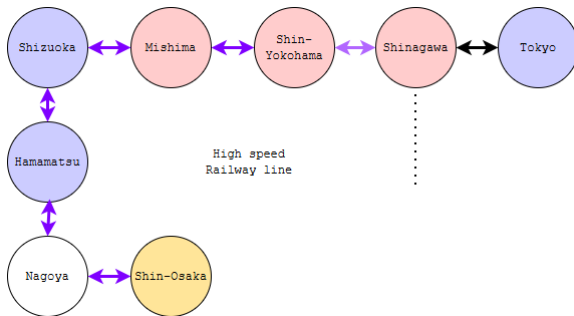
- ✓ 다음 *Hamamatsu*는 어떨까요? 당연히 *Shizuoka*는 가까워서 못 갑니다. *Tokyo*로는 갈 수 있어요.
- ✓ *Nagoya*는 어떨까요? 아슬아슬하게 $110km$ 보다 가깝기 때문에 못 갑니다.
- ✓ 대신 *Shin - Osaka* 역은 갈 수 있습니다.

J. 가희와 신칸센 1



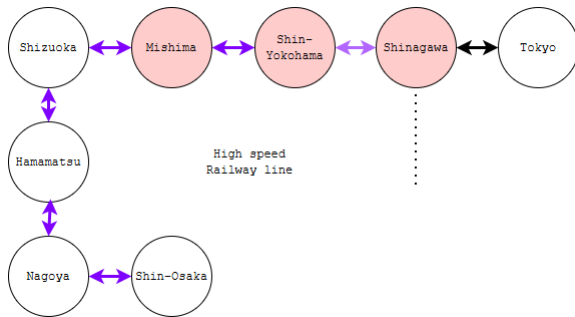
- ✓ 다음 *Nagoya*에서 출발하면 *Shin - Osaka* 역은 못 갑니다. 조건에 나와 있어요.
- ✓ 반대 방향으로 가면, *Shizuoka* 부터 갈 수 있어요.

J. 가희와 신칸센 1



- ✓ 다음 *Shin - Osaka* 역에서 시발하면 *Nagoya* 는 못 갑니다.
- ✓ 대신 *Hamamatsu* 부터 갈 수 있습니다. 이미 *Nagoya* 에서 *Shin - Osaka* 까지 거리가 *180km* 가 넘기 때문입니다.

J. 가희와 신칸센 1



- ✓ 이것만 따지면 되는가? *Shin - Yokohama* 역이 시발이 가능하да 했어요.
- ✓ *Tokyo* 방면으로 갈 때 요금을 따져 봐야죠. 7 360 원은 5 000 보다 큼니다.
- ✓ 따라서 가능합니다.

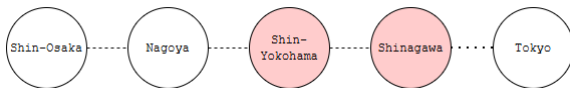
J. 가희와 신칸센 1

- ✓ 여기까지 오셨다면 *kodama* 패턴은 끝났습니다.
- ✓ 거치는 모든 역에 정차하므로 전역 정차합니다.
- ✓ 역 번호를 적당히 잘 주면 s 번 역부터 e 번 역까지 방문하는 *logic*을 설계할 수 있습니다.

J. 가희와 신칸센 1

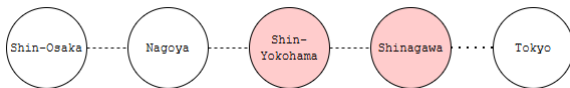
- ✓ 다음 *hikari*와 *nozomi* 패턴입니다.
- ✓ 이 둘은 통과 역이 있습니다. 게다가 시발하거나 종착할 수 있는 역도 5개밖에 없습니다. 왜?
- ✓ 시발하거나 종착이 가능한 역에서는
 - 항상 정차하는 등급의 열차들만
 - 시발 혹은 종착이 가능하다고 되어 있기 때문입니다.

J. 가희와 신칸센 1



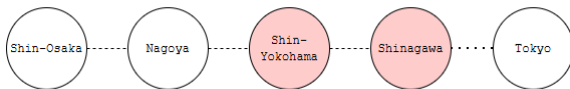
- ✓ 그림으로 그리면 위와 같습니다.
- ✓ 통과역이 있다고 했으니, *Shin - Yokomaha* 부터 *Tokyo*까지는 하나의 덩어리로 묶입니다.
- ✓ 게다가 덩어리 중 단 하나의 역만 종착이 가능하네요.

J. 가희와 신칸센 1



- ✓ 따라서 *Tokyo*나 *Shinagawa*나 *Shin - Yokomaha*에서 출발한다면
- ✓ *Nagoya*나 *Shin - Osaka* 역에서만 종착할 수 있습니다. 반대의 경우는?

J. 가희와 신칸센 1



- ✓ *Nagoya* 나 *Shin - Osaka* 에서 출발한다면
- ✓ *Tokyo* 역에서만 종착할 수 있습니다.
- ✓ *Nagoya* 에서 *Shio - Osaka* 나, *Shin - Osaka* 에서 *Nagoya* 구간은 조건상 될 수 없기 때문입니다.

J. 가희와 신칸센 1

- ✓ 이는 *nozomi*도 공통이니, 공통 로직으로 빼 버립시다. 그리고 아래 요소들은 어떨까요?
 - a 역 다음에 b 역으로 갔는데 필수 정차역을 통과했는지 검사는 가능합니다.
 - 또한 멈추지 말아야 할 역에 정차하는 경우를 체크하는 것도 공통으로 뺄 수 있어요. 왜?
 - 동그라미, 세모에 대해서 정보를 저장한 후에
 - 케이스에 따라 선택 정차역과 필수 정차역으로 분류하면 되기 때문입니다.

J. 가희와 신칸센 1

- ✓ 여기까지 하면 *nozomi*는 해결됩니다. *hiraki*는 다소 복잡합니다.
- ✓ 먼저 1구간과 2구간에 대해 패턴이 있다 하니 이것부터 봅시다.
 - 2구간 급행 조건이 더 적습니다.
 - *nozomi*와 같은 정차 패턴을 가지면 급행이라 했는데요.
 - 해당 등급은 *Nagoya ~ Kyoto*간 무정차입니다.
 - 고로 해당 패턴을 *hikari*가 가지면 급행입니다.

J. 가희와 신칸센 1

- ✓ 그런데 *Maibara* 역에 정차하지 않으면 *Gifu - Hashima* 역에 정차하지 않는다 되어있어요.
- ✓ 이 말은 역으로 *Gifu - Hashima* 역에 정차하면 *Maibara* 역에 정차함을 의미해요.
- ✓ 고로 *Maibara* 통과, *Gifu - Hashima* 정차 패턴은 없습니다.
- ✓ 고로 *Nagoya ~ Kyoto* 간 1개 역에 정차했는데 그 역이 *Gifu - Hashima* 라면 거르면 됩니다.

J. 가희와 신칸센 1

- ✓ 1 구간 급행 조건이 어지럽습니다. 셋 중 하나 이상 만족하면 됩니다.
 - 2개 이하의 역에 정차합니다.
 - 무정차 거리가 $200km$ 이상입니다.
 - 시즈오카 시에 있는 역에 정차하지 않습니다.

J. 가희와 신칸센 1

- ✓ 일단 시즈오카시에 있는 역은 *Shizuoka* 역 하나입니다. 이 역을 기준으로
 - *Shin - Yokohama*도 *Nagoya*도 200km 보다는 가깝게 있습니다.
 - *Shizuoka*는 *Nagoya*와 *Tokyo*의 중간 지점인 건 사전 지식.
 - 여러 가지 함정들 때문에 해당 사전지식을 취사 선택하는 게 쉽지 않았을 겁니다.

J. 가희와 신칸센 1

- ✓ 고로 1 구간 급행 조건은 둘 중 하나 이상을 만족하면 됩니다. 단순화 되었네요.
 - 2개 이하의 역에 정차합니다.
 - 시즈오카 시에 있는 *Shizuoka* 역에 정차하지 않습니다.

J. 가희와 신칸센 1

- ✓ 아니 어떻게 이렇게 된 것이지? 아래 두 조건이 있다고 하겠습니다.
 - s 는 개입니다.
 - s 는 동물입니다.
- ✓ 조건 1을 만족하면 2도 만족합니다. s 가 하나 이상의 조건을 만족하려면?
- ✓ 당연하게도 s 는 동물이면 됩니다.

J. 가희와 신칸센 1

- ✓ 이제 1 구간과 2 구간에서 둘 다 *Rapid*이면 거르면 됩니다.
- ✓ 1 구간만 있는 경우, *Rapid* 패턴을 거르면 됩니다.

J. 가희와 신칸센 1

- ✓ *ERROR* 코드가 여러 개 나올 수 있는 경우 가장 작은 것을 어떻게 출력할까요?
- ✓ *ERROR* 코드가 작은 것부터 잘라내시면 됩니다.
- ✓ 어려운 것 400을 봅시다.
- ✓ *Tokyo*에서 *Shin - Osaka* 방향으로 갈 때 증가하는 순으로 매겨봅시다.

J. 가희와 신칸센 1

- ✓ 진행 방향이 바뀌었다면 역 번호가 감소하다가 증가하거나
- ✓ 혹은 역 번호가 증가하다가 감소하였을 겁니다.
 - 역 s_1 의 번호와 s_0 의 번호의 차이 x_0 을 구합시다.
 - 다음에 역 s_i 의 번호와 s_{i-1} 의 번호의 차이 x_{i-1} 을 구합니다.
 - 이 두 수의 곱이 음수면 진행 방향이 바뀐 것입니다.

K. 腹ペコのムト

graph_theory, math

출제진 의도 – Challenge

- ✓ 제출 98, 정답 9명 (정답률 9.184%)
- ✓ 처음 푼 사람: **dabbler**, 50분
- ✓ 출제자: chogahui05

K. 腹ペコのムト

- ✓ 포켓몬이 나오고 ムト가 나옵니다.
- ✓ 무슨 십덕 문제인가? 사실 메이플과 포켓몬을 같이 낸 거라고 볼 수 있습니다.
- ✓ 배고픈 무토는 메이플에 있는 파티 퀘스트입니다. 여기서 재료를 모아야 합니다.
- ✓ 여기에 진화라는 키워드를 섞으니 자연스레 포켓몬 시리즈가 떠오르더라고요.
- ✓ 여담이지만 *TMS*에서는 무토를 武藤로 쓰던데, 자주 본 거 같습니다?
- ✓ 정말 메이플 무토는 싸움을 잘 하는 돌덩어리였을까요? 저도 궁금합니다.

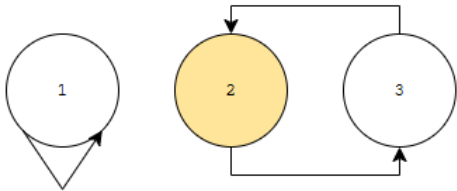
K. 腹ペコのムト

- ✓ 요약하면, 1 번 진화한 결과와 c 번 진화한 결과가 같은 재료의 개수를 Q 번 구합니다.
- ✓ 일단 재료 x 가 진화 1 번 하면 특정한 재료로 변합니다.
- ✓ *functional_graph* 로 해석하고 풀 수 있습니다. 여기까지는 아마 쉬웠을 겁니다.
- ✓ 최근 향유회에서도 나왔고, 건대 프로그래밍 대회에서도 나왔으니 인기있는 주제이기도 합니다.

K. 腹ペコのムト

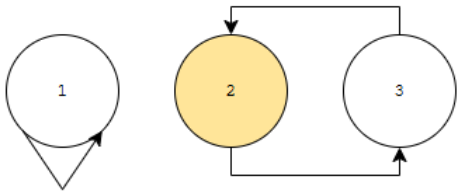
- ✓ 1번 진화한 결과와 c 번 진화한 결과가 같다는 말은 무슨 말일까요?
- ✓ 이는 사이클과 관련이 있다는 말이 됩니다. 사이클 하면 생각나는 것은 무엇인가요?
- ✓ 바로 주기입니다. 여기까지 말하면 잘 안 와닿으니, 1번 그래프를 봅시다.

K. 腹ペコのムト



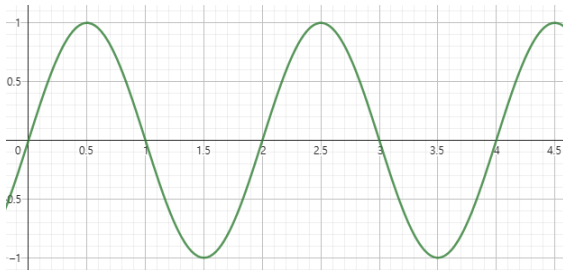
- ✓ 먼저 2번 재료가 1번 진화하면 3이 됩니다.
- ✓ 2번 재료가 2번 진화하면 2가 됩니다. 같은 재료가 됩니다.
- ✓ 2번 재료가 4번 진화하면 2가 됩니다. 같은 재료가 됩니다.

K. 腹ペコのムト



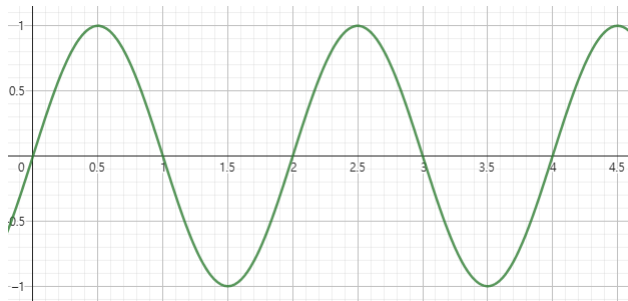
- ✓ 먼저 2번 재료가 2번 진화한 결과나 4번 진화한 결과가 왜 같은가?
- ✓ 이는 4가 2가 속한 사이클의 배수이기 때문입니다.

K. 腹ペコのムト



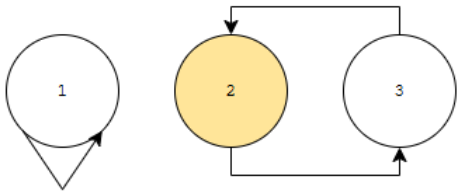
- ✓ 위 함수 $f(x)$ 를 봅시다. 주기가 2입니다.
- ✓ 왜? $f(x) = f(x + 2)$ 을 만족합니다. 이 2는..
- ✓ $f(x) = f(x + p)$ 를 만족하는 0보다 큰 가장 작은 정수입니다. p 에 2를 넣으면 딱 들어맞습니다.
- ✓ 0보다 큰 더 작은 정수를 넣으면 만족하지 않아요.

K. 腹ペコのムト



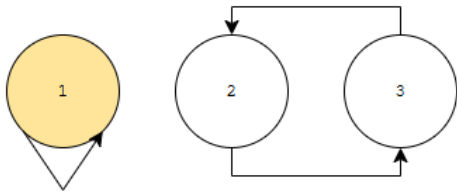
- ✓ 그런데 사실 이 함수를 x 축으로 4만큼 평행 이동시켜도 완전히 겹친다는 것을 알 수 있어요.
- ✓ 왜? 주기가 2이고 4는 2의 배수이기 때문입니다.
- ✓ 비슷한 논리를 여기서 적용하면 됩니다.

K. 腹ペコのムト



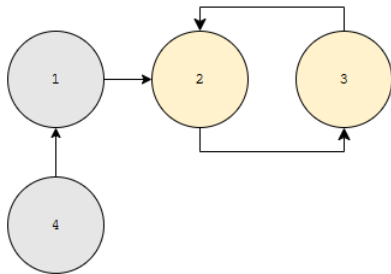
- ✓ 먼저 2번 재료가 2번 진화한 결과나 4번 진화한 결과가 왜 같은가?
- ✓ 이는 4가 2가 속한 사이클의 배수이기 때문입니다.

K. 腹ペコのムト



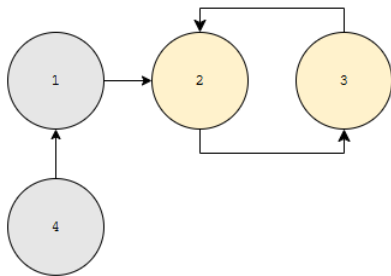
- ✓ 다음 1 번 재료가 1 번 진화한 결과는 1 이 나옵니다.
- ✓ 이는 10 번 진화한 결과도 똑같이 1 이 나옵니다. 왜?
- ✓ 당연히게도, 10은 사이클의 길이인 1의 배수이기 때문입니다.
- ✓ 여기까지는 그래도 쉽게 간파하셨을 것 같으니 다음으로 넘어갑시다.

K. 腹ペコのムト



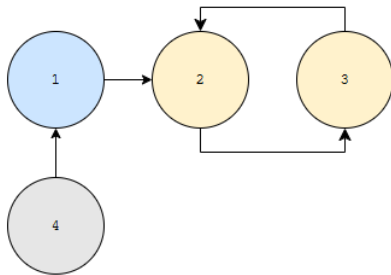
- ✓ 그래프가 위와 같다고 해 봅시다.
- ✓ 1 번 진화한 결과와 5 번 진화한 결과가 같은 재료는 몇 개일까요?
- ✓ $f(x) = f(x + 4)$ 를 만족한다. 함수 $f(x)$ 의 주기가 4 이면 만족하긴 합니다.
- ✓ 그런데 주기가 2 여도, 1 이여도 만족합니다.

K. 腹ペコのムト



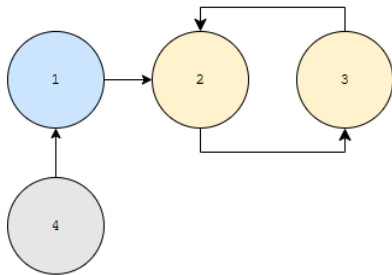
- ✓ 노란색으로 칠한 부분은 사이클이고 사이클의 길이는 2입니다.
- ✓ 주기가 2이다. 라는 걸 캐치했다면 단박에 2개라고 체크하셨을 겁니다.
- ✓ 그런데 1 번 진화한 결과가 함정입니다. 왜?

K. 腹ペコのムト



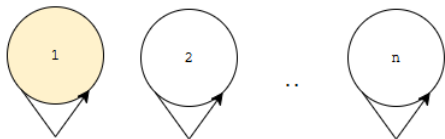
- ✓ 1 번 재료를 봅시다. 1 번 진화한 결과는 2가 됩니다.
- ✓ 5 번 진화한 결과 역시 1 입니다. 같네요? 왜?
- ✓ 1 번 진화했을 때 사이클 요소로 들어가 버리기 때문입니다.

K. 腹ペコのムト



- ✓ 고로 각 컴포넌트를 만날 때 마다 사이클의 길이 c 를 구합니다.
- ✓ 그리고 1 번 진화했을 때 사이클로 들어가는 노드의 수 j 도 구합니다.
- ✓ 여기까지는 쉽습니다.
- ✓ 그리고 이 c 의 배수인 것들에 저 둘의 합을 모두 누적한다는 생각을 하실 수 있는데요.

K. 腹ペコのムト



- ✓ 각 컴포넌트마다 c 의 배수인 것들을 일일이 모두 누적시켜버리면
- ✓ 이 경우에 시간 초과가 나겠습니다.
- ✓ 고로 사이클이 c 인 경우 조건을 만족하는 노드는 몇 개인지를 누적키고
- ✓ 처리가 끝난 후 한꺼번에 누적시켜야 합니다.

K. 腹ペコのムト

- ✓ 여기까지 잘 하셨다면 1 번 진화한 결과와 c 번 진화한 결과가 같다를
- ✓ 사이클이 $c - 1$ 의 약수인 컴포넌트들에서 조건을 만족하는 노드의 수를 구하면 됩니다.
- ✓ $ans[k]$ 가 사이클이 k 의 약수인 컴포넌트들에서 조건을 만족하는 노드의 수라 하면
- ✓ $ans[c - 1]$ 을 구하시면 됩니다.
- ✓ 우리는 이를 전처리 과정에서 했을 뿐입니다.

L. 가희와 음악

dynamic_programming, implementation, prefix_sum

출제진 의도 – **Challenge**

- ✓ 제출 12, 정답 3명 (정답률 25.000%)
- ✓ 처음 푼 사람: **jeoffrey0522**, 96분
- ✓ 출제자: chogahui05

L. 가희와 음악

- ✓ 사실 알고리즘 자체는 익숙한 소재였습니다. 25049 번하고 이 문제가 비슷합니다.
- ✓ 저 문제를 저는 몰랐습니다만, 검수진 분들이 해당 문제를 알려주셨습니다.
- ✓ 이렇게 익숙한 문제 같은 것이 대회에 나왔다면 어떻게 대처해야 할까요?
- ✓ 이 파일에서는 25049의 풀이를 안다는 전제 하에 작성하겠습니다.

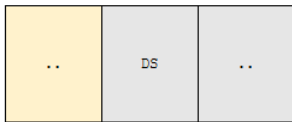
L. 가희와 음악

- ✓ 먼저 25049와의 공통점을 봅시다.
- ✓ 반복해서 들을 수 있는데 최대 3번 이상 반복되는 부분이 없다.
- ✓ 여기까지 공통되는 부분이니까 25049번과 동일 알고리즘을 쓰는 것을 알 수 있습니다.
- ✓ 그러면 차이점은 무엇인가? 반복 구간의 제약이 걸려있다는 것입니다. 어떻게?

L. 가희와 음악

- ✓ 세노와 달세노로 걸리게 됩니다.
- ✓ 이것을 넣을 수 있는 자리에 최대 2개까지 넣을 수 있으니 경우의 수를 잘 따져 봅시다.
- ✓ 세노와 달세노를 넣을 수 있는 자리가 최대 2개라면 가능한 경우의 수는 많아야 4가지.
- ✓ 1개라면 많아야 2가지가 나오게 됩니다.
- ✓ 그렇지 않으면 그냥 누적합으로 처리하면 됩니다.

L. 가희와 음악



- ✓ 먼저 1번 넣은 경우에는 어떤가요?
- ✓ 사실 세노는 아무런 의미 없습니다.
- ✓ 달세뇨를 넣은 경우, 맨 처음으로 돌아가게 되므로 노란 부분이 1번 반복됩니다.
- ✓ 달세뇨까지 누적합에 전체 누적합을 더한 것이 후보해가 되겠네요.

L. 가희와 음악



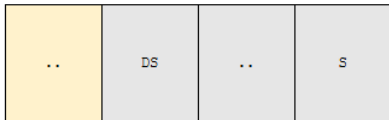
- ✓ 세뇨 세뇨 조합은 아무런 의미가 없습니다.
- ✓ 반복되는 부분도 없으니까 이미 처리한 경우입니다. 제외합니다.

L. 가희와 음악



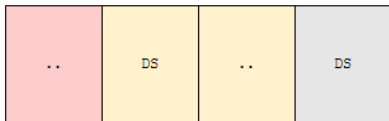
- ✓ 이 경우가 핵심 포인트입니다.
- ✓ 세뇨 뒤에 달세뇨가 붙은 경우 세뇨부터 달세뇨가 있는 위치까지 반복을 할 수 있습니다.

L. 가희와 음악



- ✓ 달세뇨 다음에 세뇨가 오는 경우는 어떨까요?
- ✓ 세뇨는 아무 의미 없습니다. 이 경우는 앞에서 처리한 케이스와 같으니까 처리 안 하면 됩니다.

L. 가희와 음악



- ✓ 달세뇨 - 달세뇨는 불가능합니다.
- ✓ 앞에 세뇨가 없으면 맨 처음으로 돌아가는데 3번 반복되는 부분이 있어요.
- ✓ 그러므로 이 부분은 걸러 줍시다.

L. 가희와 음악



- ✓ 해당 케이스만 가지고 왔다면 이제 문제는 쉬워집니다.
- ✓ 세노 위치를 만났을 때 다음 달세뇨가 가능한 위치 중 누적합이 최대인 지점은 어디인가?
- ✓ $dp_r[i]$ 를 i 이후에 오는 달세뇨 위치 중 누적합이 최대인 위치는 얼마인가로 두면 됩니다.

L. 가희와 음악

- ✓ 남은 것은 파싱인데요. $DSorS$ 와 같은 것은 공백 기준으로 나누면 토큰이 3개여서 어렵습니다.
- ✓ 이걸 어떻게 할까요? or 가 나오면 앞뒤에 $\#$ 을 넣으면 됩니다.
- ✓ 그러면 토큰이 하나가 됩니다. 기호가 들어갈 수 있는 부분이 아니면 수가 옵니다.
- ✓ 따라서 받은 토큰이 기호가 들어갈 수 있는 부분이 아니라면 수로 변환하면 됩니다.

L. 가희와 음악

- ✓ 이미 익숙한 문제의 조건들과 대회에서의 조건을 보고 공통점을 찾습니다.
- ✓ 어떤 조건이 더 걸려있는지, 덜 걸려있는지 찾습니다.
- ✓ 이 문제의 경우 25049 번 문제에 반복할 수 있는 구간 제약이 걸려 있었습니다.
- ✓ 이 부분에 주의해서 문제를 풀어주시면 됩니다.