# 復習済み

1. E

# A Full House 2

正解

# B Calculator

正解

# C Operate 1

正解

# D Diagonal Separation

不正解

解説と同じ解法。最低限の条件を正しく理解することがポイント。

## 正解コード

from collections import defaultdict

N, M = map(int, input().split())

grid = defaultdict(list)

X = set()

for \_ in range(M):

x, y, c = input().split()

x = int(x)

y = int(y)

grid[x].append((y, c))

X.add(x)

X = sorted(list(X))

black\_max\_y = N

for x in X:

grid[x].sort(reverse = True)

flag = False

for y, c in grid[x]:

if c == "W":

if flag:

print("No")

exit()

else:

continue

else:

if flag:

continue

else:

if black\_max\_y < y:

print("No")

exit()

flag = True

if flag == False:

black\_max\_y = min(black\_max\_y, y-1)

print("Yes")

# E Maximize XOR

解いてない

この問題をシンプルに解こうとすると、N個からK個選んで、そのK個の数字を足すという方法。しかしこれだと　　通りそれぞれに対してK回足すことになる。

Kの上限は2\*10^5なのでとなって計算量が多い。

------

※厳密にはがの時、Kが2×105 となることはないので2×1011となることはないが多くの時間がかかる。 <= 10^6を満たすようなNとKの組み合わせは、Nに対して1<=K<=sとN-s <= K <= 2\*10^5まである。N<22の時は全てでKを取れる。

|  |  |  |
| --- | --- | --- |
| N下限 | N上限 | s |
| 1415 | 2\*10^5 | 1 |
| 183 | 1414 | 2 |
| 72 | 182 | 3 |
| 44 | 71 | 4 |
| 33 | 43 | 5 |
| 28 | 32 | 6 |
| 25 | 27 | 7 |
| 24 | 24 | 8 |
| 23 | 23 | 9 |
| 22 | 22 | 10 |
| 22 | 22 | 11 |

この時、例えばN=1400, K = 1398とすると

となり、厳しい。

そこで K > となるときはK個選ぶのではなくN-K個選びそれを合計から引くようにすると選ぶ数が減る。

つまり

とすると計算量は となり、sの上限は表から11なので

11×10^6で計算量を抑えられる。

## 正解コード

from itertools import combinations

N, K = map(int, input().split())

A = list(map(int, input().split()))

ans = 0

if N//2 > K:

for v in combinations(A, K):

tmp = 0

for a in v:

tmp ^= a

if tmp > ans:

ans = tmp

else:

all\_sum = 0

for i in range(N):

all\_sum ^= A[i]

for v in combinations(A, N-K):

tmp = all\_sum

for a in v:

tmp ^= a

if tmp > ans:

ans = tmp

print(ans)

# F Operate K

解いてない

# G Many MST

解いてない