

CS2023 - Aula de Ejercicios N° 8
Brenner H. Ojeda Rios
Semestre 2024-0

Se sugiere que cada estudiante trate de resolver los ejercicios de forma **individual** y luego los discuta en grupo.

Ejercicios

1. (5 pts) Dada una matriz binaria de $m \times n$ que representa un mapa de '1's (tierra) y '0's (agua), devuelve el número de islas.

Una isla está rodeada de agua y se forma conectando tierras adyacentes horizontal o verticalmente. Puedes asumir que los cuatro bordes de la cuadrícula están rodeados de agua. **Sólo será considerada soluciones que usen Disjoint set.**

Ejemplo 1

Input:

```
grid = [
    ["1", "1", "1", "1", "0"],
    ["1", "1", "0", "1", "0"],
    ["1", "1", "0", "0", "0"],
    ["0", "0", "0", "0", "0"]
]
```

Output: 1

Ejemplo 2

Input:

```
grid = [
    ["1", "1", "0", "0", "0"],
    ["1", "1", "0", "0", "0"],
    ["0", "0", "1", "0", "0"],
    ["0", "0", "0", "1", "1"]
]
```

Output: 3

Restricciones:

- $m = \text{grid.length}$
- $n = \text{grid}[i].\text{length}$
- $1 \leq m, n \leq 300$

2. (6 pts) Queremos dividir un grupo de n personas (etiquetadas del 1 al n) en dos grupos de cualquier tamaño. Es posible que a cada persona no le gusten otras personas y no deben pertenecer al mismo grupo.

Dado el número entero n y un arreglo `disgustos` donde `disgustos[i] = [ai, bi]` indica que a la persona etiquetada a_i no le gusta la persona etiquetada b_i , devuelve verdadero si es posible dividir a todos en dos grupos de esta manera. **Sólo será considerada soluciones que usen Disjoint set.**

Ejemplo 1

Input:

`n = 4, dislikes = [[1,2],[1,3],[2,4]]`

Output: `true`

Explicación: El primer grupo tiene [1,4] y el segundo grupo tiene [2,3].

Ejemplo 2

Input:

`n = 3, dislikes = [[1,2],[1,3],[2,3]]`

Output: `false`

Explicación: Necesitamos al menos 3 grupos para dividirlos. No podemos ponerlos en dos grupos.

Restricciones:

- $1 \leq n \leq 2000$
- $0 \leq \text{dislikes.length} \leq 10^4$
- `dislikes[i].length = 2`
- $1 \leq a_i < b_i \leq n$
- Todos los pares de disgustos son únicos.

3. (9 pts) Se le proporciona un arreglo de números enteros `nums` y puede realizar la siguiente operación cualquier cantidad de veces en `nums`:

Intercambie las posiciones de dos elementos `nums[i]` y `nums[j]` si `mcd(nums[i], nums[j]) > 1` donde `mcd(nums[i], nums[j])` es el máximo común divisor de `nums[i]` y `nums[j]`. Devuelve verdadero si es posible ordenar los números en orden no decreciente utilizando el método de intercambio anterior, o falso en caso contrario. **Sólo será considerada soluciones que usen Disjoint set.**

Ejemplo 1

Input: `nums = [7,21,3]`

Output: `true`

Explicación: Podemos ordenar `[7,21,3]` realizando las siguientes operaciones:

- Intercambia 7 y 21 porque `mcd(7,21) = 7`. `nums = [21,7,3]`
- Intercambia 21 y 3 porque `mcd(21,3) = 3`. `nums = [3,7,21]`

Ejemplo 2

Input: `nums = [5,2,6,2]`

Output: `false`

Explicación: Es imposible ordenar el arreglo porque 5 no se puede intercambiar con ningún otro elemento.

Restricciones:

- $1 \leq \text{nums.length} \leq 3 \times 10^4$
- $2 \leq \text{nums}[i] \leq 10^5$