

صورت سوال

یک گراف بدون جهت با n رأس داریم. در ابتدا هیچ یالی در گراف وجود ندارد. سپس q عملیات از یکی از سه نوع زیر داده می‌شود:

- `add u v`: یال (u, v) را به گراف اضافه کن.

- `undo`: آخرین عملیات `add` را که هنوز برگردانده نشده است، برگردان.

- `ask u v`: بگو آیا u و v در یک مؤلفه‌ی همبند هستند یا نه.

فرض کنید هر بار که عملیات `undo` داده می‌شود، حداقل یک عملیات `add` وجود دارد که هنوز `undo` نشده است.

الگوریتمی ارائه دهید که به همه‌ی پرسش‌های `ask` پاسخ دهد و پیچیدگی زمانی کل آن $O(n + q \log n)$ باشد.

راهنمایی‌ها

DSU

$$\text{size}[i] = 1$$

$$\text{parent}[i] = i$$

`find(x)`:

while `parent[x] != x`:

`x = parent[x]`

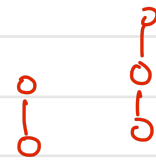
return `x`

`add(u, v)`:

`au = find(u)` ✓

`av = find(v)` ✓

$O(\log n)$



if `au == av`:

`Stack.push(NO_CHANGE)`

return



if `size[au] < size[av]`:

`swap(au, av)`

`Stack.push((av, au, size[au]))`

`parent[av] = au`

`size[au] += size[av]`

undo():

$O(1)$

last = Stack.pop()

if last == NO_CHANGE:

return

(child, root, old_size_root) = last

parent[child] = child

size[root] = old_size_root

ask(u,v):

if find(u) == find(v): return yes

return no

} $O(\log n)$

k: 2^k

→

1 → size ≥ 2

2 → size ≥ 4

⋮

k → size $\geq 2^k$

↳ $k \leq \log n$