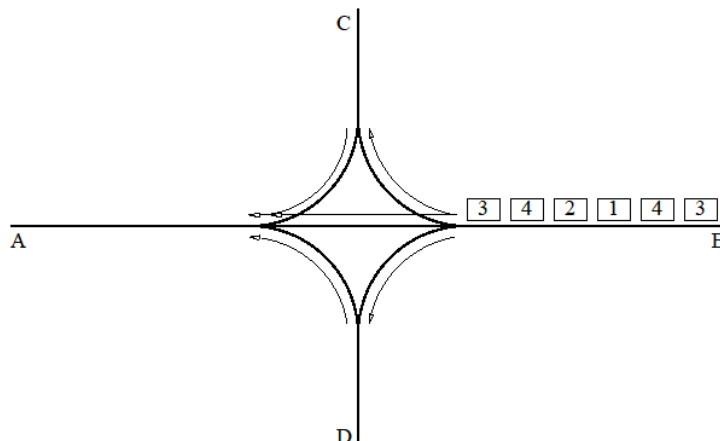


## Duplavermes vasút

Duplaverem város vasútállomásán nagy gondot okoz a szerelvények rendezése. Az állomásról továbbítandó szerelvényeket úgy kell kialakítani, hogy amikor az megérkezik a célállomásra, a szerelvény végéről mindig lekapcsolható legyen az oda továbbított kocsisor. Minden továbbítandó szerelvény négy állomást érint, ezért a rendezés előtt minden kocsit megjelölnek az 1, 2, 3 vagy 4 számokkal.

A szerelvény kocsijait rendezzük át úgy, hogy a szerelvény elején legyenek az 1-essel, aztán a 2-essel, majd a 3-assal, végül a 4-essel megjelöltek! Kezdetben a kocsik az ábrán látható B pályaszakaszon vannak.



A vasúti váltók működése csak a következő műveleteket teszi lehetővé:

- Az átrendezendő kocsisorból balról az első kocsit át lehet mozgatni vagy az **A** szakaszba a már ott lévő kocsik mögé, vagy a **C** vagy a **D** szakaszba a már ott lévő kocsik elé.
- A **C** vagy a **D** szakaszon lévő első kocsit át lehet mozgatni az **A** szakaszon kialakítandó rendezett kocsisor végére.

Készíts programot, amely kiszámítja, hogy adott szerelvények kocsijai átrendezhetőek-e úgy, hogy a kocsik számozásuk alapján monoton növekvő sorozatot alkossanak!

### Bemenet

A standard bemenet első sorában a rendezésre váró szerelvények száma van ( $1 \leq N \leq 10$ ). A következő  $N$  sor mindegyike egy legfeljebb 1000 kocsit tartalmazó rendezendő szerelvényt ír le, minden sort a 0 szám zár, ami már nem része a bemenetnek. Az utolsó 0 kivételével minden szám egy-egy kocsi számát jelöli, azaz 1 és 4 közötti egész számok egy-egy szóközzel elválasztva.

### Kimenet

A standard kimenet pontosan  $N$  sort tartalmazzon, minden szerelvényre meghatározva, hogy kialakítható-e a kívánt kocsisorrend a megfelelő bemeneti kocsisorrendből! Ha kialakítható, az IGEN szó szerepeljen, egyébként a NEM szó!

### Példa

Bemenet

```
2
1 2 3 2 4 0
2 3 1 4 2 1 0
```

Kimenet

```
IGEN
NEM
```

**Korlátok**

Időlimit: 0.1 mp.

Memórialimit: 32 MiB

Pontozás: A tesztek 40%-ában a kocsik száma  $\leq 50$ .