
Control de finestres en un escriptori virtual**V97747_ca**

En un escriptori virtual es poden obrir diverses finestres rectangulars. Cada finestra es descriu amb les coordenades (en píxels) de dues cantonades:

- (x_1, y_1) : cantonada superior esquerra.
- (x_2, y_2) : cantonada inferior dreta.

Es considera que una finestra **cobreix** un punt (x, y) si el punt és dins o a la vora del rectangle, és a dir,

$$x_1 \leq x \leq x_2 \quad \text{i} \quad y_1 \leq y \leq y_2.$$

Als jocs de proves sempre es complirà que $x_1 \leq x_2$ i $y_1 \leq y_2$.

Cal processar una seqüència d'operacions sobre les finestres:

- `ADD id x1 y1 x2 y2`: s'afegeix (obre) una finestra amb identificador `id` i coordenades (x_1, y_1, x_2, y_2) .
- `CLOSE id`: es tanca (elimina) la finestra amb identificador `id`.
- `CHECK x y`: es pregunta si existeix *alguna* finestra oberta que cobreixi el punt (x, y) .
- `TOP x y`: es pregunta quina és la finestra *més al davant* que cobreix el punt (x, y) .

Les finestres afegides més tard queden al davant de les anteriors. Per tant, per a una operació `TOP`, si diverses finestres cobreixen el punt, s'ha de mostrar la que s'ha afegit més recentment i que encara estigui oberta.

Es pot assumir que:

- Un `ADD` sempre fa referència a un identificador que no està obert en aquell moment.
- Un `CLOSE` sempre fa referència a un identificador que està obert en aquell moment.

IMPORTANT: En la solució d'aquest problema has d'usar la tupla `Punt` que està definida de la següent forma:

```
struct Punt {  
    int x, y;  
}
```

Entrada

La primera línia conté un enter N indicant el nombre d'operacions.

A continuació venen N línies, cadascuna amb una operació en un dels formats següents:

- `ADD id x1 y1 x2 y2`
- `CLOSE id`
- `CHECK x y`
- `TOP x y`

Tots els valors són enters.

Sortida

Per cada operació:

- Per a CHECK $x \ y$, s'ha d'escriure YES si almenys una finestra oberta cobreix (x,y) , i NO en cas contrari.
- Per a TOP $x \ y$, s'ha d'escriure l'identificador id de la finestra oberta més al davant que cobreix (x,y) , o NONE si cap finestra cobreix el punt.
- Per a les operacions ADD i CLOSE no s'ha d'escriure res.

Per obtenir més detalls sobre la sortida consulta els jocs de proves públics.

Exemple d'entrada 1

```
20
ADD 1 0 0 10 10
ADD 2 5 5 15 15
ADD 3 2 2 8 8
CHECK 1 1
CHECK 6 6
TOP 1 1
TOP 6 6
TOP 12 12
CHECK 20 20
TOP 20 20
CHECK 3 3
TOP 3 3
CHECK 9 9
TOP 9 9
CHECK 0 1
TOP 0 1
CHECK 15 16
TOP 15 16
CHECK 7 3
TOP 7 3
```

Exemple d'entrada 2

```
60
ADD 1 0 0 4 4
ADD 2 2 2 6 6
ADD 3 5 0 9 3
ADD 4 0 5 3 9
ADD 5 7 7 10 10
ADD 6 1 1 8 8
ADD 7 3 3 3 3
ADD 8 -2 -2 2 2
ADD 9 6 1 6 9
ADD 10 -5 5 -1 8
CHECK 1 1
TOP 1 1
CHECK 3 3
TOP 3 3
TOP 6 2
CHECK -3 6
TOP -3 6
CLOSE 7
TOP 3 3
```

Exemple de sortida 1

```
YES
YES
1
3
2
NO
NONE
YES
3
YES
2
YES
1
NO
NONE
YES
3
```

```
CLOSE 6
TOP 6 2
CHECK 1 1
TOP 1 1
CLOSE 8
TOP 1 1
CHECK 2 2
TOP 2 2
CLOSE 2
TOP 2 2
CHECK 6 6
TOP 6 6
CHECK 8 8
TOP 8 8
CLOSE 9
TOP 6 6
CHECK 6 6
ADD 11 6 6 12 12
TOP 6 6
CHECK 11 11
TOP 8 8
```

```

CLOSE 5
TOP 8 8
CHECK -3 6
CLOSE 10
CHECK -3 6
TOP -3 6
CHECK 5 1
TOP 5 1
CLOSE 3
CHECK 5 1
ADD 12 4 -1 6 2
TOP 5 1
CHECK 4 4
TOP 4 4
CLOSE 1
CHECK 4 4
ADD 13 -10 -10 10 10
TOP 4 4
CHECK 100 100
TOP 100 100

```

Exemple d'entrada 3

```

60
ADD 1 0 0 100 100
ADD 2 10 10 90 90
ADD 3 20 20 80 80
ADD 4 30 30 70 70
ADD 5 40 40 60 60
ADD 6 0 50 100 55
ADD 7 50 0 55 100
ADD 8 25 25 75 75
ADD 9 45 45 65 65
ADD 10 48 48 52 52
TOP 50 50
CHECK 5 5
TOP 5 5
CHECK 35 35
TOP 35 35
CLOSE 10
TOP 50 50

```

Exemple de sortida 2

```

YES
8
YES
7
9
YES
10
6
9
YES
8
1
YES
2
1
YES
9
YES
5
NONE
NO
11
YES
11
11
YES
NO
NONE
YES
3
NO
12
YES
1
NO
13
NO
NONE

```

```

CLOSE 9
TOP 50 50
CLOSE 8
TOP 50 50
CHECK 48 48
TOP 48 48
CLOSE 5
TOP 50 50
CLOSE 4
TOP 50 50
CHECK 50 53
TOP 50 53
CLOSE 6
TOP 50 53
CHECK 53 50
TOP 53 50
CLOSE 7
TOP 53 50
CLOSE 3
TOP 50 50

```

```
CLOSE 2
TOP 50 50
CLOSE 1
CHECK 50 50
TOP 50 50
ADD 1 0 0 100 100
ADD 2 10 10 90 90
ADD 3 20 20 80 80
ADD 4 30 30 70 70
ADD 5 40 40 60 60
ADD 6 0 50 100 55
ADD 7 50 0 55 100
ADD 8 25 25 75 75
ADD 9 45 45 65 65
ADD 10 48 48 52 52
TOP 50 50
TOP 110 0
TOP 200 200
TOP 10 5
TOP 20 5
TOP 30 5
TOP 40 5
TOP 50 5
```

Exemple de sortida 3

```
10
YES
1
YES
8
9
8
7
YES
5
7
7
YES
7
7
YES
7
3
2
1
NO
NONE
10
NONE
NONE
1
1
1
1
7
```

Informació del problema

Autoria: Bernardino Casas

Generació: 2026-01-25T13:14:40.010Z

© *Jutge.org*, 2006–2026.

<https://jutge.org>