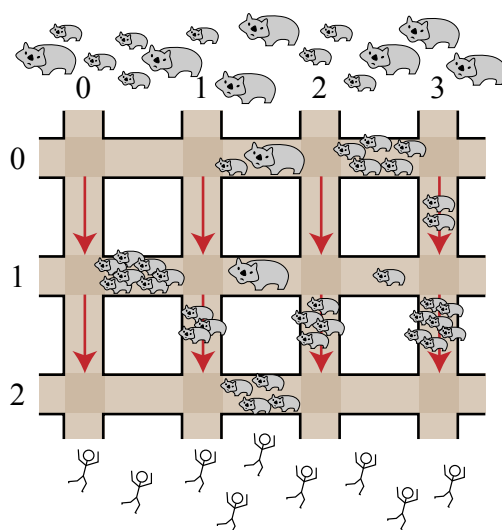


Brisbane-t megtámadták a wombatok, ezért ki kell menteni az embereket.

Brisbane utcái négyzetrácsos elrendezésűek. Van  $R$  vízszintes utca kelet-nyugati irányban  $0, \dots, (R - 1)$  közötti sorszámokkal fentről lefele, és  $C$  függőleges észak déli irányban  $0, \dots, (C - 1)$  közötti sorszámozással, balról jobbra.



A wombatok északról jöttek, ezért az emberek délre menekülnek. Az emberek balra, jobbra és lefelé mehetnek.

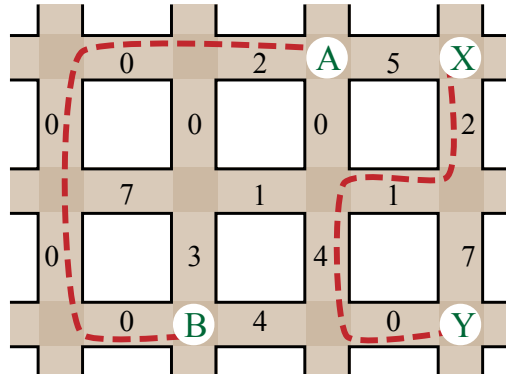
A vízszintes  $P$ . és a függőleges  $Q$ . utca kereszteződését a  $(P, Q)$  pár jelöli. Két kereszteződés közötti útszakaszon wombatok tartózkodhatnak. Ezek száma időnként nőhet vagy csökkenhet. A feladatod egy északi kereszteződésben (azaz a  $0$ . vízszintes sorban) álló embernek megadni, hogy legkevesebb hány wombattal találkozik, ha a megadott déli kereszteződésbe (azaz az  $R - 1$ . sorba), kell eljutnia.

Ismerjük a kezdeti állapotot, azaz a négyzetrács méretét és a wombatok kezdeti számát minden útszakaszon. Ezután a következő típusú események következhetnek:

- *change*, megváltoztatja a wombatok számát adott útszakaszon,
- *escape*, megadja a lehető legkevesebb wombat számot, amivel találkozhat egy a  $0$ . sor adott helyén álló ember az  $R - 1$  sor adott helyére menekülve.

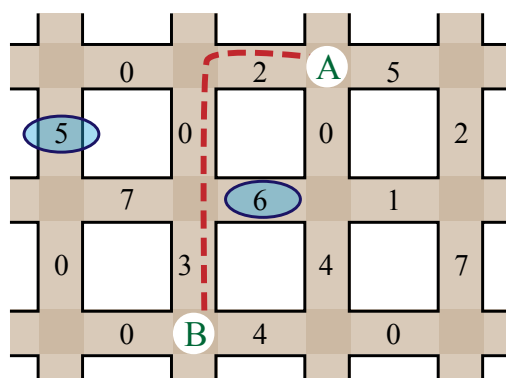
A következő eljárásokat kell megírnod: `init()`, `changeH()`, `changeV()` és `escape()`.

## Példák



Az ábrán  $R = 3$  vízszintes és  $C = 4$  függőleges utca van, jelölve az egyes útszakaszokon levő wombatok számát. Tekintsük a következő eseménysort:

- Az ember az  $A = (0, 2)$  kereszteződésből menekül a  $B = (2, 1)$  keretszteződésbe. A legkisebb wombatszám, amivel találkozhat:  $2$ , a szaggatott vonallal jelzett útvonalon.
- Egy másik ember az  $X = (0, 3)$  kereszteződésből menekül az  $Y = (2, 3)$  kereszteződésbe. A legkisebb wombatszám, amivel találkozhat:  $7$ , a szaggatott vonallal jelzett útvonalon.
- Két change esemény következik: a wombatok száma a 0. oszlop legfelső szakaszán  $0$ -ról  $5$ -re változik. A wombatok száma a vízszintes 1. utca középső szakaszán  $1$ -ről  $6$ -ra változik. A változást a bekarikázott számok jelölik.



- A harmadik ember az  $A = (0, 2)$  kereszteződésből menekül a  $B = (2, 1)$  kereszteződésbe. Most a lehető legkevesebb wombatszám, amivel találkozhat:  $5$ , az új szaggatott vonallal jelzett útvonalon haladva.

---

## Megvalósítás

Az `init()`, `changeH()` and `changeV()` eljárásokat és az `escape()` függvényt tartalmazó fájlt kell beküldened:

## Az eljárásod: `init()`

C/C++ `void init(int R, int C, int H[5000][200], int V[5000][200]);`

Pascal `type wombatsArrayType = array[0..4999, 0..199] of LongInt;  
procedure init(R, C : LongInt; var H, V : wombatsArrayType);`

## Leírás

Az eljárás megkapja a kezdeti állapotot, inicializálhatja a megoldás során használt globális változókat, adatszerkezeteket. Egyszer hívják, a `changeH()`, `changeV()` vagy `escape()` hívása előtt.

### Paraméterek

- `R` : a vízszintes utcák száma.
- `C` : a függőleges utcák száma.
- `H` :  $R \times (C - 1)$  méretű kétdimenziós tömb, ahol `H[P][Q]` a  $(P, Q)$  és a  $(P, Q + 1)$  kereszteződés közötti útszakaszon levő wombatok száma.
- `V` :  $(R - 1) \times C$  méretű kétdimenziós tömb, ahol `V[P][Q]` a  $(P, Q)$  és a  $(P + 1, Q)$  kereszteződés közötti útszakaszon levő wombatok száma

## Az eljárásod: changeH()

C/C++ `void changeH(int P, int Q, int W);`

Pascal `procedure changeH(P, Q, W: LongInt);`

## Leírás

Ezt az eljárást hívják, ha a wombatok száma a  $(P, Q)$  és a  $(P, Q + 1)$  közötti vízszintes útszakaszon változik.

### Paraméterek

- $P$  : a vízszintes utca sorszáma ( $0 \leq P \leq R - 1$ ).
- $Q$  : a baloldali függőleges utca sorszáma ( $0 \leq Q \leq C - 2$ ).
- $W$  : az wombatok új száma ( $0 \leq W \leq 1,000$ ).

## Az eljárásod: changeV()

C/C++ `void changeV(int P, int Q, int W);`

Pascal `procedure changeV(P, Q, W: LongInt);`

## Leírás

Ezt az eljárást hívják, ha a wombatok száma a  $(P, Q)$  és a  $(P + 1, Q)$  közötti függőleges útszakaszon változik.

### Paraméterek

- $P$  : a felső vízszintes utca sorszáma ( $0 \leq P \leq R - 2$ ).
- $Q$  : a függőleges utca sorszáma ( $0 \leq Q \leq C - 1$ ).
- $W$  : az wombatok új száma ( $0 \leq W \leq 1,000$ ).

## A függvényed: escape()

C/C++ `int escape(int V1, int V2);`

Pascal `function escape(V1, V2 : LongInt) : LongInt;`

## Leírás

A legkevesebb wombatszámot adja meg, ha a `(0, V1)` kereszteződésből az `(R-1, V2)` kereszteződésbe kell menekülni.

### Paraméterek

- `V1` : a menekülő ember kezdetben `(0, V1)` pontban van ( $0 \leq V1 \leq C-1$ ).
- `V2` : a menekülés végén az `(R-1, V2)` pontba kell jutnia ( $0 \leq V2 \leq C-1$ ).
- *Visszatérési érték*: a legkisebb wombatszám a menekülési úton.

---

## Mintapélda

Az ábrának megfelelő bemenet:

Function Call	Returns
<code>init(3, 4, [[0,2,5], [7,1,1], [0,4,0]], [[0,0,0,2], [0,3,4,7]])</code>	
<code>escape(2,1)</code>	2
<code>escape(3,3)</code>	7
<code>changeV(0,0,5)</code>	
<code>changeH(1,1,6)</code>	
<code>escape(2,1)</code>	5

---

## Korlátok

- Időlimit: 20 másodperc
- Memória limit: 256 MiB
- $2 \leq R \leq 5,000$
- $1 \leq C \leq 200$
- Legfeljebb 500 változás ( `changeH()` vagy `changeV()` eljárás hívás)
- Legfeljebb 200,000 `escape()` hívás
- Legfeljebb 1,000 wombat bármely útszakaszon

---

## Részfeladatok

Részfeladat	Pontszám	További korlátok
1	9	$C = 1$
2	12	$R, C \leq 20$ , és nincs <code>changeH()</code> vagy <code>changeV()</code> hívás
3	16	$R, C \leq 100$ , és legfeljebb 100 <code>escape()</code> hívás
4	18	$C = 2$
5	21	$C \leq 100$
6	24	(nincs korlát)

---

## Gyakorlás

A mintaértékelő a `wombats.in` fájlból olvassa a bemenetet, ami a következő formájú:



- 1. sor: `R C`
- 2. sor: `H[0][0] ... H[0][C-2]`
- ...
- `(R + 1)` . sor: `H[R-1][0] ... H[R-1][C-2]`
- `(R + 2)` . sor: `V[0][0] ... V[0][C-1]`
- ...
- `(2R)` . sor: `V[R-2][0] ... V[R-2][C-1]`
- következő sor, az események száma: `E`
- következő `E` sor: egy esemény minden sorban, az előfordulás sorrendjében

Ha `C = 1`, akkor nincs vízszintes útszakasz, ezért a (`2 ... R + 1`) sorok kihagyhatók.

Az események leírása a következő formájú:

- `changeH(P, Q, W) : 1 P Q W`
- `changeV(P, Q, W) : 2 P Q W`
- `escape(V1, V2) : 3 V1 V2`

A példa bemenet a fájlban a következő formátumú

```
3 4
0 2 5
7 1 1
0 4 0
0 0 0 2
0 3 4 7
5
3 2 1
3 3 3
2 0 0 5
1 1 1 6
3 2 1
```

---

## Nyelvi előírások

C/C++ You must `#include "wombats.h"` .

Pascal You must define the `unit Wombats` . All arrays are numbered beginning at `0` (not `1` ).

See the solution templates on your machine for examples.