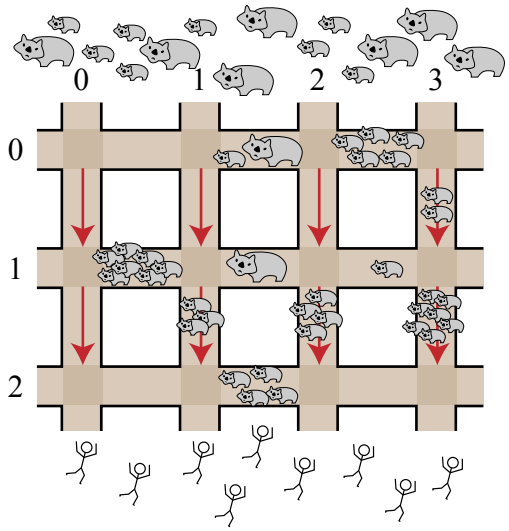
International Olympiad in Informatics 2013
6-13 July 2013
Brisbane, Australia

wombats

Croatian — 1.0

Grad Birsbane su napali mutirani bezrepi kenguri i treba junaka da spasi stanovnike.

Grad možemo prikazati kao pravokutnu mrežu ulica. Postoji `R` vodoravnih istok-zapad ulica, označenih s `0, …, (R - 1)` počevši sa sjevernijima. Postoji `C` vertikalnih sjever-jug ulica označenih s `0, …, (C - 1)` počevši od zapada kao što je prikazano na slici.



Kenguri su napali sa sjevera i stanovnici pokušavaju pobjeći na jug. Ljudi trče vodoravnim ulicama u bilo kojem smjeru, no u vertikalnim ulicama trče *samo u smjeru juga*.

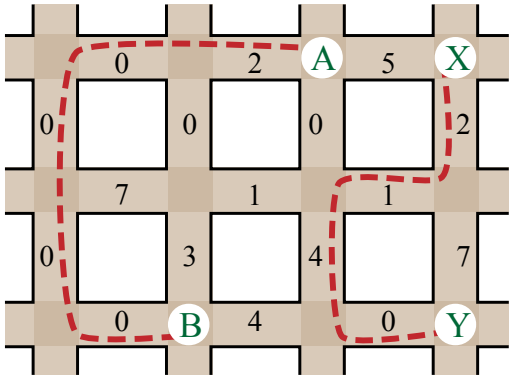
Raskrižje vodoravne ulice `P` i vertikalne ulice `Q` je označeno s `(P, Q)`. Na svakom segmentu ulice između dva raskrižja obitava neki broj kengura i taj se broj može mijenjati u vremenu. Vaš zadatak je voditi svakog stanovnika koji želi pobjeći iz nekog raskrižja na samom sjeveru (na horizontalnoj ulici `0`) do nekog raskrižja na samom jugu (na horizontalnoj ulici `R - 1`) kroz put na kojem u zbroju obitava najmanji mogući broj kengura.

Zadana je veličina mreže ulica te broj kengura na svakom segmentu. Nakon toga zadani su događaji sljedećeg oblika:

- *promjena*, koja mijenja broj kengura koji obitavaju na nekom segmentu; ili
- *bježanje*, gdje neka osoba stiže na raskrižje na samom sjeveru grada i želi pronaći put na jug kojim susreće najmanji broj kengura.

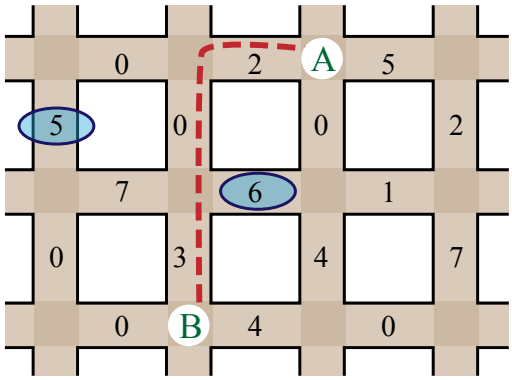
Ove događaje potrebno je obraditi sljedećim funkcijama `init()`, `changeH()`, `changeV()` i `escape()`, kao što je opisano u nastavku.

Primjer



Slika iznad prikazuje početnu mrežu u kojoj postoji `R = 3` vodoravnih ulica i `C = 4` vertikalnih ulica. Broj kengura označen je na svakom segmentu. Promatramo sljedeći niz događaja:

- Osoba dolazi na raskrižje `A = (0,2)` i želi pobjeći na raskrižje `B = (2,1)`. Najmanji broj kengura koji će susresti je `2` ako putuje iscrtkanom linijom.
- Druga osoba dolazi na raskrižje `X = (0,3)` i želi pobjeći na raskrižje `Y = (2,3)`. Najmanji broj kengura koji će susresti je `7` (također označeno iscrtkanom linijom).
- Događaju se dvije promjene. Broj kengura na sjevernom segmentu vertikalne ulice `0` mijenja se u `5`, i broj kengura na srednjem segmentu horizontalne ulice `1` mijenja se u `6`. Pogledati označene brojeve na slici ispod.



- Treća osoba stiže na raskrižje `A = (0, 2)` i želi pobjeći do raskrižja `B = (2, 1)`. Sada je najmanji broj kengura koje će sresti `5` kao što je naznačeno na novoj iscrtkanoj liniji.

Implementacija

Potrebno je priložiti datoteku koja implementira funkcije `init()`, `changeH()` te `changeV()`, `escape()`, kako slijedi:

Vaša funkcija: `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);
```

Opis

Ovom funkcijom dobivate mrežu grada i u njoj možete inicijalizirati globalne varijable i strukture podataka. Biti će pozvana samo jednom, prije bilo kojeg poziva funkcija `changeH()`, `changeV()` or `escape()`.

Parametri

- `R`: Broj horizontalnih ulica.
- `C`: Broj vertikalnih ulica.
- `H`: Polje veličine `R × (C - 1)`, gdje je `H[P][Q]` broj kengura na horizontalnom segmentu između raskrižja `(P, Q)` i `(P, Q + 1)`.
- `V`: Polje veličine `(R - 1) × C`, gdje je `V[P][Q]` broj kengura na vertikalnom segmentu između raskrižja `(P, Q)` i `(P + 1, Q)`.

Vaša funkcija: changeH()

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

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

Opis

Ova funkcija će biti pozvana kada se promijeni broj kengura na horizontalnom segmentu između raskrižja (P, Q) i $(P, Q + 1)$.

Parametri

- `P`: Naznačuje koja horizontalna ulica je promijenjena ($0 \le P \le R - 1$).
- `Q`: Naznačuje između kojih dviju vertikalnih ulica leži mijenjani segment ($0 \le Q \le C - 2$).
- `W`: Novi broj kengura koji obitavaju na naznačenom segmentu ($0 \le W \le 1{,}000$).

Vaša funkcija: `changeV()`

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

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

Opis

Ova funkcija će biti pozvana kada se promijeni broj kengura na vertikalnom segmentu između raskrižja (P, Q) i $(P + 1, Q)$.

Parametri

- `P`: Naznačuje između kojih dviju vertikalnih ulica leži mijenjani segment ($0 \le P \le R - 2$).
- `Q`: Naznačuje koja je vertikalna ulica promijenjena ($0 \le Q \le C - 1$).
- `W`: Novi broj kengura koji obitavaju na naznačenom segmentu ($0 \le W \le 1{,}000$).

Vaša funkcija: `escape()`

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

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

Opis

Ova funkcija mora izračunati najmanji broj kengura koje će osoba susresti putujući s raskrižja $(0, V1)$ do raskrižja $(R-1, V2)$.

Parametri

- `V1`: Naznačuje gdje osoba počinje na horizontalnoj ulici s oznakom `0` ($0 \le V1 \le C-1$).
- `V2`: Naznačuje gdje osoba želi pristići na horizontalnoj ulici s oznakom `R-1` ($0 \le V2 \le C-1$).
- *Povratna vrijednost*: Najmanji broj kengura kraj kojih osoba mora proći.

Test podaci

Primjer sa slike:

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

Ograničenja

- Vremensko ograničenje: 20 sekunda
- Memorijsko ograničenje: 256 MB
- $2 \le R \le 5{,}000$
- $1 \le C \le 200$
- Najviše 500 promjena (odnosno poziva funkcija `changeH()` ili `changeV()`)
- Najviše 200,000 poziva `escape()`
- Najviše 1,000 kengura na nekom segmentu u nekom trenutku.

Bodovanje

Podzadatak	Bodovi	Dodatna ograničenja
1	9	$C = 1$
2	12	$R,C \leq 20$, te neće biti poziva funkcija `changeH()` i `changeV()`
3	16	$R,C \leq 100$, i biti će najviše 100 poziva funkcije `escape()`
4	18	$C = 2$
5	21	$C \leq 100$
6	24	*(n/a)*

Lokalno testiranje

Grejder na vašem računalu čita ulaz iz datoteke `wombats.in`, koja mora biti u sljedećem obliku:

- redak 1: `R C`
- redak 2: `H[0][0] … H[0][C-2]`
- …
- redak $(R + 1)$: `H[R-1][0] … H[R-1][C-2]`
- redak $(R + 2)$: `V[0][0] … V[0][C-1]`
- …
- redak $(2R)$: `V[R-2][0] … V[R-2][C-1]`
- sljedeći redak: `E`
- sljedećih `E` redaka: jedan događaj po liniji u redosljedu kojim se oni događaju

Ako je $C = 1$, prazne linije koje sadrže broj kengura na horizontalnim ulicama (retci `2` do `R + 1`) nisu potrebni.

Redak za svaki događaj mora biti oblika:

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

Npr. primjer iznad bi izgledao ovako:

```
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
```

Napomene

C/C++ Potrebno je dodati `#include "wombats.h"`.

Pascal Potrebno je definirati `unit Wombats`. Svi nizovi biti će indeksirani od `0` (a ne od `1`).

Pogledajte predložak rješenja na vašem računalu.