



International Olympiad in Informatics

2013

6-13 July 2013

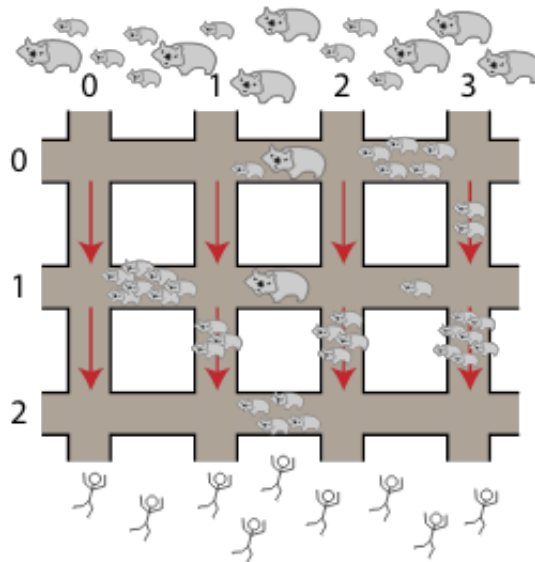
Brisbane, Australia

Вомбатууд

Монгол – 1.1

Брисбэйн хотыг том, мутант вомбатууд эзэлсэн тул та хүмүүсийг хотоос аюулгүйгээр гаргах хэрэгтэй болсон.

Брисбэйний замууд нь том торон дээр байрладаг. R тооны хэвтээ замууд зүүнээс баруун тийш чиглэлтэйгээр тавигдсан байдаг ба тэдгээрийг хойноос урагш тийш нь $0, \dots, (R - 1)$ гэж дугаарласан байдаг бол C тооны босоо замуудыг хойноос урагш чиглэлтэйгээр тавигдсан байдаг бөгөөд тэдгээрийг баруунаас зүүн тийш нь $0, \dots, (C - 1)$ гэж доорх зурагт үзүүлснээр дугаарласан байдаг.



Вомбатууд хойд хэсгийг эзэлсэн тул хүмүүс урд зүг рүү зугтаж байгаа. Хүмүүс хэвтээ зам дээр аль аль тийш гүйж чадах бол босоо зам дээр *зөвхөн урд зүг рүү гүйж* аюулгүй газар очдог.

Хэвтээ зам P болон босоо зам Q -ийн огтлолцлыг (P, Q) гэж тэмдэглэе. Хоёр огтлолцлын хооронд орших замын хэсэгт тодорхой тооны вомбатууд байх ба энэ тоо нь хугацааны явцад өөрчлөгдөж болно. Таны даалгавар бол хойд талын (хэвтээ зам 0 дээрх) өгөгдсөн огтлолцол дээр байгаа хүн болгоныг урд талын (хэвтээ зам $R - 1$ дээрх) өгөгдсөн байрлал руу, боломжит хамгийн цөөхөн вомбатыг дайрч гарах маршрутыг олж хүмүүсийг газарчлах явдал юм.

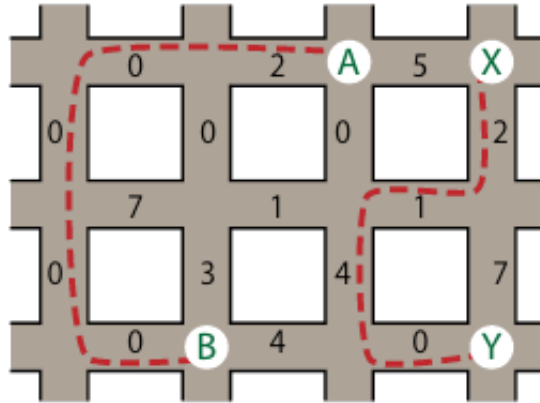
Эхлээд танд торны хэмжээ болон замын хэсэг бүр дээрх вомбатын тоог өгнө. Үүний дараа доорх үзэгдлүүдийн нэг байж болох хэд хэдэн E үйлдлүүдийн дарааллыг өгнө:

- тодорхой нэг замын хэсэг дээр байх вомбатуудын тоог өөрчлөх *change* гэдэг үйлдэл, эсвэл

- хэвтээ зам 0 дээрх өгөгдсөн огтлолцол дээр ирсэн хүнийг хэвтээ зам $R - 1$ дээрх өгөгдсөн огтлолцол руу хүргэх, дайрч гарах вомбатуудын тоо нь хамгийн бага байх маршрутыг олох ёстой болох *escape* гэсэн үйлдэл.

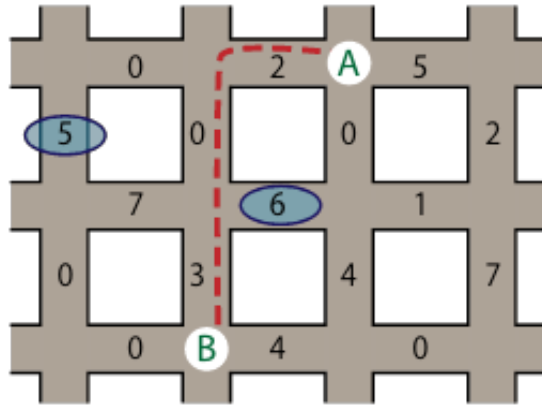
Та эдгээр үйлдлүүдийг хэрэгжүүлэхийн тулд доор тайлбарласан `init()`, `changeH()`, `changeV()` болон `escape()` функцуудыг хэрэгжүүлэх ёстой.

Жишээ



Дээрх зураг дээр $R = 3$ тооны хэвтээ зам болон $C = 4$ тооны босоо замтай, хэсэг бүр дээр байгаа вомбатын тоог нь тэмдэглэсэн анхны газрын зургийг үзүүлжээ. Доорх үйлдлүүдийн дарааллыг авч үзье:

- Нэг хүн $A = (0, 2)$ огтлолцол дээр ирээд $B = (2, 1)$ огтлолцол руу зугтахыг хүсч байгаа. Түүний дайрч гарах вомбатуудын тооны хамгийн бага утга нь 2 ба ийм маршрутыг тасархай шугамаар үзүүлсэн.
- Өөр бас нэг хүн $X = (0, 3)$ огтлолцол дээр ирээд $Y = (2, 3)$ огтлолцол руу зугтахыг хүсч байгаа. Түүний дайрч гарах вомбатуудын тооны хамгийн бага утга нь 7 ба ийм маршрутыг мөн л тасархай шугамаар үзүүлсэн.
- Өөрчлөлтийн хоёр үйлдэл хийгдсэн: босоо зам 0 -ийн хамгийн дээд талын хэсэг дээрх вомбатын тоо 5 болж өөрчлөгдсөн ба хэвтээ зам 1 -ийн дунд хэсэг дээрх вомбатын тоо 6 болж өөрчлөгдсөн. Доорх зураг дээрх дугуйлсан тоонуудыг үзнэ үү.



- Гурав дахь хүн $A = (0, 2)$ огтлолцол дээр ирээд $B = (2, 1)$ огтлолцол руу зугтахыг хүсч байгаа. Одоо түүний дайрч гарах вомбатын тооны хамгийн бага тоо нь 5 байх ба үүнийг шинэ, тасархай шугамаар дүрсэлсэн байгаа.

Хэрэгжүүлэлт

Та доорх байдлаар утга буцаахгүй `init()`, `changeH()` болон `changeV()` функцуудийн болон утга буцаах `escape()` функцийг хэрэгжүүлэлтийг агуулсан файлыг илгээх ёстой:

Таны Функц: `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);`

Тодорхойлолт

Энэ функц нь танд газрын зургийн эхний хэлбэрийг өгөх ба глобал хувьсагч болон өгөгдлийн бүтцэд анхны утга олгоход хэрэглэгдэнэ. Түүнийг ганц л удаа, `changeH()`, `changeV()` эсвэл `escape()` функцуудын алийг нь ч дуудаагүй байхад дуудна.

Параметрууд

- R : Хэвтээ замуудын тоо.
- C : Босоо замуудын тоо.
- H : $R \times (C - 1)$ хэмжээтэй, $H[P][Q]$ элемент нь (P, Q) болон $(P, Q + 1)$ огтлолцлуудын хооронд орших хэвтээ замын хэсэг дээрх вомбатуудын тоог илэрхийлэх хоёр хэмжээст массив.

- V : $(R - 1) \times C$ хэмжээтэй, $V[P][Q]$ элемент нь (P, Q) болон $(P + 1, Q)$ огтлолцлуудын хооронд орших босоо замын хэсэг дээрх вомбатуудын тоог илэрхийлэх хоёр хэмжээст массив.

Таны Функц: `changeH()`

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

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

Тодорхойлолт

Энэ функцийг (P, Q) болон $(P, Q + 1)$ огтлолцлуудын хооронд орших хэвтээ хэсэг дээрх вомбатуудын тоо өөрчлөгдөх үед дуудна.

Параметрууд

- P : Аль хэвтээ замд өөрчлөлт орсныг заана ($0 \leq P \leq R - 1$).
- Q : Уг хэсэг ямар хоёр босоо замын хооронд оршихыг заана ($0 \leq Q \leq C - 2$).
- W : Уг замын хэсэг дээрх вомбатуудын шинэ тоо ($0 \leq W \leq 1,000$).

Таны Функц: `changeV()`

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

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

Тодорхойлолт

Уг функцийг (P, Q) болон $(P + 1, Q)$ огтлолцлуудын хооронд орших босоо замын хэсэг дээрх вомбатуудын тоо өөрчлөгдөх үед дуудна.

Параметрууд

- P : Уг хэсэг ямар хоёр хэвтээ замын хооронд оршиж байгааг заана ($0 \leq P \leq R - 2$).
- Q : Аль босоо замд өөрчлөлт орсныг заана ($0 \leq Q \leq C - 1$).
- W : Уг замын хэсэг дээрх вомбатуудын шинэ тоо ($0 \leq W \leq 1,000$).

Таны Функц: `escape()`

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

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

Тодорхойлолт

Уг функц нь хэн нэгэн хүн $(0, V1)$ огтлолцлоос $(R-1, V2)$ огтлолцол руу зугатах үедээ дайрч өнгөрөх вомбатын тооны боломжит хамгийн бага утгыг тооцоолно.

Параметрууд

- $V1$: Тухайн хүн хэвтээ зам 0-ийн хаанаас эхэлж байгааг заана ($0 \leq V1 \leq C-1$).
- $V2$: Тухайн хүн хэвтээ зам $R-1$ -ийн хаана ирж байгааг заана ($0 \leq V2 \leq C-1$).
- *Буцаах утга*: Тухайн хүний дайрч гарах ёстой вомбатуудын тооны хамгийн бага утга.

Жишээ Өгөгдөл

Доорх өгөгдөл нь дээрх жишээг тодорхойлно:

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

Хязгаарлалтууд

- Хугацааны хязгаарлалт: 20 секунд
- Санах ойн хязгаарлалт: 256 Мб
- $2 \leq R \leq 5,000$
- $1 \leq C \leq 200$
- Дээд тал нь 500 өөрчлөлт хийгдэнэ (`changeH()` болон `changeV()` функцуудийн нийт дуудалт)
- `escape()` функцийг дээд тал нь 200,000 удаа дуудна

- Ямар ч үед аль ч хэсэг дээр дээд тал нь 1,000 вомбат байна

Дэд бодлогууд

Дэд бодлого	Оноо	Оролтын Нэмэлт Хязгаарлалтууд
1	9	$C = 1$
2	12	$R, C \leq 20$, мөн <code>changeH()</code> болон <code>changeV()</code> функцийг огт дуудахгүй
3	16	$R, C \leq 100$, мөн <code>escape()</code> функцийг дээд тал нь 100 удаа дуудна
4	18	$C = 2$
5	21	$C \leq 100$
6	24	(Байхгүй)

Туршилт

Таны компьютер дээрх жишээ шалгагч нь оролтоо доорх форматтай `wombats.in` файлаас уншина:

- 1-р мөр: `R C`
- 2-р мөр: `H[0][0] ... H[0][C-2]`
- ...
- $(R + 1)$ -р мөр: `H[R-1][0] ... H[R-1][C-2]`
- $(R + 2)$ -р мөр: `V[0][0] ... V[0][C-1]`
- ...
- $(2R)$ -р мөр: `V[R-2][0] ... V[R-2][C-1]`
- дараагийн мөр: `E`
- дараагийн `E` тооны мөрүүд: хийгдсэн дарааллаараа, нэг мөрөнд нэг үйлдэл өгөгдөнө

Хэрэв $C = 1$ бол хэвтээ замууд (2 -р мөрөөс $R + 1$ -р мөрүүд дээр байгаа) дээрх вомбатуудын тоог агуулах хоосон мөрнүүд шаардлагагүй.

Үйлдлийг тодорхойлох мөр бүр доорхын аль нэг хэлбэртэй байна:

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

Жишээ нь, дээрх жишээний өгөгдөл нь доор үзүүлсэн форматтай байх хэрэгтэй:

```
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
```

Хэлнүүдийн тайлбар

C/C++ Энэ мөрийг агуулах ёстой `#include "wombats.h"`.

Pascal Та `unit Wombats`-г тодорхойлох ёстой. Бүх массивууд `0`-ээс эхлэх ёстой (`1`-ээс эхлэхгүй).

Жишээ болгож компьютер дээрээ байгаа бодолтын загварыг харна уу.