

Qutunu tap

| Tapşırıq adı | Find the Box |
|---------------|--------------|
| Zaman Limiti | 1 saniyə |
| Yaddaş Limiti | 1 GB |

Maya Lund Universitetində işləyən bir robotika tədqiqatçısıdır. O, universitetin zirzəmisində dəyərli bir xəzinən olduğunu öyrəndi. Xəzinə yerin altındakı otaqların birindəki qututadır. Təəssüf ki, Maya rahatlıqla gedib qutunu axtara bilmir. Ora çox qaranlıqdır, və fənər ilə ora getmək şübhə yaradar. Onun xəzinəni tapması üçün yeganə yol zirzəmidə yerləşən robot tozсорanı uzaqdan idarə etməkdir.

Zirzəmi $H \times W$ ölçülü qrid formasındadır. Burada sətirlər 0'dan $H - 1$ 'ə (yuxarıdan aşağıya) və sütunlar 0'dan $W - 1$ 'ə (soldan sağa) nömrələnib, yəni sol üst xananın koordinatı $(0, 0)$ və sağ aşağı xananın koordinatı $(H - 1, W - 1)$ 'dir. Xəzinənin olduğu qutu $(0, 0)$ 'dan başqa hansısa bilinməyən bir xanadadır. Hər gecə robot sol üst xanada başlayır və zirzəmidə hərəkət etməyə başlayır.

Hər gecə Maya robotu necə hərəkət etməsi ilə bağlı sorğu verir. Bu sorğu " $<$ ", " $>$ ", " \wedge " və " \vee " işarələrindən ibarət string'dir. Əgər robot hər tərəfi boş olan (r, c) xanasında dayanıbsa, " $<$ " təlimatı robotu sola doğru $(r, c - 1)$ xanasına, " $>$ " təlimatı robotu sağa doğru $(r, c + 1)$ xanasına, " \wedge " təlimatı robotu yuxarıya doğru $(r - 1, c)$ xanasına, və " \vee " təlimatı robotu aşağıya doğru $(r + 1, c)$ xanasına hərəkət etdirir.

Zirzəminin divarları möhkəmdir, yəni əgər robot qridən çölə çıxmağa çalışsa, heç bir şey baş verməyəcək. Həmçinin, qutu da möhkəmdir, yəni onu itələmək olmur. Hər gecənin sonunda robot harada olduğunu bildirərək yenidən sol üst xanaya qayıdacaq.

Zaman azalır, buna görə də Maya qutunu mümkün qədər az gecəyə tapmaq istəyir.

İnteraksiya

Bu, interaktiv tapşırıqdır.

- Proqramınız başladığında bir sətirdə iki tam ədəd olan H və W ədədlərini: qridin hündürlük və genişliyini oxumalıdır.

- Daha sonra sizin proqramınız qreyder ilə interaksiya etməlidir. İnteraksiyanın hər raundunda “?” işarəsi, daha sonra boş olmayan “<”, “>”, “^”, “v” işarələrindən ibarət s string'i çıxışa verməlisiniz. Bu string'in uzunluğu ən çox 20 000 ola bilər. Daha sonra proqramınız robotun təlimatı yerinə yetirdikdən sonrakı koordinatını göstərən iki tam ədəd oxumalıdır r, c ($0 \leq r \leq H - 1, 0 \leq c \leq W - 1$). Qeyd edək ki, robot hər sorğudan sonra $(0, 0)$ xanasına geri qaydır.
- Qutunun yerini bildikdən sonra “!”, daha sonra qutunun sətir və sütununu göstərən iki tam ədəd r_b, c_b çıxışa verin ($0 \leq r_b \leq H - 1, 0 \leq c_b \leq W - 1$). Bundan sonra sizin proqramınız başqa sorğu vermədən dayanmalıdır. Çıxışa verilən sonuncu sətir sizin balınız hesablanan zaman sorğu olaraq nəzərə alınmayacaq.

Sorğu soruşduqdan sonra standart çıxışı flush etməyi unutmayın, əks halda sizin həlliniz Time Limit Exceeded nəticəsi ala bilər. Python dilində `print()` avtomatik flush edir. C++ dilində, `cout << endl`; çıxışa təzə sətir verməkdən əlavə həmçinin flush edir; əgər `printf` istifadə etsəniz `fflush(stdout)` sətirini də yazın.

Qreyder adaptiv deyil, yəni qutunun yeri interaksiya başlamazdan əvvəl təyin olunur.

Məhdudiyyətlər və Qiymətləndirmə

- $1 \leq H, W \leq 50$.
- Qutu heç vaxt $(0, 0)$ xanasında olmayacaq. Yəni $H + W \geq 3$.
- Hər sorğu maksimum 20 000 təlimatdan ibarət ola bilər.
- Ən çox 2 500 sorğu soruşa bilərsiniz. (Cavabı çıxışa vermək əlavə sorğu sayılmır)

Sizin həlliniz bir neçə test üzərində yoxlanılacaq. Əgər sizin həlliniz *hansısa* testi keçə bilməsə (Məsələn pozisiyanı səhv tapsanız (Wrong Answer), icra xətası baş versə (Runtime Error), zaman limitini aşsanız (Time Limit Exceeded), və s.), bu zaman siz 0 bal və uyğun nəticəni alacaqsınız.

Əgər proqramınız pozisiyanı *bütün* testlərdə düzgün tapsa, Accepted nəticəsi alacaqsınız və balınız aşağıdakı kimi hesablanacaq:

$$\text{score} = \min \left(\frac{100\sqrt{2}}{\sqrt{Q}}, 100 \right) \text{ bal,}$$

Burada Q bütün testləri nəzərə aldıqda soruşduğunuz maksimum sorğu sayıdır. Cavabı çıxışa vermək sorğu sayılmır. Bal ən yaxın tam ədədə yuvarlaşdırılacaq.

100 bal almaq üçün sizin proqramınız hər testi ən çox $Q = 2$ sayda sorğuya həll etməlidir. Aşağıdakı cədvəl bəzi Q dəyərləri üçün müvafiq balları göstərir.

| | | | | | | | | | | |
|-----|-----|----|----|----|-----|----|-----|----|-----|------|
| Q | 2 | 3 | 4 | 5 | ... | 20 | ... | 50 | ... | 2500 |
| Bal | 100 | 82 | 71 | 63 | ... | 32 | ... | 20 | ... | 3 |

Yoxlama aləti

Həllinizi yoxlamağı asanlaşdırmaq üçün yükləyə biləcəyiniz sadə bir alət verilir. Kattis'də məsələnin olduğu səhifənin aşağısındakı "attachments" hissəsinə baxın. Aləti dəyişə və ya istifadə etməyə də bilərsiniz. Qeyd edək ki rəsmi qreyder sizə verilən yoxlama alətindən fərqli işləyir.

Nümunə istifadə ($H = 4$, $W = 5$, və qutu $r = 2$, $c = 3$ xanasındadır):

Python proqramları üçün, məsələn faylın adı `solution.py` olsun (normalda `pypy3 solution.py` sətiri ilə işə salınır):

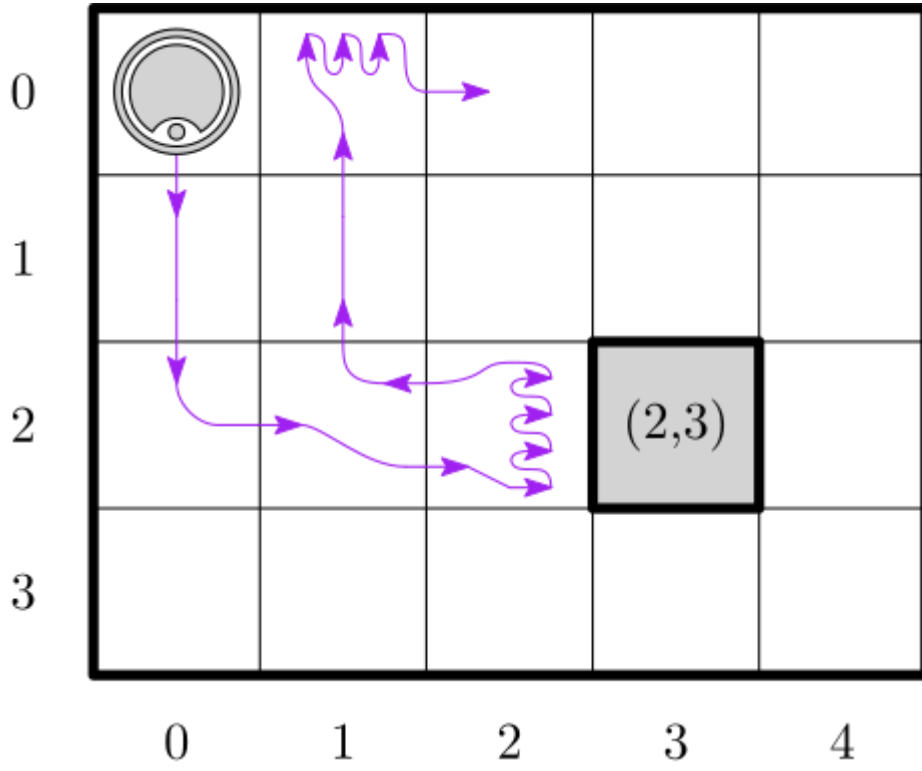
```
python3 testing_tool.py pypy3 solution.py <<<"4 5 2 3"
```

C++ proqramları üçün, əvvəlcə kompayl edin (məsələn `g++ -g -O2 -std=gnu++17 -static solution.cpp -o solution.out`) daha sonra işə salın:

```
python3 testing_tool.py ./solution.out <<<"4 5 2 3"
```

Nümunə

Nümunə testə nəzər yetirin. Qridin hündürlüyü $H = 4$ eni $W = 5$ 'dir, və qutu $(r, c) = (2, 3)$ xanasındadır. Aşağıdakı şəkil robotun birinci sorğu olan "`? vv>>>>>><^^^^^>`" sətirindən sonra izlədiyi yolu göstərir. Sonda robot $(r, c) = (0, 2)$ xanasında dayanır. İkinci sorğudan əvvəl robot yenidən $(0, 0)$ xanasına geri qayıdır. Daha sonra həll yeni sorğu soruşur: "`?>>>>>>>vvvvvvvvvv`". Bu sorğudan sonra robot sağ aşağı olan $(r, c) = (3, 4)$ xanasında dayanır. Artıq həll cavabı təxmin eləməyə çalışaraq "`! 2 3`" çıxışa verir, hansı ki düzgün cavabdır.



| qreyderin çıxış verilənləri | sizin çıxış verilənləriniz |
|-----------------------------|----------------------------|
| 4 5 | |
| | ? w>>>>>>>><^>>>>>> |
| 0 2 | |
| | ? >>>>>>>>>w>>>>>>>> |
| 3 4 | |
| | ! 2 3 |