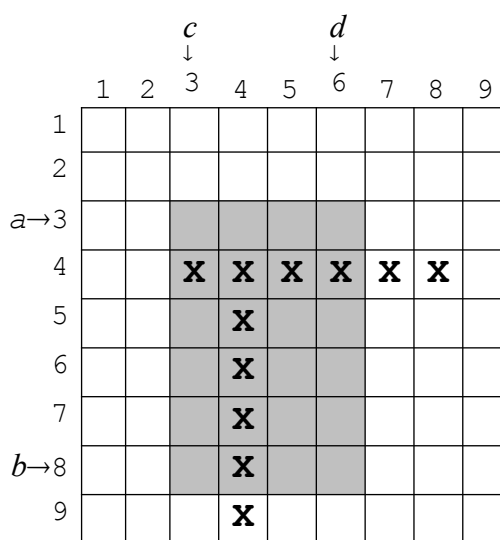


Две пръчки

ЗАДАЧА

Пръчка е хоризонтална или вертикална редица от поне 2 последователни клетки на решетка. Две пръчки – една хоризонтална и една вертикална – са поставени в решетка с N реда и N стълба. На Фигура 1 двете пръчки са означени със знаци **X**. Пръчките могат да са с еднаква или различна дължина, както и да имат обща клетка. Ако някоя клетка може да се разглежда както като принадлежаща, така и като не принадлежаща на някоя пръчка, както е с клетката (4,4) на Фигура 1, то тогава се приема, че клетката принадлежи и на двете пръчки. Следователно най-горна клетка на вертикалната пръчка на Фигура 1 е клетката (4,4), а не клетката (5,4).



В началото разположението на двете пръчки е неизвестно. Задачата е да се напише програма, която определя това разположение. Да наречем хоризонталната пръчка ROD1, а вертикалната – ROD2. Всяка клетка на решетката се представя с двойката (r, c) , където r е редът, а c – стълбът, в който се намира, като клетката в горния ляв ъгъл на решетката е (1,1). Всяка пръчка се представя чрез крайните си две клетки $\langle (r_1, c_1), (r_2, c_2) \rangle$. На Фигура 1 пръчката ROD1 се представя с $\langle (4,3), (4,8) \rangle$, а ROD2 – с $\langle (4,4), (9,4) \rangle$.

За решаване на задачата трябва да се използват библиотечни функции за въвеждане на данните, за намиране на решението и за извеждане на получения резултат. Размерът на решетката се получава като резултат от работата на библиотечната функция `gridsize`, която трябва да се извика в началото на обработката на всеки тестов пример. За определяне на разположението на пръчките, може да използвате само библиотечната функция `rect(a, b, c, d)`, която проверява правоъгълната област $[a, b] \times [c, d]$ (потъмнена на Фигура 1), където $a < b$ и $c < d$. Ако в някоя от клетките на правоъгълника $[a, b] \times [c, d]$ се намира част от пръчка, функцията `rect` връща 1. В противен случай – връща 0. За примерът от Фигура 1 извикването `rect(3, 8, 3, 6)` ще върне 1. Напишете програма, която да определя разположението на двете пръчки, като използва ограничен брой извиквания на функцията `rect`.

За да изведете получения резултат използвайте библиотечната функция `report(r1, c1, r2, c2, p1, q1, p2, q2)`, където $\langle (r_1, c_1), (r_2, c_2) \rangle$ е пръчката ROD1, а $\langle (p_1, q_1), (p_2, q_2) \rangle$ е ROD2. Извикването на `report` прекратява изпълнението на програмата. Напомняме, че ROD1 е хоризонталната, а ROD2 – вертикалната пръчка и че (r_1, c_1) е левият край на ROD1, а (p_1, q_1) – горният край на ROD2. Следователно $r_1=r_2$, $c_1 < c_2$, $p_1=p_2$, $q_1 < q_2$. Ако параметрите на `report` не отговарят на тези изисквания, ще получите съобщение за грешка на стандартния изход.

ОГРАНИЧЕНИЯ

- Вход може да получавате само с извикване на библиотечните функции `gridsize` и `rect`.
- За размера на решетката N е в сила $5 \leq N \leq 10000$.
- Броят на извикванията на функцията `rect` трябва да бъде най-много 400. Ако програмата извика `rect` за 401-ви път – нейното изпълнение се преустановява.
- Програмата трябва да извика повече от един път функцията `rect` и точно един път функцията `report`.
- Ако програмата извика неправилно `rect` (т.е. заявеният правоъгълник излиза извън границите на решетката) нейното изпълнение се преустановява.
- Програмата не трябва да чете и/или пише никакви файлове и не трябва да използва стандартния вход и/или изход.

БИБЛИОТЕКИ

Дадени са следните библиотеки:

Библиотека за FreePascal ([`prectlib.ppu`](#), [`prectlib.o`](#))

```
function gridsize: LongInt;  
function rect(a,b,c,d : LongInt) : LongInt;  
procedure report(r1, c1, r2, c2, p1, q1, p2, q2 :  
LongInt);
```

Инструкция: За да компилирате вашата `rods.pas`, включете оператора `uses prectlib;`

в изходния текст и компилирайте с
`fpc -So -O2 -XS rods.pas`

Програмата `prodstool.pas` е пример за използване на тази библиотека.

Библиотека за GNU C/C++ ([`crectlib.h`](#), [`crectlib.o`](#))

```
int gridsize();  
int rect(int a, int b, int c, int d);  
void report(int r1, int c1, int r2, int c2, int p1, int  
q1,
```

```
int p2, int q2);
```

Instructions: За да компилирате вашата `rods.c`, използвайте
`#include "crectlib.h"`

в изходния текст и компилирайте с:

```
gcc -O2 -static rods.c crectlib.o -lm  
g++ -O2 -static rods.cpp crectlib.o -lm
```

Програмата `crodstool.c` е пример за използване на тази библиотека.

За пишещите на C/C++ в RHIDE

Непременно задайте на Option->Linker configuration стойност `crectlib.o`.

ЕКСПЕРИМЕНТИ

За да правите експерименти с библиотеката, трябва да създадете файл `rods.in`. Файлът трябва да се състои от три реда. Първият ред да съдържа размера на решетката N . Вторият ред да съдържа координатите $r_1 c_1 r_2 c_2$ на ROD1, където (r_1, c_1) е левият край на ROD1. Третият ред да съдържа координатите $p_1 q_1 p_2 q_2$ на ROD2, където (p_1, q_1) е горният край на ROD2.

След изпълнението на вашата програма, която извиква `report`, ще получите изходния файл `rods.out`. Той съдържа броя на извикванията на функцията `rect`, както и координатите на пръчките, които програмата е извела с `report`. Ако програмата е допуснала грешки или отклонения от изискванията, файлът ще съдържа съответно съобщение за грешка.

Диалогът между вашата програма и библиотеката се регистрира във файла `rods.log`. Той съдържа последователността от извиквания на функции, извършени от програмата във вида " $k : rect(a, b, c, d) = ans$ ", което означава, че k -тото извикване на `rect` с параметри a, b, c, d връща ans .

ПРИМЕРЕН ВХОД И ИЗХОД

Пример:

`rods.in`

9
4 3 4 8
4 4 9 4

`rods.out`

20
4 3 4 8
4 4 9 4

ОЦЕНЯВАНЕ

Ако програмата наруши някое от ограничения (например извика повече от 400 пъти `rect`) или резултатът (местоположението на пръчките) не е вярно, ще получите 0 точки.

Ако резултатът е правилен, броят на точките, които ще получите зависи от броя на извикванията на функцията `rect`. Ако при работа над един тестов пример



броят на извикванията не надвишава 100, ще получите 5 точки. Ако извикванията са от 101 до 200 – 3 точки, а ако извикванията са от 201 до 400 – само 1 точка.