



ТРЕНИРОВЪЧНО СЪСТЕЗАНИЕ НА РАЗШИРЕНИЯ НАЦИОНАЛЕН ОТБОР

9 юни 2024 г.

Група G, 9 – 12 клас

Задача GT2. СЕДЛОВА ТОЧКА

🕒 0,1 сек. 📄 256 MB

Позабравената ни героиня Дени се забърка в ново приключение. Тя се изгуби в лабиринт с формата на таблица, пълна с не какво и да е - ами с числа. Таблицата е с размери $N \times M$ и числата в нея са цели положителни. Дени е умно момиче и затова тя е разбрала, че изходите от таблицата са в т.нар. *локални седлови* точки. *Локална седлова* точка на таблицата е такава клетка, която има стойност строго по-малка от тази на своите съседи. Две клетки са съседни, ако имат обща страна.

Дени може да проучва каква е стойността в произволни клетки от таблицата и иска с възможно най-малък брой проучвания да намери изход от таблицата.

Задача

Напишете програма **saddle**, която да намира *локална седлова* точка на неизвестна таблица. Тя трябва да съдържа функцията `find_saddle_point`, която ще се компилира с програмата на журито.

Детайли по имплементацията

Функцията `find_saddle_point` трябва да е в следния формат:

```
pair <int, int> find_saddle_point (int N, int M);
```

Тя се вика веднъж от програмата на журито с два параметъра - съответно броя редове и броя колони на таблицата. Функцията трябва да върне наредена двойка с номера на реда и номера на колоната на *локална седлова* точка. Редовете са номерирани последователно с числата от 1 до N , а колоните - последователно с числата от 1 до M .

Функцията, чрез която може да правите проучване на стойността на клетка има следния формат:

```
long long int value (int r, int c);
```

Параметрите r и c задават реда и колоната на клетката, която проучвате, а върнатата стойност е стойността на съответната клетка. Ако параметрите не задават валидна клетка на таблицата, то ще получите `Wrong answer` за съответния тест. Сложността на функцията е константна. Може да викате тази функция най-много $N \cdot M$ пъти, иначе също ще получите `Wrong answer`.

Вашата програма **saddle** трябва да имплементира функцията `find_saddle_point`. Тя може да съдържа и друг код, функции и глобални променливи, необходими за работата Ви, но не трябва да съдържа главната функция `main`. Също така, не трябва да четете от стандартния вход или да отпечатвате на стандартния изход. Програмата Ви трябва да включва хедър файла `saddle.h` чрез указание към препроцесора:

```
#include "saddle.h"
```

Ограничения

- ♣ $1 \leq N, M \leq 500$.
- ♣ Стойностите в таблицата са цели числа в интервала $[1; 10^{18}]$.
- ♣ Гарантирано е, че има поне една *локална седлова* точка в таблицата.
- ♣ Няма съседни клетки в таблицата с равни стойности.



ТРЕНИРОВЪЧНО СЪСТЕЗАНИЕ НА РАЗШИРЕНИЯ НАЦИОНАЛЕН ОТБОР

9 юни 2024 г.

Група G, 9 – 12 клас

Подзадачи

Подзадача	Точки	Необходими подзадачи	N	Други ограничения
1	0	–	–	Таблицата от примерната комуникация.
2	21	–	$= 1$	–
3	79	1 – 2	≤ 500	Има тестове, за които програмата на журито е адаптивна - при тях няма предварително фиксирана таблица, а тя се получава динамично спрямо извършените проучвания.

Точките за дадена подзадача се получават само ако се преминат **успешно** всички тестове, предвидени за нея и необходимите подзадачи, и са равни на минималната оценка на тест в нея, умножена по точките на подзадачата.

Оценяване

Всеки тест получава оценка, която е дробно число между 0 и 1 включително. Ако тест има положителна оценка, то той се счита за **успешен** за вашето решение. Един тест има положителна оценка, ако правилно намерите *локална седлова* точка в таблицата в рамките на ограниченията по памет и време. Ако означим с cnt броя извиквания на функцията `value` за даден тест, то оценката на теста се смята по следния начин:

- Подзадача 2.
 - Ако $cnt < 30$, то оценката е равна на 1.
 - Ако $30 \leq cnt < 300$, то оценката е равна на $\min(\frac{30}{cnt}, 1)$.
 - Ако $cnt \geq 300$, то оценката е равна на 0.1.
- Подзадача 3.
 - Ако $cnt < 2024$, то оценката е равна на 1.
 - Ако $2024 \leq cnt < 5000$, то оценката е равна на $\min((\frac{2024}{cnt})^{\max(\frac{4000}{cnt}, 1)}, 1)$.
 - Ако $5000 \leq cnt < 10000$, то оценката е равна на $\min((\frac{2024}{cnt})^{1.2}, 0.4)$.
 - Ако $cnt \geq 10000$, то оценката е равна на 0.1.

Примерна комуникация

Нека имаме следната таблица с 3 реда и 4 колони:

5	3	5	6
10	6	4	10
2	9	8	2

Действия на вашата програма	Действия и отговори на журито
	<code>find_saddle_point(3, 4)</code>
<code>value(1, 2)</code>	<code>return 3</code>
<code>value(1, 1)</code>	<code>return 5</code>
<code>value(1, 3)</code>	<code>return 5</code>
<code>value(2, 2)</code>	<code>return 6</code>
<code>return {1, 2}</code>	



ТРЕНИРОВЪЧНО СЪСТЕЗАНИЕ НА РАЗШИРЕНИЯ НАЦИОНАЛЕН ОТБОР

9 юни 2024 г.

Група G, 9 – 12 клас

Локално тестване

За локално тестване са предоставени файловете `saddle.h`, `Lgrader.cpp` и примерен файл `saddle.cpp`, в който може да напишете вашата програма. Когато предоставените файлове са в една папка, може да компилирате заедно вашата програма `saddle.cpp` и `Lgrader.cpp`. Така ще получите програма, с която можете да проверите верността на функцията `Vi`.

Програмата ще изисква от стандартния вход следната последователност от данни:

- на първия ред: две цели положителни числа – броят редове N и броят колони M ;
- на следващите N реда по M цели положителни числа, които описват последователно стойностите в таблицата от първия ред и първата колона до последния ред и последната колона.

Ако не спазвате протокола за комуникация или не намерите *локална седлова точка*, то ще получите подходящо съобщение за грешка. Иначе, при успешно изпълнение на програмата, ще получите съобщение "Correctly found local saddle point."