



Боење според броеви

Боење според броеви е позната загатка. Тука ќе разгледаме едноставна еднодимензионална верзија на оваа загатка. Во оваа верзија на загатката, даден е еден ред од n квадратчиња. Квадратчињата се нумерирани од лево кон десно со целите броеви од 0 до $n - 1$. За да се реши загатката, секое квадратче треба да се обои црно или бело. Со знакот 'X' ќе ги означуваме квадратчињата кои се обоени во црно, а со '_' квадратчињата обоени во бело.

Во оваа загатка дадена е низа $c = [c_0, \dots, c_{k-1}]$ од k позитивни цели броеви чии елементи се наречени *ключеви*. За да се реши загатката, треба да се обојат квадратчињата така што црните квадратчиња во редот формираат точно k блокови од последователни квадратчиња. Всушност, бројот на црни квадратчиња во i -тиот блок (првиот блок има индекс 0) од лево кон десно е еднаков на c_i . На пример, ако клучевите се $c = [3, 4]$, решената загатка има точно два блока од последователни црни квадратчиња: еден блок со должина 3 и уште еден блок со должина 4. Оттука, ако $n = 10$ и $c = [3, 4]$, едно решение кое може да се добие со дадените клучеви е "XXX XXXX". Да забележиме дека "XXXX XXX" не ги исполнува барањата дадени од клучевите затоа што блоковите од црни квадратчиња не се во точниот редослед. Исто така, "XXXXXXXX" не ги исполнува барањата дадени од клучевите затоа што во ова обојување има само еден блок од црни квадратчиња, наместо да има два одделни блока.

Дадена е делумно решена загатка Боење според броеви. Поточно, дадени се n и c , и дополнително за некои квадратчиња е дадено дека мора да се црни, а за некои квадратчиња дека мора да се бели. Ваша задача е да најдете други информации за загатката, како што е опишано подолу.

Нека *валидно решение* е она решение кое ги исполнува барањата дадени од клучевите и е во согласност со дадените информации за обојувањето на познатите квадратчиња. Вие треба да напишете програма во која ќе се најдат квадратчињата кои се обоени црно во сите валидни решенија и квадратчињата кои се обоени бело во сите валидни решенија.

Може да претпоставите дека влезот е таков да секогаш ќе постои барем едно валидно решение.

Детали за имплементација

Треба да ја имплементирате следната функција (метод):

- `string solve_puzzle(string s, int[] c)`
 - s : стринг со должина n . За секое i , $0 \leq i \leq n - 1$, знакот i е:
 - 'X', ако квадратчето i мора да е црно,

- ' _ ', ако квадратчето i мора да е бело,
- '. ', ако нема информација за квадратчето i .
- c : низа со должина k која содржи клучеви, како што е дефинирано погоре,
- функцијата треба да врати стринг со должина n . За секое i , $0 \leq i \leq n - 1$, знакот i од излезот треба да биде:
 - 'X', ако квадратчето i е црно во секое валидно решение,
 - ' _ ', ако квадратчето i е бело во секое валидно решение,
 - '?', инаку (т.е., ако постојат две валидни решенија така што квадратчето i е црно во едното, а бело во другото решение).

Во програмскиот јазик C потписот на функцијата е малку различен:

- `void solve_puzzle(int n, char* s, int k, int* c, char* result)`
- n : должина на стрингот s (бројот на квадратчиња),
- k : должина на низата c (бројот на клучеви),
- другите параметри се исти како погоре,
- функцијата треба да го зачува стрингот од n знаци во резултатот `result`.

ASCII кодовите на знаците кои се користат во оваа задача се:

- 'X': 88,
- ' _ ': 95,
- '. ': 46,
- '?': 63.

Ве молиме да ги користите дадените темплејт датотеки за детали околу имплементацијата во вашиот програмски јазик.

Примери

Пример 1

```
solve_puzzle(".....", [3, 4])
```

Сите можни валидни решенија на оваа загатка се:

- "XXX_XXXX_ ",
- "XXX__XXXX_ ",
- "XXX___XXXX",
- "_XXX_XXXX_ ",
- "_XXX__XXXX",
- "__XXX_XXXX".

Може да се забележи дека квадратчињата со индекси 2, 6 и 7 (индексот на првото квадратче е 0) се црни во сите валидни решенија. Сите други квадратчиња можат, но не мора да бидат црни. Оттука, точниот одговор е "??X???XX??".

Пример 2

```
solve_puzzle(".....", [3, 4])
```

Во овој пример има само едно валидно решение и точниот одговор е “XXX_XXXX”.

Пример 3

```
solve_puzzle("..._. . . . .", [3])
```

Во овој пример може да се заклучи дека квадратчето 4 мора да биде бело — не може да се формираат 3 последователни црни квадратчиња меѓу белите квадратчиња со индекси 3 и 5. Оттука, точниот одговор е “???_???”.

Пример 4

```
solve_puzzle(".X. . . . .", [3])
```

Има само две валидни решенија кои ги задоволуваат дадените барања:

- “XXX_____”,
- “_XXX_____”.

Оттука, точниот одговор е “?XX?_____”.

Подзадачи

Во сите подзадачи $1 \leq k \leq n$, и $1 \leq c_i \leq n$ за секое i , $0 \leq i \leq k - 1$.

1. (7 поени) $n \leq 20$, $k = 1$, s содржи само ‘.’ (празна загатка),
2. (3 поени) $n \leq 20$, s содржи само ‘.’,
3. (22 поени) $n \leq 100$, s содржи само ‘.’,
4. (27 поени) $n \leq 100$, s содржи само ‘.’ и ‘_’ (информација само за бели квадратчиња),
5. (21 поени) $n \leq 100$,
6. (10 поени) $n \leq 5\,000$, $k \leq 100$,
7. (10 поени) $n \leq 200\,000$, $k \leq 100$.

Оценувач

Оценувачот ги чита влезните податоци во следниот формат:

- линија 1: стринг s ,
- линија 2: цел број k , по кој следат k цели броеви c_0, \dots, c_{k-1} .