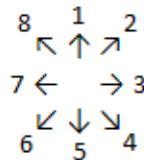


Проблем 1. Осмосмерка

Дата је осмосмерка која уместо слова има бројеве (од 0 до 10^4). Нису познате речи које треба да се нађу у осмосмерци, али зато знамо за сваку реч где почиње, колико је дугачка и у ком смеру се простире. Ваш задатак је да помоћу тих података решите осмосмерку, односно да пронађете поља која не припадају ниједној од тих речи (поља која би остала непрецртана).

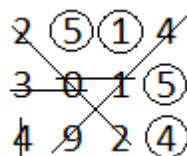
Улаз. (Улазни подаци се учитавају из датотеке `osmosmerka.in.`) У првом реду улазне датотеке се налазе димензије осмосмерке n и m ($n, m \leq 100$). Затим се у сваком од наредних n редова налази по m бројева - они представљају садржај осмосмерке. Следи ред у коме се налази број k ($k \leq 10.000$), број речи које се налазе у осмосмерци. У сваком од наредних k редова се налазе по четири броја i, j, s и l , што значи да одговарајућа реч почиње са поља (i, j) , иде у смеру s и дужине је l . Прво поље у осмосмерци је поље $(1, 1)$. Смер је број од 1 до 8 и сваки од њих одговара смеровима као на слици.



Израз. (Изразни подаци се исписују у датотеку `osmosmerka.out.`) У првом реду излазне датотеке треба да се налази број t , укупан број непрецртаних поља осмосмерке. Затим у наредних t редова треба исписати вредности у тим пољима, оним редоследом којим се јављају ако се посматра ред по ред осмосмерке, сваки од њих слева на десно.

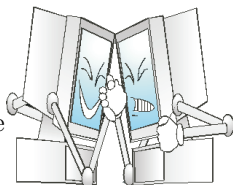
Пример 1.

<code>osmosmerka.in</code>	<code>osmosmerka.out</code>
3 4	4
2 5 1 4	5
3 0 1 5	1
4 9 2 4	5
5	4
3 3 8 3	
3 2 2 3	
2 3 7 2	
3 1 1 1	
2 1 3 2	



Објашњење. На слици је приказано како изгледа решена осмосмерка. Може се видети да су непрецртани бројеви редом 5, 1, 5, 4.

Напомена. У 40% тест примера постојаће само смерови 1, 3, 5, 7.



Проблем 2. Бешење

Мирко и Славко играју игру бешења. Игра започиње тако што Мирко замишља реч из датог речника и записује по једну цртицу за свако слово замишљене речи. Тада Славко почиње са погађањем: Славко набраја слова за која мисли да се могу наћи у замишљеној речи. Уколико Славко погоди, сваку цртицу која је пре замењивала то слово Мирко замењује са њим. Уколико се Славково слово не налази у речи, Мирко добија један поен.

Догађај	Стање на папиру
Мирко је замислио реч ВАКІСА.	-----
Славко пита за појављивање слова А.	-А---А
Славко пита за појављивање слова Е.	-А---А
Славко пита за појављивање слова В.	ВА---А

Пример тока игре.

Међутим, Мирко и Славко су морали да прекину игру у неком тренутку. Како Мирко није желео да открије своју реч, Славко је решио да сам израчуна колико речи из речника могу бити тражене речи. Помозите Славку и напишите програм који ће решити Славкове муке.

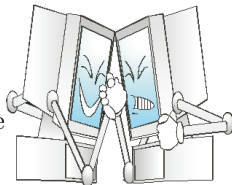
Улаз. (Улазни подаци се учитавају из датотеке `besenje.in.`) У првом реду улазне датотеке налазе се два природна броја n и m ($1 \leq n \leq 100.000$, $0 \leq m \leq 25$) који представљају број речи у речнику и број слова за које је Славко питао, редом. У другом реду улаза налази се низ од m великих слова енглеског алфабета, раздвојених по једним знаком размака, који представљају слова за чије појављивање је Славко питао. Следећи ред садржи реч састављену од великих слова енглеског алфабета и знака "-", која представља стање на папиру на крају игре. Наредних n линија садрже по једну реч из речника. Речи су састављена од великих слова енглеског алфабета дужина не већих од 30.

Изназ. (Изназни подаци се исписују у датотеку `besenje.out.`) У првом и једином реду исписати број речи који задовољавају дато стање и низ погађања.

Пример 1.

```
besenje.in                               besenje.out
3 3                                       1
A C D
--CA
KUSA
ZGRADA
MACA
```

Објашњење. Једина реч која задовољава стање је реч "KUSA".



Проблем 3. Упити

Дат је низ од n бројева. Над низом се извршавају, један за другим, m упита једног од следећа два типа:

- 1 i - сече тренутни низ после i -тог елемента и затим други део низа (од $(i + 1)$ -вог елемента до краја) ставља на почетак, при чему се добија нови низ од n елемената. Нпр. за низ 5 3 10 8 8, упит '1 2' даје 5 3 | 10 8 8 \rightarrow 10 8 8 | 5 3 \rightarrow 10 8 8 5 3.
- 2 j - треба одговорити који се број налази на j -тој позицији у тренутном низу.

Одговорити на све упите типа 2.

Улаз. (Улазни подаци се читавају из датотеке `upiti.in.`) У првом реду улазне датотеке налазе се 2 природна броја n и m који представљају, редом, број елемената низа и број упита ($1 \leq n, m \leq 10^5$). Следећи ред садржи n целих бројева - елементе низа у датом редоследу (сви елементи су из $[0, 10^9]$). Следећих m редова садрже упите већ описаног формата: ' $a b$ ' где је $a \in \{1, 2\}$ и $1 \leq b \leq n$. Упити се извршавају у редоследу датим на улазу.

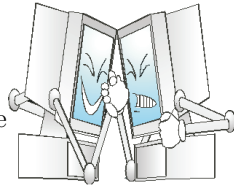
Израз. (Изразни подаци се исписују у датотеку `upiti.out.`) За сваки упит типа 2 из улазне датотеке исписати у нови ред изразне датотеке одговор на тај упит (одговоре исписивати у одговарајућем редоследу). Постојаће бар један упит типа 2.

Пример 1.

<code>upiti.in</code>	<code>upiti.out</code>
5 4	10
5 3 10 8 8	8
2 3	
1 2	
1 4	
2 3	

Објашњење. На трећој позицији у почетном низу је број 10. После два сечења низ постаје 3 10 8 8 5. На трећој позицији у овом низу је број 8.

Напомена. У 30% тест примера биће $n, m \leq 10^3$.



Проблем 4. Робот

Имамо робота коме се могу задати 3 различите команде:

- P - говори роботу да се помери за 1 метар унапред у правцу у коме је окренут,
- L - говори роботу да се окрене за 90° у лево у месту (локација му се не мења),
- D - говори роботу да се окрене за 90° у десно у месту (локација му се не мења).

Дато је K блокова од по N оваквих команди, блокови се могу извршити у било ком редоследу, док се команде унутар једног блока морају извршити у редоследу у коме су дате. Сви блокови се морају извршити. Одредити у колико различитих редоследа извршавања блокова команди ће се робот на крају наћи назад у истом месту из кога је пошао.

Улаз. (Улазни подаци се учитавају из датотеке `robot.in`.) У првом реду улазне датотексе са налазе два броја K и N ($1 \leq K \leq 8$, $1 \leq N \leq 100.000$). У следећих K редова налази се по N знакова од којих је сваки P, L или D.

Издаз. (Издазни подаци се исписују у датотеку `robot.out`.) У првом и једином реду издаза исписује се тражени број редоследа извршавања блокова команди.

Пример 1.

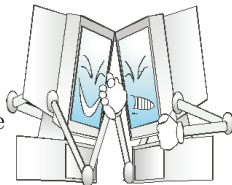
<code>robot.in</code>	<code>robot.out</code>
3 4	1
PPLP	
PLPD	
LPLD	

Објашњење. Ако је робот на почетку окренут ка северу, дати су сви могући редоследи извршавања блокова команди и релативна позиција робота на крају у односу на почетну тачку.

- 1 - 2 - 3 [PPLP—PLPD—LPLD] 0 метара на север, 2 метара на запад
- 1 - 3 - 2 [PPLP—LPLD—PLPD] 0 метара на север, 0 метара на исток
- 2 - 1 - 3 [PLPD—PPLP—LPLD] 2 метара на север, 2 метара на запад
- 2 - 3 - 1 [PLPD—LPLD—PPLP] 0 метара на север, 4 метара на запад
- 3 - 1 - 2 [LPLD—PPLP—PLPD] 2 метара на југ, 2 метара на запад
- 3 - 2 - 1 [LPLD—PLPD—PPLP] 2 метара на југ, 4 метара на запад

Само у другом случају када се блокови извршавају у редоследу 1 - 3 - 2 се робот на крају налази у истој тачки одакле је почео.

Напомена. У 30% тест примера биће $K \leq 6$ и $N \leq 1000$.



Проблем 5. Магацин

Добили сте посао магационера у новом магацину кутија! Магацин је огроман и у њему се налази n гомила кутија, у свакој гомили су кутије поређане једна на другу. Међутим, неке гомиле су превисоке а неке прениске, а ви волите ред, па сте решили да прераспоредите неке кутије.

На располагању вам је виљушкар који одједном може да преноси тачно k кутија (ни мање ни више). Према томе, у једном пребацивању можете узети тачно k кутија са неке гомиле (која има бар k кутија) и пребацити их на било коју другу гомилу. Желите да извршите неколико пребацивања тако да на крају добијете n што приближнијих гомила, тј. да **разлика броја кутија на највећој и најмањој гомили буде минимална**. Одредите ту разлику.

Улаз. (Улазни подаци се учитавају из датотеке `magacin.in`.) У првом реду улазне датотеке налазе се 2 природна броја n и k који представљају, редом, број гомила и капацитет виљушкара ($1 \leq n, k \leq 10^6$). Следећи ред садржи n бројева a_i раздвојених размаком - број кутија на одговарајућим гомилама ($1 \leq a_i \leq 10^9$).

Излаз. (Излазни подаци се исписују у датотеку `magacin.out`.) У првом и једином реду излазне датотеке исписати минималну могућу разлику између броја кутија на највећој и најмањој гомили после оптималног низа пребацивања.

Пример 1.

<code>magacin.in</code>	<code>magacin.out</code>
5 7	6
20 3 8 19 29	

Објашњење. Уколико пребацимо 7 кутија са прве на другу гомилу, 7 кутија са пете на другу и 7 кутија са пете на трећу, добијамо гомиле 13 17 15 19 15 где је разлика између највеће и најмање $19 - 13 = 6$. Ниједан други низ пребацивања не даје мању разлику.