

ToPAS'13

Torneio de Programação para Alunos do Secundário

Departamento de Ciência de Computadores

<http://www.dcc.fc.up.pt/topas/>

Conjunto de Problemas



Faculdade de Ciências da Universidade do Porto

10 de maio de 2013

Este conjunto de problemas deverá conter sete (7) problemas e dezasseis (16) páginas.
Se faltar algum problema, por favor avise a organização.



Edição realizada em colaboração com o Departamento de Engenharia Electrónica e Informática, Faculdade de Ciências e Tecnologia, Universidade do Algarve.

ToPAS'13

Torneio de Programação para Alunos do Secundário

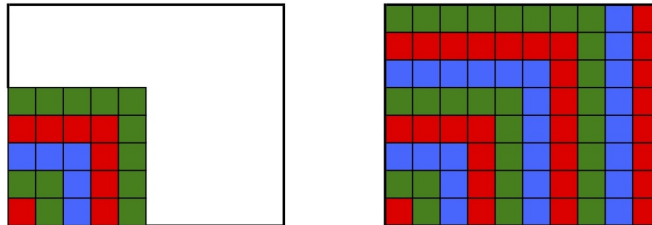
Dep. Ciência de Computadores – FCUP
10 de Maio de 2013

Conteúdo

| | |
|--|----|
| Problema A: Mãe, mudei o chão | 3 |
| Problema B: Arte ou engenho? | 5 |
| Problema C: Sopinha de letras | 7 |
| Problema D: Mas que calor! | 9 |
| Problema E: Topqui | 11 |
| Problema F: Bilheteira Boas-Fitas | 13 |
| Problema G: Calimero no túnel | 15 |

Problema A

Mãe, mudei o chão



João fartou-se do monótono chão do seu quarto, todo em ripas de madeira castanha, e decidiu substituí-lo por algo mais alegre. Como aprendeu há pouco o modelo de cores RGB – R para *red*, vermelho, G para *green*, verde, e B para *blue*, azul – optou por fazer ele próprio um chão com mosaicos dessas três cores fundamentais, com a certeza de que a sua mãe não se oporia.

Para não se perder nas operações, planeou a obra cuidadosamente, assim: começando por um dos cantos, colocar lá um mosaico vermelho; depois, à volta desse mosaico vermelho, uma fiada de mosaicos verdes; depois, à volta da fiada de mosaicos verdes, uma fiada de mosaicos azuis; depois, uma fiada de mosaicos vermelhos; e assim por diante, até o chão estar todo coberto.

Os desenhos acima ilustram o estado do chão, depois da segunda fiada verde e depois de estar já completo. Nesse caso, o quarto tem comprimento 10 e largura 8.

A questão é: para cobrir todo o chão, quantos mosaicos de cada cor deve o João comprar?

Tarefa

Escrever um programa que, dados o comprimento e a largura do quarto, calcule o número de mosaicos vermelhos, verdes e azuis necessários para cobrir o chão do quarto usando o esquema indicado. Os mosaicos são quadrados e o lado mede uma unidade.

Input

Uma linha com dois números inteiros: o primeiro é o comprimento do quarto e o segundo a largura. São ambos maiores que zero e menores ou iguais a 10000.

Output

Terá uma linha onde surgem três números inteiros, separados por um espaço: o primeiro representa o número de mosaicos vermelhos, o segundo o número de mosaicos verdes e o terceiro o número de mosaicos azuis.

Exemplo 1

Input

5 5

Output

8 12 5

Exemplo 2

Input

8 10

Output

29 27 24

Exemplo 3

Input

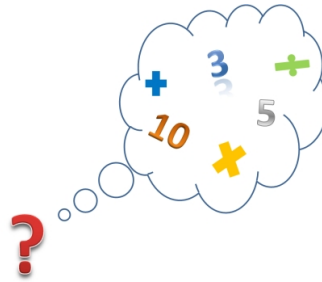
5 2

Output

3 5 2

Problema B

Arte ou engenho?



– Mário, eu consigo adivinhar números por telepatia. – disse a Marta. – Pensa num número e não mo digas.

O Mário acenou com a cabeça, e respondeu:

– Já pensei.

– Se o número que pensaste é par, multiplica por cinco; senão, multiplica por sete - continuou a Marta. - Agora, soma-lhe cinco; multiplica por dois; soma-lhe 10; divide por dois, já está? Soma cinco outra vez e multiplica por três. Ainda não te perdeste, pois não? Bem, se o número em que tinhas pensado era ímpar, soma-lhe agora o quádruplo desse número. Já está quase, só falta dividires por cinco. Ora agora, acho que já pensaste suficientemente no número para eu o adivinhar. Quanto é que te deu?

– Trinta e três. – responde o Mário, curioso.

– Aah! Ah! – exclama a Marta, vitoriosa mas cansada de tantas contas de cabeça – Então pensaste no número oito, certo?

– Foi, foi. – assentiu o Mário, muito intrigado. E disse para com os seus botões: “Isto será arte ou engenho? Telepatia ou truque?”

Tarefa

Fazer um programa para ajudar a Marta a adivinhar o número que o Mário pensou. O programa deve determinar o número que o Mário pensou.

Input

Um inteiro positivo que representa a resposta do Mário.

Output

Uma linha com um número inteiro que representa a resposta da Marta, adivinhando o número em que o Mário pensou inicialmente. Caso não exista um número inteiro que possa dar origem ao número que o Mário obteve, a Marta responderá “Acho que te enganaste nas contas.” (sem aspas).

Exemplo 1

Input

27

Output

6

Exemplo 2

Input

24

Output

3

Exemplo 3

Input

30

Output

Acho que te enganaste nas contas.

Problema C

Sopinha de letras

| | | | | | |
|---|---|---|---|---|---|
| A | T | R | Y | K | L |
| Q | D | X | I | E | T |
| A | E | X | T | V | A |
| M | O | R | L | K | Z |
| Z | A | P | O | S | I |
| S | S | T | O | A | S |

A sopa de letras é um género de puzzle usado muitas vezes como exercício, para melhorar o vocabulário dos alunos num certo domínio. Este puzzle consiste numa matriz de letras onde se formam palavras, podendo estas ocorrer numa mesma linha, coluna ou diagonal. Para tornar o desafio mais interessante as palavras podem até estar escritas de trás para a frente. Às vezes é tão difícil encontrar todas as palavras!

Tarefa

Escrever um programa que lê uma matriz de letras e um conjunto de palavras e escreve a localização da primeira última letra de cada palavra lida. As letras das palavras estão em posições contíguas da mesma linha, coluna ou diagonal, podendo estar na ordem inversa. Pode-se assumir que cada palavra só ocorre uma vez, nenhuma palavra é uma sub-palavra de outra, embora as mesmas posições da matriz possam ser usadas em várias palavras. As células da matriz são referenciadas pela sua linha e coluna, sendo a linha número 1 a primeira contar do topo e a coluna número 1 a mais à esquerda.

Input

A primeira linha contém dois inteiros, m e n , separados por um espaço, ambos maiores ou iguais a 4 e menores ou iguais a 20. Esses inteiros representam as dimensões da matriz, sendo m o número de linhas e n o número de colunas.

As m linhas seguintes contêm sequências de caracteres com n letras. Segue-se uma linha com um inteiro k , que é o número de palavras na matriz, e k linhas contendo sequências de caracteres, que são as palavras a procurar na sopa de letras, sendo $1 \leq k \leq 100$.

Output

É constituído por k linhas, cada uma com quatro inteiros separados por um espaço, $L_i C_i L_f C_f$, que identificam a posição da palavra correspondente (segundo a ordem de leitura), sendo L_i a linha

inicial da palavra, C_i a coluna inicial, e L_f e C_f a linha final e a coluna final, com $1 \leq L_i \leq m$, $1 \leq C_i \leq n$, $1 \leq L_f \leq m$, e $1 \leq C_f \leq n$.

Exemplo

Input

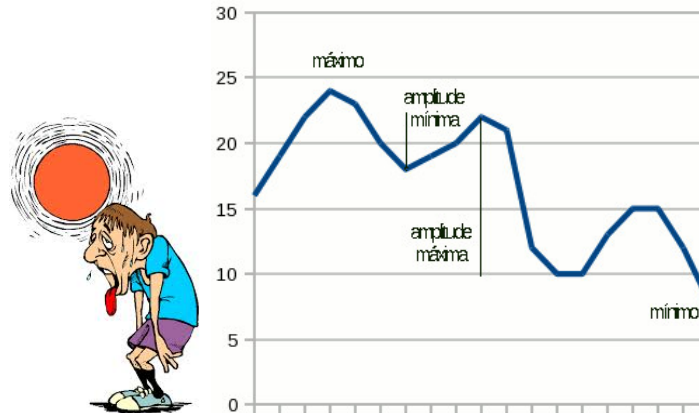
```
6 6
ATRYKL
QDXIET
AEXTVA
MORLKZ
ZAPOSI
SSTOAS
3
SOPA
DE
LETRAS
```

Output

```
5 5 5 2
2 2 3 2
1 6 6 1
```


Problema D

Mas que calor!



Toda a gente está de acordo que o clima está a mudar. Uns acham que os dias estão mais quentes, outros garantem que são as noites que já não estão tão frias, outros ainda que são as amplitudes térmicas (a diferença entre a temperatura máxima e mínima consecutivas) que aumentaram, ou talvez o contrário, agora não se distinguem as temperaturas. Para tirar isto a limpo vamos fazer um programa que determine todos estes máximos, mínimos e variações.

Tarefa

Escrever um programa que calcule o máximo, o mínimo, a amplitude máxima e a amplitude mínima de uma sequência de valores de temperaturas. As temperaturas são dadas por valores inteiros em graus centígrados. A amplitude é a diferença entre um máximo local e um mínimo local consecutivos. Um valor é um máximo local se for maior ou igual aos seus vizinhos de ambos os lados. Do mesmo modo, um valor é um mínimo local se for menor ou igual aos seus vizinhos. De notar que o valor máximo ou mínimo se poderá manter ao longo de leituras consecutivas, como acontece no exemplo com o mínimo local onde é atingida a amplitude máxima. Podemos assumir que os valores iniciais e finais da série dada são sempre um máximo ou mínimo local. Assumimos também que não há três registos consecutivos iguais.

Input

Uma primeira linha com o inteiro n , com $10 \leq n \leq 10000$, seguida de n de linhas, cada uma com um inteiro t_i , que é o valor da temperatura registada, sendo $-100 < t_i < 100$.

Output

Uma única linha com quatro inteiros separados por espaços e que representam, o máximo, o mínimo, a amplitude máxima, e a amplitude mínima dos valores lidos.

Exemplo

Input

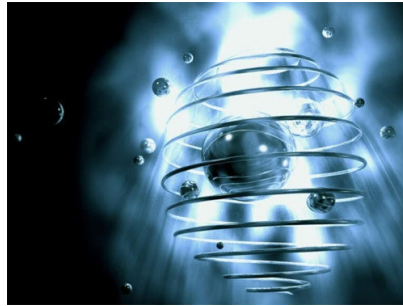
19
16
19
22
24
23
20
18
19
20
22
21
12
10
10
13
15
15
12
8

Output

24 8 12 4

Problema E

Topqui



Na galáxia Andrómeda há um torneio de programação quântica interplanetário (TOPQUI) muito popular, parecido com o ToPAS. Neste concurso participam imensas equipas de extraterrestres e a equipa vencedora ganha o conjunto de prémios atribuídos pelos planetas que organizam o concurso. Há algo obscuro na regra de distribuição dos prémios: (1) é o chefe da equipa que faz a distribuição dos prémios, agrupando os prémios em lotes e entregando um lote a cada membro da equipa; (2) há prémios de diversos valores; (3) a diferença entre o número de prémios de cada lote não deve ser superior a 1 e (4) o chefe da equipa não pode ficar com mais prémios do que qualquer outro membro da equipa.

Ora, também na Andrómeda, há um ditado popular que diz que “quem parte e reparte e não fica com a melhor parte, ou é tolo ou não tem arte”. Portanto, no TOPQUI, o chefe da equipa vencedora vai sempre tentar repartir os prémios de forma a ficar com o lote mais valioso que for possível.

Tarefa

Escrever um programa para ajudar o chefe da equipa vencedora a escolher o lote de prémios mais valioso que for possível, calculando o valor desse lote.

Input

O input é constituído por duas linhas. A primeira linha contém dois números inteiros que representam, respetivamente, o número de extraterrestres da equipa vencedora e o número de prémios. A segunda linha contém uma sequência de números inteiros, cada um representando o valor de um dos prémios. As regras do concurso garantem que há sempre pelo menos tantos prémios quantos os membros da equipa vencedora e que não há mais de 1000 prémios.

Output

O output é uma linha com apenas um número inteiro que representa o valor do lote de prémios que o chefe da equipa reservará para si.

Exemplo 1

Input

```
2 7
1 1 1 3 3 3 4
```

Output

```
10
```

Exemplo 2

Input

```
5 17
8 1 2 3 5 3 1 4 4 5 6 5 6 4 6 2 8
```

Output

```
22
```

Problema F

Bilheteira Boas-Fitas



As salas de cinema *Boas-Fitas, Lda* necessitam de um programa para gestão de reservas de bilhetes e estão a recrutar programadores para o desenvolverem. As salas de cinema são retangulares. Cada lugar é identificado por uma fila e coluna. As filas são identificadas por letras maiúsculas consecutivas (A a Z no máximo). As colunas são identificadas por inteiros positivos consecutivos (1 a 99 no máximo). Há dois tipos de pedidos de reserva: *reservas exatas*, em que é dada a fila e a coluna de início e o número de lugares contíguos pretendidos, e *reservas livres*, em que é dado o número de lugares contíguos pretendidos, devendo estes ser preenchidos por ordem de chegada, começando na fila A, e por ordem crescente de colunas. Os lugares reservados para um pedido ficam numa única fila.

Tarefa

Escrever um programa para processar uma lista de pedidos de reserva de bilhetes para uma sessão numa sala de cinema e dar resposta a cada pedido de acordo com as reservas anteriores.

Input

A entrada começa com dois inteiros positivos que representam o número de filas e colunas da sala. Em seguida vem o número de pedidos de reserva, seguido dos pedidos propriamente ditos, um em cada linha. Os pedidos de reserva exata começam com X seguido de uma linha e coluna, um espaço e o número de lugares. Os pedidos de reserva livre começam por L seguido do número de lugares.

Output

Em cada linha tem a resposta a um pedido de reserva, sendo as respostas apresentadas pela ordem de chegada. As respostas podem ser OK seguido de um espaço, a fila e coluna inicial e o número de lugares reservados ou NA se a reserva não pode ser efetuada.

Exemplo

Input

3
10
5
L3
XA4 3
L4
XA1 2
L2

Output

OK A1 3
OK A4 3
OK A7 4
NA
OK B1 2

Problema G

Calimero no túnel



As primeiras séries de desenhos animados Calimero estrearam na RTP nos anos 1970. Calimero era um pintainho, simpático e ingénuo, que se sentia infeliz e injustiçado. Criado pelos italianos Nino e Toni Pagot, para um anúncio de um sabão, em 1963, festeja agora os 50 anos com uma nova série em 3D. “Abusam porque sou pequenino” e “É uma injustiça, pois é” eram duas das suas frases emblemáticas. Vá-se lá saber porquê, estas frases ganharam uma nova atualidade, a par de uma outra, que tem a ver com “ver luz ao fundo do túnel”. Daria um tema para um novo episódio: Calimero, em apuros, num túnel. Conseguirá sair sem que a sua energia se esgote?

O túnel não forma ciclos e está numa região retangular definida por uma grelha. Cada posição da grelha tem uma letra T, F, X, N, S, E, O ou L, ou um traço (-, indicando que está livre). O túnel é a sequência de T's na região, tem pelo menos dois T's e termina em L (saída do túnel). No instante zero, Calimero está no início do túnel (posição com T, não adjacente a L, e com uma única adjacente T). Cada uma das restantes posições do túnel tem uma adjacente com T, que dá continuidade ao túnel, ou com L, que é a sua saída. Existe um único L na grelha e é adjacente a apenas um T. As posições com N, S, E, e O contêm focos que emitem um raio de luz (pode haver tipos repetidos). No instante zero, esses raios estão direcionados para Norte, Sul, Este e Oeste, respetivamente, e, em cada instante, rodam 90 graus no sentido horário. Calimero tem alguma energia, inicialmente, que é positiva. Em cada instante, pode avançar uma posição, perdendo uma unidade de energia. Pode recuperar energia em certas posições através de frestas (assinaladas por F). Se chegar a uma posição adjacente a uma fresta no instante em que esta está no alcance do raio emitido por algum foco, recebe a energia correspondente. A energia do raio só poderá chegar ao Calimero se entre o foco e a fresta não houver nenhum T, nem X (obstáculo) nem outros focos. A energia recebida é igual à diferença entre a intensidade do foco e a distância a que se encontra dele. É zero se esse valor for zero ou negativo. A distância é dada pelo número de células desde o foco até à posição em que Calimero está. A intensidade dos focos é igual para todos os focos. Calimero pode receber energia de vários focos em simultâneo. Se chegar a uma posição adjacente a uma fresta e não receber energia (positiva), fica sem perceber se tal se deve a algum desfasamento temporal. Por isso, se o seu nível de energia for **superior** a 4 unidades, arrisca esperar até receber algum raio (se não receber nenhum, avança ao fim de três instantes). Enquanto está à espera, perde uma unidade de energia também em cada instante. Se ficar com energia zero no instante em que é atingido por algum raio, recebe energia e prossegue. Se não for salvo por um raio, desfalece, e não sairá. Pode aguardar mesmo estando na posição adjacente a L. Os raios não se propagam no túnel. Não há focos incidentes em L. Para nunca recuar, Calimero vai assinalando por onde passou.

Tarefa

Escrever um programa para seguir o Calimero no túnel a partir do instante zero e verificar o que lhe sucederá: se desfalece dentro do túnel, em que instante tal sucedeu; se consegue sair do túnel, em que instante chega L e com que energia fica. Se ficar com energia zero no instante em que chega a L, assume-se que mesmo assim saiu do túnel.

Input

Uma linha com as dimensões m e n da grelha (m é o número de linhas e n o de colunas, e são ambos não superiores a 50). Segue-se um inteiro que define a intensidade dos focos (positivo e inferior a 150) e um outro inteiro que representa a energia de Calimero no instante zero (positivo e menor do que 5000). A seguir tem o estado inicial da grelha, dado em m linhas. Em cada linha, tem uma sequência de n símbolos formada por T, F, X, L, -, N, S, E, ou O (um O de Oeste).

Output

Se chegar a L, terá uma linha com dois inteiros separados por um espaço: o primeiro é o instante em que chega a L e o segundo a energia final. Se não, terá uma linha com um inteiro apenas, que representa o instante em que desfaleceu.

Exemplo 1

Input

```
7 9
8
10
---E-----
---S-----
---F---X-
--TTTL-X-
N-TF-----
--TE-X---
-----
```

Output

```
8 2
```

Exemplo 2

Input

```
6 7
100
3
TTTTTTL
T----F-
TF-O-X-
TTF--E-
-TS----
-T----X
```

Output

```
15 181
```

Exemplo 3

Input

```
8 10
100
3
TTTTTTTTT--
T----F-T--
T-----L--
TF-O-X----
TTF--O---N
-TS-----
-T----X--S
-----E----
```

Output

```
3
```