

CS Challenge #01

Marcos Santana | marcos_santana@ieee.org

1 Saving Energy

1.1 Desafio

Para incentivar as pessoas a economizar mais energia elétrica, são criados meios de controle maleáveis para incentivar um consumo mais limitado. As empresas cobram em geral, uma taxa/KWH (Kilo Watt-hora) para uma certa quantia limitada, e uma cobrança adicional taxa/KWH pelo excedente. Para os primeiros 1000 KWH são cobrados *taxaInicial*, e para o excedente disso é cobrada *taxaExcedente*.

Para auxiliar as pessoas a calcularem e entenderem melhor a sua conta de energia, crie um algoritmo que calcule a partir taxa/KWH iniciais e taxa/KWH excedente e o consumo, o valor que será cobrado na conta de energia elétrica.

1.2 Formato de Entrada

A primeira linha contém o valor da *taxaInicial* e *taxaExcedente* respectivamente.

A segunda linha contém um número positivo n que representa quantas operações que serão realizadas.

As linhas a seguir contém o valor *kwh* que representa a quantidade de KWH gasta.

1.3 Formato de Saída

Imprimir as n linhas com o consumo de energia elétrica seguido da cobrança.

1.4 Exemplo de Entrada

```
1 2
5
500
1000
1500
2000
2500
```

1.5 Exemplo de Saída

```
500 500
1000 1000
1500 2000
2000 3000
2500 4000
```

2 Torre de Hanói - Zero

2.1 Desafio

Zero criou seu próprio jogo da Torre de Hanói. Seu jogo segue as mesmas regras do original. A diferença de seu jogo para o original é o que o jogador pode escolher a quantidade de torres e discos, bem como escolher a posição inicial dos mesmos. Outra diferença é que para ganhar o jogo, deve-se colocar todos os pinos na torre central.

2.2 Formato de Entrada

A primeira linha contém a quantidade de torres t e discos d , respectivamente.

A segunda linha contém a disposição dos discos d nas torres t , começando sempre do maior disco para o menor.

A terceira linha contém a quantidade de jogadas qtd que o jogador poderá realizar.

As demais linhas são as jogadas, contendo a torre origem to e a torre destino td , respectivamente.

2.3 Formato de Saída

Após se iniciar o jogo, deve-se imprimir a matriz do tabuleiro, com as torres como colunas. Onde não houver discos, deve ser preenchido com zeros. Após cada jogada, deve-se mostrar uma mensagem, e depois, imprimir o tabuleiro novamente (exceto quando se ganha/perde o jogo).

As Mensagens de movimento não são mostrada caso houver alguma irregularidade no movimento.

2.3.1 Mensagens

ERRO

Quando tentamos usar pinos de origem e/ou destino que não estejam entre 1 e t : "Tanto o pino de origem como o pino de destino devem ser valores de 1 a t ".

Quando fazemos uma jogada que não é permitida pelas regras da torre de Hanói: "Jogada nao permetida".

DERROTA/VITÓRIA

Quando se esgota o numero de jogadas: "Jogadas esgotadas".

Quando se ganha o jogo: "Voce ganhou com x jogada(s)", onde x é o número de jogadas.

MOVIMENTAR DISCO

Movendo o disco da torre to para torre td .

2.4 Exemplo de Entrada

```
3 3
1 2 3
10
```



2 1
3 2
1 3
2 3
1 2
3 1
3 2
2 1
-1 -2
4 5
1 2

2.5 Exemplo de Saída

0 0 0
0 0 0
3 2 1

Movendo o disco da torre 2 para torre 1

0 0 0
2 0 0
3 0 1

Movendo o disco da torre 3 para torre 2

0 0 0
2 0 0
3 1 0

Movendo o disco da torre 1 para torre 3

0 0 0
0 0 0
3 1 2

Movendo o disco da torre 2 para torre 3

0 0 0
0 0 1
3 0 2

Movendo o disco da torre 1 para torre 2



0 0 0
0 0 1
0 3 2

Movendo o disco da torre 3 para torre 1

0 0 0
0 0 0
1 3 2

Movendo o disco da torre 3 para torre 2

0 0 0
0 2 0
1 3 0

Jogada nao permitida

0 0 0
0 2 0
1 3 0

Tanto o pino de origem como o pino de destino devem ser valores de 1 a 3

0 0 0
0 2 0
1 3 0

Tanto o pino de origem como o pino de destino devem ser valores de 1 a 3

0 0 0
0 2 0
1 3 0

Movendo o disco da torre 1 para torre 2

0 1 0
0 2 0
0 3 0

Voce ganhou com 8 jogada(s)

3 Word's Neighborhood

3.1 Desafio

Considere a seguinte vizinhança:

Figura 1: Vizinhança

a	b	c	d	e	f	g	h	i
j	k	l	m	n	o	p	q	r
s	t	u	v	w	x	y	z	

O problema consiste em classificar algumas letras e palavras, de acordo com os seguintes critérios:

- *Idênticas*: Quando escolhemos duas palavras e comparamos o seu tamanho, e também letra por letra, e concluimos que elas são iguais. Exemplo: "a" e "a" ou "coelho" e "coelho";
- *Vizinhas*: Quando escolhemos duas palavras e os seus tamanhos são iguais, além de que todas as letras da primeira palavra, são vizinhas das respectivas letras da segunda palavra. Exemplo: "abc" e "jkl";
- *Diferentes*: Quando duas palavras não são Idênticas nem Vizinhas. Exemplo: "a" e "c" ou "ab" e "abc".

3.2 Formato de Entrada

A primeira linha contém n , que será o número de testes a serem feitos.
As linhas seguintes contém duas strings de 1 até 20 letras.

3.3 Formato de Saída

Para cada teste, imprimir 1 se as strings são *Idênticas*, 2 se são *Vizinhas*, e 3 caso forem *Diferentes*.

3.4 Exemplo de Entrada

```
5
a k
a z
coelho coelho
coelho coelhos
az za
```



3.5 Exemplo de Saída

2
3
1
3
3

4 Doubles in the basketball

4.1 Desafio

No basquete, quando um jogador faz dez ou mais pontos, chamamos isso de duplo! Crie um algoritmo para auxiliar o técnico a analisar melhor os seus jogadores, baseados em 3 pontuações que eles fizeram em 3 jogos diferentes.

4.2 Formato de Entrada

A primeira linha contém n , que é o número de jogadores a serem analisados.

As linhas seguintes contém $pt1$, $pt2$ e $pt3$ que são as pontuações do 1º, 2º e 3º jogo respectivamente.

4.3 Formato de Saída

Para cada jogador, imprima:

- "nada" se em nenhum dos jogos o jogador fez uma pontuação maior que 10;
- "duplo" se o jogador fez 10 ou mais pontos em um jogo;
- "duplo-duplo" se o jogador fez 10 ou mais pontos em 2 jogos;
- "triplo-duplo" se o jogador fez 10 ou mais pontos em 3 jogos.

4.4 Exemplo de Entrada

```
4
7 8 9
10 11 12
10 0 0
10 0 11
```

4.5 Exemplo de Saída

```
nada
triplo-duplo
duplo
duplo-duplo
```

5 Soccer Brothers

5.1 Desafio

Zero tem dois filhos que jogam futebol, um deles se chama *Elton* que veste a *camisa 9*, e o outro se chama *Gabriel* que veste a *camisa 10*. Como os dois são muito parecidos, Zero tem que identificar seus filhos através do número de suas respectivas camisas. Crie um algoritmo que o auxilie com isso.

5.2 Formato de Entrada

A primeira linha contém n que representa quantos conjuntos serão analisados.

As linhas seguintes contém um conjunto de 22 números C_i , que são os números nas camisas dos jogadores.

5.3 Formato de Saída

Para cada linha de entrada, verifique se os filhos de Zero estão presentes no conjunto.

- Imprima "Nenhum" caso nenhum deles esteja no conjunto;
- Imprima "Elton" caso só ele esteja no conjunto;
- Imprima "Gabriel" caso só ele esteja no conjunto;
- Imprima "Ambos" caso os dois estejam no conjunto.

5.4 Exemplo de Entrada

```
4
11 99 88 10 19 20 12 13 33 44
11 12 13 14 15 16 66 88 19 20
20 9 55 66 77 88 10 33 44 11
12 23 34 45 56 67 78 89 91 9
```

5.5 Exemplo de Saída

```
Gabriel
Nenhum
Ambos
Elton
```


6 Brazil vs Argentina

6.1 Desafio

Nas olimpíadas, há duas formas de se fazer a pontuação de um país. A primeira é o método da contagem, onde é feita a contagem total de medalhas (independente da cor), e a segunda é pelo método da cor, onde se contabiliza número total de medalhas de ouro, usando as medalhas de prata, e depois bronze como critério de desempate.

A partir de uma conjunto de dados, verifique através de quais métodos o o Brasil ganha da Argentina.

6.2 Formato de Entrada

A primeira linha contém n indicando o número de checagens.

As linhas seguintes contém 6 valores, sendo os 3 primeiros as medalhas de ouro, prata e bronze do Brasil, respectivamente, e os 3 últimos também sendo as medalhas de ouro, prata e bronze também respectivamente, porém da Argentina.

6.3 Formato de Saída

Imprima as n linhas dizendo:

- "nenhuma caso o Brasil não ganhe por nenhum dos métodos da Argentina;
- "contagem" caso o Brasil ganhe apenas pelo método da contagem;
- "cor" caso o Brasil ganhe apenas pelo método da cor;
- "ambas" caso o Brasil ganhe pelo método da contagem e da cor.

6.4 Exemplo de Entrada

```
5
10 5 15 10 1 0
10 5 15 10 6 10
12 5 10 5 20 30
10 0 15 10 5 30
10 5 15 10 5 15
```

6.5 Exemplo de Saída

```
ambas
contagem
cor
nenhuma
nenhuma
```

7 The Real Italian Pie

7.1 Desafio

Durante as reuniões de staff do CS Challenge, Paula que detém o conhecimento de como se fazer A Verdadeira Torta Italiana, sempre levava várias pedaços para fazer alguns lanchinhos entre as pautas das reuniões. Quando era necessário, ela sempre cortava os pedaços que restavam em mais, para que durasse até o final da reunião.

Ajude Paula criando um algoritmo que diga a ela como deve cortar sua torta, para que a torta acabe apenas no final da reunião.

7.2 Formato de Entrada

A primeira linha contém n , que representa quantas reuniões são feitas;

A segunda linha (e sucessivas) contém a quantidade de pessoas $qtdPessoas$ na reunião seguida da quantidade $qtdTorta$ de pedaços de tortas feitas por Paula;

A terceira linha (e sucessivas) contém a quantidade p de pautas que tem a reunião;

A quarta linha (e sucessivas) contém quantas pessoas $qtdComeram$ que foram comer a torta em cada intervalo de pauta.

7.3 Formato de Saída

Imprimir as n reuniões e das p pautas com a quantidade de pessoas $qtdComeram$ que foram comer em cada intervalo entre as pautas, seguida pela quantidade de pedaços de tortas $qtdTorta$.

7.4 Exemplo de Entrada

```
2
20 60
8
15
10
20
18
9
12
2
10
15 100
4
1
2
3
5
```



7.5 Exemplo de Saída

15 45
10 35
20 15
18 12
9 3
12 12
2 10
10 10

1 99
2 97
3 94
5 89

8 Milhões

8.1 Desafio

Mateus está agora em um negócio próprio vendendo limonadas. Ele tem uma meta diária m de vendas de copo de limonada, e nunca vai embora antes de atingi-lá. Para isso, Mateus deve manter um rígido controle de seu estoque de limões e açúcar, para que nunca falte. Infelizmente, com a crise, o preço do limão e do açúcar se altera diariamente, então Mateus deve levar isso em conta quando for fazer sua compra de suprimentos.

Ele compra os limões de maneira individuais e açúcar em sacos de 5 Kg ($1 \text{ kg} = 16 \text{ colheres}$). Mateus compra os ingredientes na parte da manhã, antes de qualquer venda. Os limões duram por um dia por serem comprados individualmente, mas como ele compra açúcar por Kg, pode sobrar algumas colheres que podem ser utilizadas no próximo dia, já que ela não estraga.

Ajude Mateus a descobrir o custo mínimo para comprar os limões e o açúcar.

8.2 Formato de Entrada

A primeira linha contém n que será o número de casos processados.

A primeira linha de cada teste terá três inteiros positivos, sendo eles: d o número de dias em que Mateus deseja fazer a venda de limonadas; l e s sendo o número de limões e colheres de açúcar necessários para fazer um copo de limonada, respectivamente.

As seguintes linhas d conterão a meta m de Mateus, o preço do limão pl e o preço para o saco de açúcar ps em centavos, respectivamente para cada respectivo dia.

8.3 Formato de Saída

Para cada caso de teste, imprima o número mínimo de suprimentos (em centavos) necessários para que a meta de venda de Mateus seja alcançada.

8.4 Exemplo de Entrada

```
2
3 3 2
200 10 399
300 8 499
400 12 499
2 5 10
9 10 199
8 20 99
```

8.5 Exemplo de Saída

```
31977
1347
```