

[ED282] Black Friday

Neste problema deverá submeter uma classe **ED282** contendo um programa completo para resolver o problema (ou seja, com o método main). Pode assumir que no Mooshak terá acesso a todas as classes base dadas de listas, pilhas e filas dadas nas aulas (não precisa de as incluir na submissão).

Um vendedor de uma loja tecnológica tem uma enorme fila de clientes acampados à porta da loja e dispostos a comprar todas as televisões em desconto para poderem ver o mundial de futebol. Ele sabe que só pode vender um total de T televisões por dia a cada cliente. Decidiu por isso usar o seguinte algoritmo:

1. Pega no primeiro cliente da fila e vende-lhe um máximo de T televisões
 - o Se o cliente já comprou todas as televisões que queria sai da fila contente e volta para casa
 - o Caso contrário, o cliente volta para o final da fila e o número de televisões que ainda quer comprar é decrementando em T
2. Se a certa altura o primeiro cliente da fila for um cliente que já foi atendido nesse dia, o vendedor fecha a loja. Regressa depois no dia seguinte para continuar a vender televisões aos clientes que ainda estão na fila, recomeçando a partir do ponto um.

Imagina por exemplo que $T=3$ e que tinha a seguinte fila, onde o número indica a quantidade de televisões que cada cliente quer comprar. Iria acontecer o seguinte:

Dia 1

Dia 1 | Quantidade de TVs vendidas (no dia = 0 / no total = 0)

pedro 6	miguel 8	helder 10	maria 2
------------	-------------	--------------	------------

O vendedor começa por atender o *pedro*, que é movido para o final da fila com menos 3 televisões para comprar:

Dia 1 | Quantidade de TVs vendidas (no dia = 3 / no total = 3)

miguel 8	helder 10	maria 2	pedro 3
-------------	--------------	------------	------------

Vende 3 televisões a *miguel*, que é movido para o final da fila com menos 3 televisões para comprar:

Dia 1 | Quantidade de TVs vendidas (no dia = 6 / no total = 6)

helder 10	maria 2	pedro 3	miguel 5
--------------	------------	------------	-------------

Vende 3 televisões a *helder*, que é movido para o final da fila com menos 3 televisões para comprar:

Dia 1 | Quantidade de TVs vendidas (no dia = 9 / no total = 9)

maria 2	pedro 3	miguel 5	helder 7
------------	------------	-------------	-------------

Vende 2 televisões a *maria*, que tendo obtido tudo o que queria sai da fila:

Dia 1 | Quantidade de TVs vendidas (no dia = 11 / no total = 11)

pedro 3	miguel 5	helder 7
------------	-------------	-------------

Neste momento *pedro* está no início da fila e já tinha sido atendido nesse dia, pelo que o vendedor fecha a loja, vai descansar e regressa apenas no dia seguinte.

Dia 2

Dia 2 | Quantidade de TVs vendidas (no dia = 0 / no total = 11)

pedro 3	miguel 5	helder 7
------------	-------------	-------------

O vendedor atende o *pedro*, resultando em:

Dia 2 | Quantidade de TVs vendidas (no dia = 3 / no total = 14)

miguel	helder
5	7

O vendedor atende o *miguel*, resultando em:

Dia 2 | Quantidade de TVs vendidas (no dia = 6 / no total = 17)

helder	miguel
7	2

O vendedor atende o *helder*, resultando em:

Dia 2 | Quantidade de TVs vendidas (no dia = 9 / no total = 20)

miguel	helder
2	4

Neste momento *miguel* está no início da fila e já tinha sido atendido nesse dia, pelo que o vendedor fecha a loja, vai descansar e regressa apenas no dia seguinte.

Dia 3

Dia 3 | Quantidade de TVs vendidas (no dia = 0 / no total = 20)

miguel	helder
2	4

O vendedor atende o *miguel*, resultando em:

Dia 3 | Quantidade de TVs vendidas (no dia = 2 / no total = 22)

helder
4

O vendedor atende o *helder*, resultando em:

Dia 3 | Quantidade de TVs vendidas (no dia = 5 / no total = 25)

helder
1

Neste momento *helder* está no início da fila e já tinha sido atendido nesse dia, pelo que o vendedor fecha a loja, vai descansar e regressa apenas no dia seguinte.

Dia 4

Dia 4 | Quantidade de TVs vendidas (no dia = 0 / no total = 25)

helder
1

O vendedor a última televisão ao *helder*, esvaziando a fila:

Dia 4 | Quantidade de TVs vendidas (no dia = 1 / no total = 26)

Fila vazia

A tua tarefa é simular este processo, escrevendo de cada vez que um cliente sai da fila NOME_CLIENTE DIA TVS_DIA TVS_TOTAL onde *DIA* é o dia em que a cliente comprou a sua última televisão, *TVS_DIA* é a quantidade de TVs vendida nesse dia e *TVS_TOTAL* é a quantidade total de TVs vendida (as quantidades de TVs devem ser no momento em que o cliente sai da fila, depois de ter comprado todas as suas TVs). Se fosse o exemplo dado anteriormente, o output devia ser:

```
maria 1 11 11
pedro 2 3 14
miguel 3 2 22
helder 4 1 26
```

Dicas: (é livre para fazer como quiser, incluindo usar listas do próprio Java, mas é sugerido fazer da seguinte maneira):

- Crie uma classe *Cliente* para conter um cliente, com dois atributos: *name* e *tvs* (nome e número de televisões que ainda quer comprar)
- Para representar as dúvidas, use uma fila (*MyQueue<Client>*).
- Mantenha três variáveis indicando o dia atual, a quantidade de TVs vendida no dia e quantidade total de TVs já vendidas.
- Simule o processo descrito em cima. Por exemplo, se o cliente ainda quer comprar mais do que *T* televisões, decremente *tvs* e adicione novamente o cliente ao final da fila...

Input

A primeira linha contém um inteiros **T** ($1 \leq T \leq 100$), o máximo diário de televisões que podem ser vendidas ao mesmo cliente. A segunda linha contém um inteiro **N**, o número de clientes na fila. Seguem-se **N** linhas ($1 \leq N \leq 100$) descrevendo os clientes na ordem em que estão inicialmente na fila, cada linha no formato **NOME_CLIENTE NUMERO_TVS**. O nome é constituído unicamente por letras minúsculas (e tem tamanho inferior a 10) e **TVS** é um inteiro positivo menor ou igual a 100.

Output

O output deve conter **N** linhas, descrevendo os clientes, pela ordem em que cada uma das suas últimas dúvidas foram sendo esclarecidas, no formato **NOME_CLIENTE DIA TVS_DIA TVS_TOTAL**, como foi atrás escrito.

Exemplo de Input/Output

Input	Output
3	maria 1 11 11
4	pedro 2 3 14
pedro 6	miguel 3 2 22
miguel 8	helder 4 1 26
helder 10	
maria 2	

Teste Prático de Estruturas de Dados (CC1007)