**Problema**

Balanceamento de carga é muito importante em ambientes Cloud. Estamos sempre tentando minimizar os custos para que possamos manter o número de servidores o menor possível. Em contrapartida a capacidade e performance aumenta quando adicionamos mais servidores. Em nosso ambiente de simulação, em cada tick (unidade básica de tempo da simulação), os usuários conectam aos servidores disponíveis e executam uma tarefa. Cada tarefa leva um número de ticks para ser finalizada (o número de ticks de uma tarefa é representado por ttask ), e após isso o usuário se desconecta automaticamente.

Os servidores são máquinas virtuais que se auto criam para acomodar novos usuários. Cada servidor custa R$ 1,00 por tick e suporta no máximo umax usuários simultaneamente. Você deve finalizar servidores que não estão sendo mais usados. O desafio é fazer um programa em Python que recebe usuários e os aloca nos servidores tentando manter o menor custo possível.

**Input**

* Um arquivo onde: a primeira linha possui o valor de ttask ;
* a segunda linha possui o valor de umax ;
* as demais linhas contém o número de novos usuários para cada tick.

**Output**

Um arquivo onde cada linha contém uma lista de servidores disponíveis no final de cada tick, representado pelo número de usuários em cada servidor separados por vírgula e, ao final, o custo total por utilização dos servidores.

**Limites**

1 ≤ ttask ≤ 10

1 ≤ umax ≤ 10