

Nome do arquivo: jogo.x, onde x deve ser c, cpp, java, js ou py

Jogo de par ou ímpar

Dois amigos, Alice e Bob, estão jogando um jogo muito simples, em que um deles grita ou "par" ou "ímpar" e o outro imediatamente responde ao contrário, respectivamente "ímpar" ou "par". Em seguida, ambos exibem ao mesmo tempo uma mão cada um, em que alguns dedos estão estendidos e outros dobrados. Então eles contam o número total de dedos estendidos. Se a soma for par, quem gritou "par" ganha. Se a soma for ímpar, quem gritou "ímpar" ganha. Por exemplo, suponhamos que a Alice gritou "par" e o Bob respondeu "ímpar". Em seguida, Alice não deixou nenhum dos seus dedos estendidos, ao passo que Bob deixou três dedos estendidos. A soma então é três, que é ímpar, portanto Bob ganhou. Seu programa deve determinar quem ganhou, tendo a informação de quem gritou par e o número de dedos estendidos de cada um.

Entrada

A entrada contém três linhas, cada uma com um número inteiro, P, D_1 e D_2, nesta ordem. Se P = 0 então Alice gritou "par", ao passo que se P=1 então Bob gritou "par". Os números D_1 e D_2 indicam, respectivamente, o número de dedos estendidos da Alice e do Bob.

Saída

Seu programa deverá imprimir uma única linha, contendo um único número inteiro, que deve ser 0 se Alice foi a ganhadora, ou 1 se Bob foi o ganhador.

Restrições

- $P = 0$ ou $P = 1$
- $0 \leq D_1 \leq 5$
- $0 \leq D_2 \leq 5$

Exemplo de entrada 1 0 0 3	Exemplo de saída 1 1
Exemplo de entrada 2 1 0 3	Exemplo de saída 2 0
Exemplo de entrada 3 0 1 5	Exemplo de saída 3 0

Nome do arquivo: *pedagio.x*, onde *x* deve ser *c*, *cpp*, *java*, *js* ou *py*

Pedágio

A invenção do carro tornou muito mais rápido e mais barato realizar viagens de longa distância. Realizar uma viagem rodoviária tem dois tipos de custos: cada quilômetro percorrido na rodovia tem um custo associado (não só devido ao consumo de combustível mas também devido ao desgaste das peças do carro, pneus, etc.), mas também é necessário passar por vários pedágios localizados ao longo da rodovia.

Os pedágios são igualmente espaçados ao longo da rodovia; o começo da estrada não possui um pedágio, mas o seu final pode estar logo após um pedágio (por exemplo, se a distância entre dois pedágios consecutivos for de 37 km e a estrada tiver 111 km, o motorista deve pagar um pedágio aos 37 km, aos 74 km e aos 111 km, logo antes de terminar a sua viagem).

Tarefa

Dadas as características da rodovia e os custos com gasolina e com pedágios, calcule o custo total da viagem.

Entrada

A entrada consiste de duas linhas. A primeira linha da entrada contém dois inteiros L e D ($1 \leq L, D \leq 10^4$), indicando o comprimento da estrada e a distância entre pedágios, respectivamente. A segunda linha contém dois inteiros K e P ($1 \leq K, P \leq 10^4$), indicando o custo por quilômetro percorrido e o valor de cada pedágio. O primeiro pedágio está localizado no quilômetro D da estrada (ou seja, a distância do início da estrada para o primeiro pedágio é D quilômetros).

Saída

Seu programa deve imprimir uma única linha contendo um único inteiro, indicando o custo total da viagem.

Exemplo de entrada 1 111 37 1 10	Exemplo de saída 1 141
Exemplo de entrada 2 100 30 3 14	Exemplo de saída 2 342
Exemplo de entrada 3 20 70 9 17	Exemplo de saída 3 180

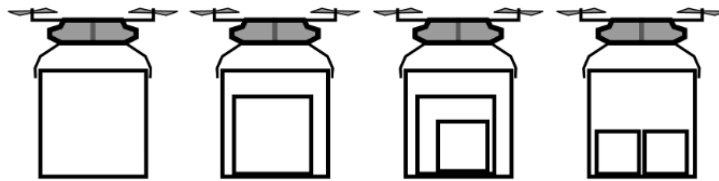
Nome do arquivo: caixas.x, onde x deve ser c, cpp, java, js ou py

Entrega de Caixas

Você precisa transportar três caixas vazias usando um drone que pode levantar uma caixa por vez apenas em cada viagem. Quer dizer, sempre dá para transportar as três caixas vazias fazendo três viagens do drone. Mas talvez dê para fazer menos do que três viagens, se for possível colocar uma caixa dentro de outra. As caixas têm formato de cubo e a única restrição para uma caixa ser colocada dentro de outra é o tamanho, não importando o peso.

Uma caixa de tamanho X pode ser colocada dentro de uma caixa de tamanho Y se $X < Y$. Note, portanto, que uma caixa não cabe dentro de outra do mesmo tamanho. Além disso, duas caixas de tamanhos X e Y podem ser colocadas, lado a lado, dentro de uma caixa de tamanho Z se $(X+Y) < Z$.

A figura ilustra as quatro configurações possíveis para o drone fazer uma viagem.



Neste problema, os tamanhos das três caixas são dados em ordem crescente e seu programa deve computar o número mínimo de viagens que o drone pode fazer para transportar todas as três caixas.

Entrada

A primeira linha da entrada contém um inteiro A . A segunda linha da entrada contém um inteiro B . A terceira linha da entrada contém um inteiro C . Os três inteiros representam os tamanhos das três caixas.

Saída

Seu programa deve imprimir uma linha contendo um inteiro, representando o número mínimo de viagens que o drone pode fazer para transportar todas as três caixas.

Restrições

- $1 \leq A \leq B \leq C \leq 1000$

Exemplos

Entrada	Saída
12 45 188	1

Entrada	Saída
67 67 67	3

Entrada	Saída
111 463 463	2

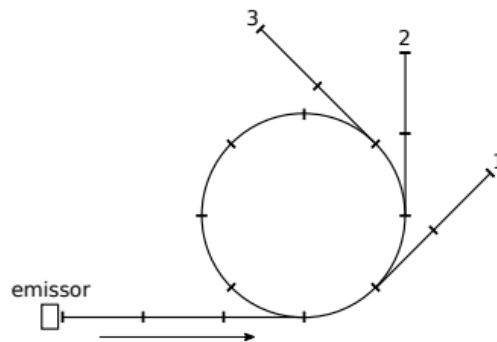
Entrada	Saída
72 72 345	1

Entrada	Saída
72 72 345	1

Nome do arquivo: "acelerador.x", onde x deve ser c, cpp, pas, java, js ou py

Acelerador de Partículas

A universidade está inaugurando um grande acelerador de partículas, com um emissor e três sensores, numerados 1, 2 e 3. Uma partícula, após sair do emissor, entra no acelerador onde pode dar várias voltas sendo acelerada a velocidades muito altas. Num determinado momento, a partícula sai do acelerador por uma das três saídas, atingindo um dos sensores. A figura mostra o caminho por onde as partículas trafegam, com uma graduação de 1 quilômetro. Por exemplo, do emissor até o acelerador são 3 quilômetros e a circunferência do acelerador tem 8 quilômetros.



Neste problema, será dada a distância total, em quilômetros, percorrida por uma certa partícula trafegando do emissor até algum sensor e seu programa deve determinar qual sensor foi atingido pela partícula. Por exemplo, veja que se a distância total for 23 quilômetros, então a partícula tem que ter atingido o sensor 2.

Entrada

A entrada consiste de apenas uma linha contendo um inteiro D , representando a distância total percorrida pela partícula

Saída

Seu programa deve imprimir uma linha contendo um inteiro, representando o número do sensor que a partícula atingiu.

Restrições

- $6 \leq D \leq 800008$. D sempre será a distância total percorrida entre o emissor e algum sensor.

Exemplos

Entrada 23	Saída 2
----------------------	-------------------

Entrada 6	Saída 1
---------------------	-------------------

Entrada 9192	Saída 3
------------------------	-------------------