

1. BITS TROCADOS (OLIMPIÁDA BRASILEIRA DE INFORMÁTICA)

Nome do arquivo fonte: bits.c, bits.cpp, bits.pas, bits.java, bits.py

As ilhas Weblands formam um reino independente nos mares do Pacífico. Como é um reino recente, a sociedade é muito influenciada pela informática. A moeda oficial é o Bit: existem notas de B\$ 50,00, B\$ 10,00, B\$ 5,00e B\$ 1,00. Você foi contratado(a) para ajudar na programação dos caixas automáticos de um grande banco das ilhas Weblands.

1. Tarefa

Os caixas eletrônicos das ilhas Weblands operam com todos os tipos de notas disponíveis, mantendo um estoque de cédulas para cada valor (B\$ 50,00, B\$ 10,00, B\$ 5,00e B\$ 1,00). Os clientes do banco utilizam os caixas eletrônicos para efetuar retiradas de um certo número inteiro de Bits.

Sua tarefa é escrever um programa que, dado o valor de Bits desejado pelo cliente, determine o número de cada uma das notas necessário para totalizar esse valor, de modo a minimizar a quantidade de cédulas entregues. Por exemplo, se o cliente deseja retirar B\$ 50,00, basta entregar uma única nota de cinquenta Bits. Se o cliente deseja retirar B\$ 72,00, é necessário entregar uma nota B\$ 50,00, duas de B\$ 10,00 e duas de B\$ 1,00.

2. Entrada

A entrada é composta por vários conjuntos de teste. Cada conjunto de teste é composto por uma única linha, que contém um número inteiro positivo V, que indica o valor solicitado pelo cliente. O final da entrada é indicado por V=0.

Exemplo de Entrada:

```
1
72
0
```

3. Saída:

Para cada conjunto de teste de entrada seu programa deve produzir três linhas na saída. A primeira linha deve conter um identificador do conjunto de teste, no formato “Teste n”, onde n é numerado a partir de 1. Na segunda linha devem aparecer o conjunto de quatro inteiros I, J, K, L que representam o resultado encontrado pelo seu programa: I indica o número de cédulas de B\$ 50,00, J indica o número de cédulas de B\$ 10,00, K indica o número de cédulas de B\$ 5,00 e L indica o número de cédulas de B\$ 1,00. A terceira linha deve ser deixada em branco. A grafia mostrada no exemplo abaixo deve ser seguida rigorosamente.

Exemplo de Saída:

```
Teste 1
0 0 0 1
```

```
Teste 2
1 2 0 2
```

Esta saída corresponde ao exemplo da entrada mostrada no Exemplo de Entrada. Restrições: $0 \leq V \leq 10000$ (V=0 apenas para indicar o fim da entrada).

2. METEOROS (OLIMPÍADA BRASILEIRA DE INFORMÁTICA)

Nome do arquivo fonte: meteoros.c, meteoros.cpp, meteoros.pas, meteoros.java, meteoros.py

Em noites sem nuvens, pode-se muitas vezes observar pontos brilhantes no céu que se deslocam com grande velocidade, e em poucos segundos desaparecem de vista: são as chamadas estrelas cadentes ou meteoros. Meteoros são, na verdade, partículas de poeira de pequenas dimensões que, ao penetrar na atmosfera terrestre, queimam-se rapidamente (normalmente a uma altura entre 60 e 120 quilômetros). Se os meteoros são suficientemente grandes, podem não queimar completamente na atmosfera e dessa forma atingem a superfície terrestre: nesse caso são chamados de meteoritos.

Zé Felício é um fazendeiro que adora astronomia e descobriu um portal na Internet que fornece uma lista das posições onde caíram meteoritos. Com base nessa lista, e conhecendo a localização de sua fazenda, Zé Felício deseja saber quantos meteoritos caíram dentro de sua propriedade. Ele precisa da sua ajuda para escrever um programa de computador que fala essa verificação automaticamente.

1. Tarefa

São dados:

- Uma lista de pontos no plano cartesiano, onde cada ponto corresponde à posição onde caiu um meteorito;
- As coordenadas de um retângulo que delimita uma fazenda.

As linhas que delimitam a fazenda são paralelas aos eixos cartesianos. Sua tarefa é escrever um programa que determine quantos meteoritos caíram dentro da fazenda (incluindo meteoritos que caíram exatamente sobre as linhas que delimitam a fazenda).

2. Entrada

Seu programa deve ler vários conjuntos de testes. A primeira linha de um conjunto de testes contém quatro números inteiros X_1 , Y_1 , X_2 e Y_2 , onde (X_1, Y_1) é a coordenada do canto superior esquerdo e (X_2, Y_2) é a coordenada do canto inferior direito do retângulo que delimita a fazenda. A segunda linha contém um inteiro N , que indica o número de meteoritos. Seguem-se N linhas, cada uma contendo dois números inteiros X e Y , correspondendo às coordenadas de cada meteorito. O final da entrada é indicado por $X_1=Y_1=X_2=Y_2=0$.

Exemplo de Entrada

```
2 4 5 1
2
1 2
3 3
2 4 3 2
3
1 1
2 2
3 3
0 0 0 0
```

3. Saída

Para cada conjunto de teste da entrada, seu programa deve produzir três linhas na saída. A primeira linha deve conter um identificador do conjunto de teste, no formato “Teste n ”, onde n é numerado a partir de 1. A segunda linha deve conter o número de meteoritos que caíram dentro da fazenda. A terceira linha deve ser deixada em branco. A grafia mostrada no Exemplo de Saída, abaixo, deve ser seguida rigorosamente.

Exemplo de Saída

```
Teste 1
1
```

```
Teste 2
2
```

Esta saída corresponde ao exemplo de entrada acima.

4. Restrições

$$0 \leq N \leq 10000$$

$$0 \leq X \leq 10000$$

$$0 \leq Y \leq 10000$$

$$0 \leq X_1 \leq X_2 \leq 10000$$

$$0 \leq Y_1 \leq Y_2 \leq 10000$$

3. EXTENSÕES DE TOMADAS (OLIMPIÁDA NÓRDICA DE PROGRAMAÇÃO)

Nome do arquivo fonte: tomadas.c, tomadas.cpp, tomadas.pas, tomadas.java, tomadas.py

Roy acabou de se mudar para um novo apartamento. Bem, realmente, o apartamento em si não é muito novo, datando dos dias em que as pessoas não tinham eletricidade em suas casas. Por causa disso, o apartamento de Roy tem apenas uma única tomada na parede, então Roy pode apenas um de seus aparelhos elétricos por vez.

Roy gosta de assistir TV enquanto trabalha no computador, e ouvir se com (no máximo volume) enquanto ele trabalha, então usar uma única tomada não é uma opção. Realmente, ele quer ter todos os seus aparelhos conectados à rede elétrica ao mesmo tempo. A resposta, é claro, são as extensões de tomada, e Roy tem alguns modelos velhos destes que usava em seu antigo apartamento. Entretanto, seu antigo apartamento tinha muito mais tomadas nas paredes, então ele não está certo se suas extensões fornecerão, agora, tomadas suficientes.

Sua tarefa é ajudar Roy a contar quantos aparelhos ele pode fornecer com eletricidade, dado um conjunto de extensões. Note que, sem qualquer extensão, Roy pode ligar um único aparelho na tomada da parede. Lembre-se também, de que devemos manter o filtro de linha ligado à rede elétrica para ser usado.

Especificações de Entrada

A entrada vai começar com um único inteiro $1 \leq N \leq 20$, indicando o número de testes que se seguem. Então seguem-se as N linhas. Cada caso de teste começa com um inteiro $1 \leq K \leq 20$, indicando o número de extensões em cada caso de teste. Então seguem, na mesma linha, K inteiros separados por um único espaço, $O_1 O_2 \dots O_K$, onde $2 \leq O_i \leq 10$, indicando o número de tomadas de cada extensão.

Especificações de Saída

Deve ser produzida uma linha para cada caso de teste, com o número máximo de aparelhos que podem ser ligados.

Exemplo de entrada

```
3
3 2 3 4
10 4 4 4 4 4 4 4 4 4
4 10 10 10 10
```

Exemplo de Saída

```
7
31
37
```