Problema C - Remoções Aproximadas

A Rita tem um multiconjunto de inteiros $\boldsymbol{A}$, e um conjunto de inteiros $\boldsymbol{B}$. Ela quer apagar alguns elementos de $\boldsymbol{A}$ de tal forma que, no multiconjunto que sobra, nunca aconteça que a soma de dois elementos pertença a $\boldsymbol{B}$.

Mais precisamente, depois das remoções, para quaisquer dois elementos restantes x e y (podem ser iguais), deve verificar-se que $x + y$ **não** pertence a $\boldsymbol{B}$. Em particular, é permitido formar um par usando duas vezes o mesmo elemento restante, mesmo que esse valor apareça apenas uma vez em $\boldsymbol{A}$.

Encontrar o número mínimo de remoções é difícil. Por isso, neste problema basta apresentar uma solução *aproximada*:

- seja $\boldsymbol{OPT}$ o menor número de elementos que é preciso remover para tornar $\boldsymbol{A}$ válido;
- a tua submissão é aceite se imprimir uma remoção válida com no máximo $2 \cdot \boldsymbol{OPT}$ elementos.

Se um valor aparecer várias vezes em $\boldsymbol{A}$, cada ocorrência pode ser removida separadamente. No entanto, se imprimires um valor t vezes no output de um certo caso, esse valor tem de aparecer pelo menos t vezes no input desse caso.

Existem $\boldsymbol{T}$ casos de teste independentes e para cada um terás de imprimir uma solução.

Exemplo

Considera um exemplo em que $\boldsymbol{A} = \{1, 2, 3, 7, 10\}$ e $\boldsymbol{B} = \{4, 13\}$.

Neste caso, qualquer solução válida tem de remover o valor 2, porque $2 + 2 = 4$ pertence a $\boldsymbol{B}$.

Depois disso, se o valor 3 permanecer, continuam a existir conflitos, porque $1 + 3 = 4$ e $3 + 10 = 13$ pertencem a $\boldsymbol{B}$. Logo, para obter uma solução com apenas 2 remoções, também é necessário remover o valor 3.

De facto, removendo $\{2, 3\}$ de $\boldsymbol{A}$, ficamos com os valores $\{1, 7, 10\}$. As somas possíveis entre valores restantes são $1 + 1 = 2$, $1 + 7 = 8$, $1 + 10 = 11$, $7 + 7 = 14$, $7 + 10 = 17$ e $10 + 10 = 20$. Nenhuma delas pertence a $\boldsymbol{B} = \{4, 13\}$, logo esta remoção é válida.

Neste exemplo, $\boldsymbol{OPT} = 2$, pelo que qualquer resposta válida com até 4 remoções seria aceite. Por exemplo, remover $\{1, 2, 3, 10\}$ deixaria apenas $\{7\}$, o que também seria válido, embora não fosse ótimo.

Restrições

São garantidos os seguintes limites em todos os casos de teste que irão ser colocados ao programa:

$1 \le \boldsymbol{T} \le 1000$	número de casos de teste
$1 \le \boldsymbol{N_i} \le 10^5$	número de elementos de $\boldsymbol{A}$ no caso i
$1 \le \boldsymbol{M_i} \le 100$	número de elementos de $\boldsymbol{B}$ no caso i
$\sum_{i=1}^{\boldsymbol{T}} \boldsymbol{N_i} \le 10^5$	soma de todos os valores de $\boldsymbol{N_i}$
$-10^9 \le a_{i,j} \le 10^9$	valor do j-ésimo elemento de $\boldsymbol{A}$ no caso i
$-2 \cdot 10^9 \le b_{i,j} \le 2 \cdot 10^9$	valor do j-ésimo elemento de $\boldsymbol{B}$ no caso i
$b_{i,j} \ne b_{i,k}$ para $j \ne k$	todos os valores de $\boldsymbol{B}$ são distintos dentro de cada caso

Os casos de teste estão organizados nos seguintes grupos:

Grupo	Número de Pontos	Restrições adicionais
1	20	$\sum_{i=1}^{\boldsymbol{T}} \boldsymbol{N_i} \le 20$
2	20	$\boldsymbol{M_i} = 1$ em todos os casos
3	25	Em cada caso, todos os valores de $\boldsymbol{A}$ são distintos
4	35	Sem restrições adicionais

Formato de Input

A primeira linha contém um inteiro $\boldsymbol{T}$: o número de casos de teste.

Depois seguem $\boldsymbol{T}$ casos de teste. Cada caso tem o seguinte formato:

- uma linha com dois inteiros $\boldsymbol{N}$ e $\boldsymbol{M}$, separados por um espaço: o número de elementos de $\boldsymbol{A}$ e o número de elementos de $\boldsymbol{B}$;
- uma linha com $\boldsymbol{N}$ inteiros $a_1, a_2, \ldots, a_{\boldsymbol{N}}$, separados por espaços: os elementos de $\boldsymbol{A}$;
- uma linha com $\boldsymbol{M}$ inteiros $b_1, b_2, \ldots, b_{\boldsymbol{M}}$, separados por espaços: os elementos de $\boldsymbol{B}$.

Formato de Output

Para cada caso de teste, pela mesma ordem do input, imprime duas linhas:

- na primeira linha, um inteiro $\boldsymbol{k}$: o número de elementos que decides remover nesse caso;
- na segunda linha, $\boldsymbol{k}$ inteiros, cada um igual a um valor presente em $\boldsymbol{A}$ desse caso, descrevendo o multiconjunto removido.

Se um valor x aparecer t vezes no output de um caso, então x tem de aparecer pelo menos t vezes no input desse caso.

A tua resposta para um certo caso é aceite se:

- depois de remover exatamente essas ocorrências, não restar nenhum par x, y (possivelmente com $x = y$) tal que $x + y$ pertença a $\boldsymbol{B}$; o mesmo elemento restante pode ser usado nas duas posições do par;
- $\boldsymbol{k} \leq 2 \cdot \boldsymbol{OPT}$, onde $\boldsymbol{OPT}$ é o menor número possível de remoções nesse caso.

Nota sobre o exemplo seguinte: como este é um problema de aproximação, o output apresentado não tem de coincidir com a solução ótima. No primeiro caso da amostra, por exemplo, qualquer resposta válida com no máximo $2 \cdot \boldsymbol{OPT}$ remoções seria aceite.

Input do Exemplo 1

```
2
5 2
1 2 3 7 10
4 13
4 1
5 5 5 5
10
```

Output do Exemplo 1

```
2
2 3
4
5 5 5 5
```

Organização

Alto Patrocínio

Com o Alto Patrocínio de Sua Excelência

O Presidente da República

Patrocinadores

Apoios