

Problema C - Jogo de Triângulos

Este é um **problema de interação**.

Ao contrário dos outros problemas em que deves fazer leitura de dados e escrita do output, neste problema deves interagir com o avaliador através da implementação de uma função e da interação com as funções fornecidas.

Parabéns, conseguiste chegar ao desafio final do jogo de televisão ONI (Onde Nada é Impossível), que até agora nunca foi vencido por nenhum concorrente! É agora a tua oportunidade!

Em frente ao apresentador do programa, são apresentadas N caixas numa linha, nas posições $1, 2, \dots, N$. Não consegues ver o interior das caixas, porém, é-te informado que dentro de cada caixa há um papel com um número entre 1 e N escrito, tais que todos esses números são distintos. Formalmente, se P_i é o número na i -ésima caixa a contar da esquerda, sabemos que $P_i \neq P_j$ se $i \neq j$, ou seja, os números dentro das caixas formam uma permutação.

O apresentador informa-te as regras do jogo: Tens um número ilimitado de perguntas. Em cada pergunta poderás selecionar um conjunto S de números entre 1 e N , e como resposta o apresentador providencia o número de ternos a, b, c tais que $a < b < c$ e P_a, P_b e P_c formam os lados de um triângulo com área positiva. Formalmente, isto é equivalente a $P_a < P_b + P_c$, $P_b < P_c + P_a$ e $P_c < P_a + P_b$.

Alguns exemplos de ternos que não formam lados de um triângulo são $\{2, 3, 5\}$ e $\{3, 4, 8\}$, uma vez que 5 não é menor que $2 + 3$ e 8 não é menor que $3 + 4$.

Porém, ao questionares sobre um conjunto S , deverás pagar $|S|^2$ euros aos organizadores do programa, ou seja, o quadrado do tamanho do conjunto. O apresentador também te informa de que na caixa que está numerada com 1 (ou seja a x -ésima caixa a contar da esquerda onde x é tal que $P_x = 1$) está localizado um diamante no valor de Q euros, que será teu se acertares a sua localização! Além disso, se conseguires acertar todos os números que estão dentro das caixas, para além do diamante serás premiado com um dos objetos mais raros do universo: uma máquina que consegue determinar se um programa para ou não!

T vai representar o teu objetivo neste jogo. Com $T = 1$, o teu objetivo será apenas o diamante, porém, com $T = 2$ terás de acertar nos números de todas as caixas. Como o Diamante vale Q euros, decidiste que nunca podes gastar mais do que essa quantidade no jogo. Fascinado com as



tuas possibilidades, conseguirás vencer a grande final?

Ficheiros para Download

Podes começar por descarregar os ficheiros correspondentes à tua linguagem (ou um [arquivo zip](#) contendo tudo):

Linguagem	Ficheiro a implementar	Avaliador	Outros Ficheiros	Input exemplo
C++	resolver.cpp	avaliador.cpp	avaliador.h	input.txt

Nota que a implementação do avaliador a usar nos testes oficiais será diferente.

Implementação

Deves submeter um único ficheiro que implementa uma função:

- A função `find(N, Q)`, que recebe um inteiro N , que representa o número de caixas, e um inteiro Q , que representa o preço do diamante.

Para isso deves usar o ficheiro `resolver.cpp` que descarregaste, colocando no interior das funções o teu código. Podes acrescentar outras funções, mas devem ficar todas neste ficheiro que é o único que deves submeter.

Funções a implementar:

C++: `void find(int N, int Q)`

A tua função deve invocar as seguintes funções:

- A função `check(V)`, que recebe um vetor de inteiros V e devolve a resposta do apresentador ao conjunto V . Nota que ao uso desta função pagas $|V|^2$ euros.
- A função `resposta(P)`, que recebe um vetor P , que corresponde à tentativa de adivinhar a numeração das caixas. Esta função deve ser invocada no final do teu programa.

Nota o seguinte:

- Os valores de V devem ser entre 1 a N , se alguma chamada não respeitar estas condições, o vosso código terá o resultado de **Wrong Answer**.
- Os valores de P devem ser uma permutação entre 1 a N , se alguma chamada não respeitar estas condições, o vosso código terá o resultado de **Wrong Answer**.
- Se o preço total das questões exceder Q euros, o vosso código terá o resultado de **Wrong Answer**.

Funções do avaliador:

C++: `int check(vector<int> V)`

C++: `void resposta(vector<int> P)`

A vossa função não deve ler nem escrever para os canais de entrada/saída padrão.

Exemplo

Se tivermos $N = 8$, $P = [3, 5, 6, 2, 4, 1, 7, 8]$:

Invocação	Resultado	Descrição
–	–	Início
<code>check({1, 2, 3, 6})</code>	1	Apenas $a = 1, b = 2, c = 3$ forma triângulo
<code>check({4, 5, 6})</code>	0	Não há triângulos
<code>check({1, 2, 3, 5})</code>	4	Todos os ternos formam triângulos
<code>resposta({3, 5, 6, 2, 4, 1, 7, 8})</code>	–	Resposta correta

Nota que, se a última invocação tivesse sido `resposta({2, 3, 4, 5, 6, 1, 8, 7})`, o resultado seria aceite quando $T = 1$, uma vez que acertamos na posição com 1, porém quando $T = 2$ o resultado não seria aceite uma vez que não acertamos em todas as posições.

Restrições

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

$1 \leq T \leq 2$ Tipo de problema
 $8 \leq N \leq 10^4$ Número de caixas
 $Q = 15 \times 10^6$ Valor do diamante

Sumário de subtarefas

Os casos de teste do problema estão organizados em cinco grupos com restrições adicionais diferentes:

Grupo	Número de Pontos	Restrições adicionais
1	10	$N \leq 50$ e $T = 1$
2	25	$N \leq 500$ e $T = 1$
3	25	$T = 1$
4	20	$N \leq 500$ e $T = 2$
5	20	$T = 2$

Testes no vosso computador

É disponibilizado um avaliador exemplo em cada linguagem (`avaliador.cpp`) que pode ser utilizado para testar a vossa submissão. Está ainda disponível um ficheiro auxiliar (`avaliador.h`). **Este avaliador não corresponde ao utilizado pelo sistema de avaliação.**

Este avaliador começa por receber como input um inteiro N , um inteiro Q e um inteiro T , correspondendo, respetivamente, ao número de caixas, ao preço do Diamante e ao tipo de problema. Segue-se uma linha com N inteiros distintos entre 1 e N , representando os valores de P_i .

O avaliador irá automaticamente invocar a função `find(N,Q)`, por vocês implementada. O avaliador indicará se a resposta é considerada correta ou não quando chamarem a função `resposta`.

Disponibilizamos um ficheiro de teste:

- `input.txt` que contém o caso de exemplo referido acima.

Um exemplo de teste na tua máquina (supondo que tens os compiladores oficiais instalados) seria o seguinte:

Linguagem	Compilar	Executar com o exemplo
C++	<code>g++ -Wall -std=gnu++14 -O2 avaliador.cpp resolver.cpp</code>	<code>./a.out < input.txt</code>

Organização



Alto Patrocínio

Com o Alto Patrocínio de Sua Excelência



O Presidente da República



REPÚBLICA PORTUGUESA

GABINETE DO MINISTRO DA EDUCAÇÃO

Patrocinadores



Fundação para a Ciência e a Tecnologia



FUNDAÇÃO CALOUSTE GULBENKIAN



Apoios

