



OBI2008

Caderno de Tarefas

Modalidade Programação • Seletiva IOI, Teste 2

A PROVA TEM DURAÇÃO DE DUAS HORAS

LEIA ATENTAMENTE ESTAS INSTRUÇÕES ANTES DE INICIAR A PROVA

- Este caderno de tarefas é composto por 2 páginas (não contando esta folha de rosto), numeradas de 1 a 2. Verifique se o caderno está completo.
- A prova deve ser feita individualmente.
- É proibido consultar a Internet, livros, anotações ou qualquer outro material durante a prova. É permitida a consulta ao *help* do ambiente de programação se este estiver disponível.
- As tarefas têm o mesmo valor na correção.
- A correção é automatizada, portanto siga atentamente as exigências da tarefa quanto ao formato da entrada e saída de seu programa.
- Não implemente nenhum recurso gráfico nas suas soluções (janelas, menus, etc.), nem utilize qualquer rotina para limpar a tela ou posicionar o cursor.
- As tarefas não estão ordenadas, neste caderno, por ordem de dificuldade; procure resolver primeiro as questões mais fáceis.
- Preste muita atenção no nome dos arquivos fonte indicados nas tarefas. Soluções na linguagem C devem ser arquivos com sufixo *.c*; soluções na linguagem C++ devem ser arquivos com sufixo *.cc* ou *.cpp*; soluções na linguagem Pascal devem ser arquivos com sufixo *.pas*. Para problemas diferentes você pode escolher trabalhar com linguagens diferentes, mas apenas uma solução, em uma única linguagem, deve ser submetida para cada problema.
- Ao final da prova, para cada solução que você queira submeter para correção, copie o arquivo fonte para o seu diretório de trabalho ou disquete, conforme especificado pelo seu professor.
- Não utilize arquivos para entrada ou saída. Todos os dados devem ser lidos da entrada padrão (normalmente é o teclado) e escritos na saída padrão (normalmente é a tela). Utilize as funções padrão para entrada e saída de dados:
 - em Pascal: *readln*, *read*, *writeln*, *write*;
 - em C: *scanf*, *getchar*, *printf*, *putchar*;
 - em C++: as mesmas de C ou os objetos *cout* e *cin*.
- Procure resolver o problema de maneira eficiente. Na correção, eficiência também será levada em conta. As soluções serão testadas com outras entradas além das apresentadas como exemplo nas tarefas.

Super-heróis

Nome do arquivo fonte: `heroi.c`, `heroi.cpp` ou `heroi.pas`

Jornalistas são um perigo para os super-heróis: seja porque acabam apaixonando-se, seja porque jornalistas têm a mania de querer descobrir suas identidades secretas.

Recentemente, um grupo de jornalistas uniu esforços para tentar determinar a identidade secreta de todos os super-heróis da cidade. Os jornalistas coletaram informações sobre pessoas e super-heróis presentes a eventos como salvamentos e prisões, de modo a ter pistas sobre a identidade dos super-heróis. Por exemplo, se em um determinado evento foi relatado que o Super-Homem salvou o Sr. Percival e a Sra. Rosaura, sabemos que nem o Sr. Percival nem a Sra. Rosaura podem ser a identidade secreta do Super-Homem.

Ser um super-herói é psicológica e fisicamente muito estressante, de forma que os jornalistas sabem que uma pessoa comum pode ser a identidade secreta de no máximo um super-herói. Além disso, os jornalistas já determinaram quais pessoas comuns são super-heróis, embora não saibam ainda a correspondência exata entre eles.

Dado um conjunto de informações coletadas pelos jornalistas, você deve escrever um programa que determine, para cada super-herói, quais podem ser suas identidades secretas.

Entrada

A entrada contém um único conjunto de testes, que deve ser lido do *dispositivo de entrada padrão* (normalmente o teclado). A primeira linha do conjunto de testes contém dois números inteiros N e E que indicam respectivamente o número de pessoas comuns e super-heróis ($2 \leq N \leq 100$) e o número de eventos relatados ($0 \leq E \leq 10000$).

A segunda linha contém N nomes de super-heróis, separados por um espaço em branco; a terceira linha contém N nomes de pessoas comuns, também separados por um espaço em branco. Cada uma das E linhas seguintes começa com um inteiro M ($2 \leq M \leq N$) indicando quantos super-heróis ou pessoas foram vistas juntas em um evento. A seguir, na mesma linha, há a lista dos M nomes, separados por um espaço em branco.

Cada nome, de super-heróis e de pessoas comuns, tem no máximo 20 caracteres e é composto somente por letras (maiúsculas e minúsculas), dígitos e hífen ('-'). Todos os nomes são distintos.

Saída

Seu programa deve imprimir, na *saída padrão*, uma linha para cada super-herói, contendo o nome do super-herói, seguido do caractere ':', seguido de um espaço em branco, seguido de uma lista de possíveis identidades secretas para esse super-herói; os nomes da lista devem ser separados por um espaço em branco. Tanto a ordem da lista de super-heróis quanto a ordem da lista de identidades secretas devem ser a mesma ordem dada na entrada.

Caso as informações dadas sejam inconsistentes, imprima apenas uma linha na saída, contendo a palavra 'IMPOSSIVEL' (sem acento).

Exemplo de entrada	Exemplo de saída
4 5 Superman Hulk Spider-Man Batman Bruce-Banner Peter-Parker Clark-Kent Bruce-Wayne 3 Bruce-Banner Bruce-Wayne Superman 4 Superman Hulk Bruce-Wayne Peter-Parker 3 Peter-Parker Hulk Batman 4 Clark-Kent Bruce-Banner Spider-Man Batman 4 Hulk Spider-Man Bruce-Wayne Clark-Kent	Superman: Clark-Kent Hulk: Bruce-Banner Spider-Man: Peter-Parker Batman: Bruce-Wayne

Exemplo de entrada	Exemplo de saída
4 2 Rogue Storm Gambit Wolverine James-Howlett Ororo-Monroe Remy-LeBeau Anna-Marie 4 Wolverine Gambit Ororo-Monroe Anna-Marie 3 Gambit Remy-LeBeau James-Howlett	IMPOSSIVEL

Exemplo de entrada	Exemplo de saída
2 2 Batman Robin Bruce-Wayne Tim-Drake 2 Robin Batman 2 Tim-Drake Bruce-Wayne	Batman: Bruce-Wayne Tim-Drake Robin: Bruce-Wayne Tim-Drake

Informações sobre a pontuação

- Para um subconjunto dos casos de teste totalizando 30 pontos, $2 \leq N \leq 10$ e $0 \leq E \leq 100$.
- Para um subconjunto dos casos de teste totalizando 55 pontos, $2 \leq N \leq 50$ e $0 \leq E \leq 2500$.