

SPOJ Brasil

Sair
luishpmdes

minha conta

status
submiter
problemas
search

classificação

notícias
comments
informações
regras
links

SPOJ
2014-08-28
01 : 06 : 02

Submeter

Minhas submissões

Todas submissões

Melhores

PS

PDF

Voltar



SPOJ Problem Set (obi)

0

11015. Escalonamento ótimo

g+1

Problema: ESCALO11

O SBC (*System for Batch Computing*) é um sistema operacional voltado para a execução sequencial de tarefas. O operador do sistema cria tarefas e o sistema operacional é responsável por agendar a execução destas tarefas.

Cada tarefa pode depender da conclusão de algumas tarefas para poder começar. Se uma tarefa A depende de uma tarefa B , a tarefa B deve terminar antes que a tarefa A inicie sua execução.

Além disto, cada tarefa possui uma prioridade. É sempre mais vantajoso para o sistema começar executando uma tarefa de mais alta prioridade, depois continuar executando uma tarefa de mais alta prioridade dentre as que sobraram e assim por diante.

Neste problema, será dado um inteiro N , que irá representar o número de tarefas no sistema. As tarefas serão numeradas de 0 até $N-1$. Tarefas com índice menor possuem prioridade maior, de forma que a tarefa 0 é a tarefa de mais alta prioridade, a tarefa 1 é a tarefa com a segunda maior prioridade e assim por diante, até a tarefa $N-1$, que é a tarefa com a menor prioridade. Além disso, serão dadas M relações de dependência entre as tarefas.

Seu objetivo será decidir se é possível executar as tarefas em alguma ordem. Caso seja possível, você deverá produzir uma ordem de execução ótima para as tarefas, isto é, desempate as ordens possíveis pela prioridade da primeira tarefa. Se o empate ainda persistir, desempate pela prioridade da segunda tarefa, e assim por diante.

Entrada

A primeira linha da entrada contém inteiros N e M . As próximas M linhas descrevem, cada uma, uma dependência entre as tarefas da entrada. Cada uma dessas linhas irá conter dois inteiros A e B que indicam que a tarefa B depende da tarefa A , isto é, que a tarefa A deve terminar antes que a tarefa B inicie.

Saída

Se não for possível ordenar as tarefas de forma que as dependências sejam satisfeitas, imprima uma única linha contendo o caracter "*". Caso contrário, imprima N linhas contendo cada uma um número inteiro. O inteiro na i -ésima linha deve ser o índice da i -ésima tarefa a ser executada na ordem ótima de execução das tarefas.

Restrições

- $0 \leq N \leq 50000$.
- $0 \leq M \leq 200000$.
- $0 \leq A, B < N$.

Exemplos

Entrada

```
3 1
2 0
```

Time for a new job?

Saída

```
1
2
0
```

Entrada

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

Saída

```
*
```

Adicionado por: Wanderley Guimarães
Data: 2012-03-10
Tempo limite: 1s-3s
Tamanho do fonte: 50000B
Memory limit: 256MB
Cluster: Pyramid (Intel Pentium III 733 MHz)
Linguagem permitida: Todas exceto: CLOJ ERL F# PERL 6 PYTH 3.2.3 n SCALA TCL
Origem: OBI 2011 - fase 2 nível 2

hide comments

2014-08-09 01:07:43 João Baptista de Paula e Silva

O mais interessante é que a diferença de tempo entre o stdio e o iostream é o suficiente para dar TLE.

Leave a Comment

Notes:

1. Don't post any source code here.
2. Please be careful, leave short comments only. Don't spam here.
3. For more discussion (hints, ideas, solutions) please visit our [forum](#).
4. Authors are allowed to delete the post and use html code here (e.g. to provide some useful links).

Visualização: [800x600](#) [1024x768](#) [Tela cheia](#) Tema: [olive](#) [banana](#) [plum](#)

© Spoj.com. All Rights Reserved. Spoj uses [Sphere Engine™](#) © by [Sphere Research Labs](#)