

## Jarní úklid

<b>Termín odevzdání:</b>	<b>02.05.2014 23:59:59</b>
<b>Hodnocení:</b>	<b>1.0000</b>
<b>Max. hodnocení:</b>	<b>1.0000</b> (bez bonusů)
<b>Odevzdaná řešení:</b>	2 / 30 Volné pokusy + 20 Penalizované pokusy (-0 % penalizace za každé odevzdání)
<b>Nápovědy:</b>	0 / 30

Uklízeč firma Fiala-Ivan-Tříška se chystá na jarní úklid celého města. Bohužel předchozí sezóna a neblahá finanční situace se dost podepsala na jejich vozovém parku a zůstalo jim jen jediné staré křehké vozidlo, které takový úklid zvládne, než se také rozpadne. Je tedy třeba vozidlo maximálně šetřit a chovat se k němu s láskou.

Město je soustava ulic a křižovatek, každá křižovatka je spojena s další křižovatkou nejvýše jednou ulicí. Vozidlo může projíždět ulice libovolným směrem, na vyčištění mu stačí jeden průjezd.

Úkolem je rozvrhnout trasu městem tak, aby vozidlo vyčistilo každou ulici města a zároveň neprojíždělo žádnou ulicí vícekrát. Úklid končí ve chvíli, kdy je vyčištěna poslední ulice v městě. Výstupem programu je takové pořadí křižovatek, v jakém je má vozidlo projíždět tak, aby projelo každou ulicí právě jednou. Pokud taková cesta není pro zadané město možná, hledání skončí neúspěchem. Pozor na to, že vozidlo nemusí začít a skončit ve stejné křižovatce.

Město řídí nezodpovědný starosta Petr Furtprší, který občas uzavře některé ulice kvůli pořádání různých akcí a tak není zaručena dostupnost všech částí města.

Řešení realizujte formou funkce **trasa** s rozhraním níže. Funkce má parametrem dvě jména souborů: zdrojový a cílový. Funkce čte zdrojový soubor a zapisuje výslednou trasu do cílového souboru. Návratovou hodnotou funkce je TRUE V případě úspěšného nalezení trasy městem a FALSE V případě nenalezení trasy městem.

### Formát vstupu

Křižovatky jsou očíslovány přirozenými čísly od **0** do **N-1**. Na vstupu dostanete na prvním řádku počet křižovatek a počet ulic oddělený mezerou, následovaný (na dalších řádcích) seznamem dvojic křižovatek, mezi kterými vede ulice. Každý řádek seznamu obsahuje 2 čísla oddělená mezerou.

```
N M
(kde N je počet křižovatek a M je počet ulic)
DVOJICE
(dvojice indexů křižovatek, mezi kterými je ulice. Počet řádků s dvojicemi je M)
```

Poznámka: Můžete se spolehnout na to, že vstup je zadán korektně

### Formát výstupu

Trasa, po které má vozidlo jet ve formátu SEZNAM, kde první řádek obsahuje počet prvků výsledného seznamu. Každý další řádek obsahuje právě jedno číslo křižovatky. Pokud taková trasa nelze vytvořit, návratová hodnota funkce je FALSE.

```
N
(kde N je počet křižovatek trasy)
T
(indexy křižovatek, kudy vede trasa)
```

Poznámka: limity - maximálně 20 000 křižovatek, 65 000 cest

### Příklad vstupu a výstupu

**Vstup:**

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

**Výstup:**

```
6
```

3  
1  
0  
3  
2  
1

Při implementaci máte k dispozici datové struktury z STL, viz ukázka. Struktury z STL použít můžete, ale nemusíte.

```
#ifndef __PROGTEST__
#include <cstdio>
#include <cstdlib>
#include <cstring>
#include <string>
#include <iostream>
#include <fstream>
#include <map>
#include <vector>
#include <algorithm>
#include <set>
#include <queue>
#include <list>
#include <stdint.h>

using namespace std;
#endif /* __PROGTEST__ */

bool trasa( const char * inFile, const char * outFile )
{
    // mandatory, todo
}

#ifndef __PROGTEST__
int main ( int argc, char * argv [] )
{
    // tests
    return 0;
}
#endif /* __PROGTEST__ */
```

2	11.04.2014 22:36:25	Download
<b>Stav odevzdání:</b> Ohodnoceno		
<b>Hodnocení:</b> 1.0000		
<ul style="list-style-type: none"><li>• <b>Hodnotitel: automat</b><ul style="list-style-type: none"><li>◦ Program zkompileován</li><li>◦ Test 'Zakladni test se soubory dle ukazky': Úspěch<ul style="list-style-type: none"><li>▪ Dosaženo: 100.00 %, požadováno: 100.00 %</li><li>▪ Celková doba běhu: 0.000 s (limit: 2.000 s)</li><li>▪ Využití paměti: 13632 KiB (limit: 29962 KiB)</li><li>▪ Úspěch v závazném testu, hodnocení: 100.00 %</li></ul></li><li>◦ Test 'Test na malých grafech': Úspěch<ul style="list-style-type: none"><li>▪ Dosaženo: 100.00 %, požadováno: 100.00 %</li><li>▪ Celková doba běhu: 0.011 s (limit: 2.000 s)</li><li>▪ Využití paměti: 13632 KiB (limit: 29962 KiB)</li><li>▪ Úspěch v závazném testu, hodnocení: 100.00 %</li></ul></li><li>◦ Test 'Test na velkých grafech': Úspěch<ul style="list-style-type: none"><li>▪ Dosaženo: 100.00 %, požadováno: 50.00 %</li><li>▪ Celková doba běhu: 0.934 s (limit: 1.989 s)</li><li>▪ Využití paměti: 21812 KiB (limit: 29962 KiB)</li><li>▪ Úspěch v závazném testu, hodnocení: 100.00 %</li></ul></li><li>◦ Celkové hodnocení: 100.00 % (= 1.00 * 1.00 * 1.00)</li></ul></li><li>• Celkové procentní hodnocení: 100.00 %</li><li>• Celkem bodů: 1.00 * 1.00 = 1.00</li></ul>		
<b>SW metriky:</b>		
Celkem      Průměr      Maximum      Jméno funkce		

Funkce:	11	--	--	--
Řádek kódu:	147	13.36 ± 21.42	80	trasa(const char *,const char *)
Cyklomatická složitost:	35	3.18 ± 4.59	17	trasa(const char *,const char *)

1

11.04.2014 22:33:51

Download

Stav odevzdání:

Ohodnoceno

Hodnocení:

0.0000

• Hodnotitel: automat

◦ Program zkompileován

◦ Test 'Zakladni test se soubory dle ukazky': Úspěch

▪ Dosaženo: 100.00 %, požadováno: 100.00 %

▪ Celková doba běhu: 0.000 s (limit: 2.000 s)

▪ Využití paměti: 13632 KiB (limit: 29962 KiB)

▪ Úspěch v závazném testu, hodnocení: 100.00 %

◦ Test 'Test na malých grafech': Neúspěch

▪ Dosaženo: 90.00 %, požadováno: 100.00 %

▪ Celková doba běhu: 0.012 s (limit: 2.000 s)

▪ Využití paměti: 13632 KiB (limit: 29962 KiB)

▪ Neúspěch v závazném testu, hodnocení: 0.00 %

▪ Nesprávný výstup [Zpřístupnit nápovědu (42 B)]

◦ Celkové hodnocení: 0.00 % (= 1.00 \* 0.00)

• Celkové procentní hodnocení: 0.00 %

• Celkem bodů: 0.00 \* 1.00 = 0.00

SW metriky:

Funkce:

11

--

-- --

Řádek kódu:

147

13.36 ± 21.42

80

trasa(const char \*,const char \*)

Cyklomatická složitost:

35

3.18 ± 4.59

17

trasa(const char \*,const char \*)