



A 2015/2016 tanévi Országos Középiskolai Tanulmányi Verseny döntő forduló javítási-értékelési útmutató

INFORMATIKA II. (programozás) kategória

1. feladat: Épület (30 pont)

Egy N szintes épület szintjeit fehér (F), piros (P) és zöld (Z) színnel festhetjük ki. Fehér emeletet csak piros emelet követhet, zöld emeletet pedig nem követhet piros! Minden színezés egy N karakterből álló szöveggel adható meg.

Készíts programot, amely megadja, hogy az ábécé sorrendben i -edik színezést!

A *standard bemenet* első sorában az emeletek száma ($1 \leq N \leq 50$) és az I sorszám ($1 \leq I \leq 1\,000\,000\,000\,000\,000$) van. Feltehető, hogy az N emeletes házban van I sorszámú színezés.

A *standard kimenet* egyetlen sorába az I . színezést leíró N karakteres szöveget kell kiírni!

Példa:

bemenet	kimenet
3 7	PPZ

Magyarázat, az összes színezés: FPF, FPP, FPZ, PFP, PPF, PPP, PPZ, PZF, PZZ, ZFP, ZZF, ZZZ

Korlátok:

Időlimit: 0.5 mp

Memórialimit: 32MiB

A tesztek 60%-ában $N < 10$.

Értékelés:

1. $N=1, I=1 \rightarrow F$	1 pont
2. $N=1, I=2 \rightarrow P$	1 pont
3. $N=1, I=3 \rightarrow Z$	1 pont
4. $N=2, I=3 \rightarrow PZ$	1 pont
5. $N=2, I=5 \rightarrow ZF$	1 pont
6. $N=3, I=1 \rightarrow FPF$	1 pont
7. $N=3, I=5 \rightarrow PPF$	1 pont
8. $N=3, I=8 \rightarrow PZF$	1 pont
9. $N=3, I=10 \rightarrow ZZZ$	1 pont
10. $N=5, I=10 \rightarrow FPZFP$	1 pont
11. $N=5, I=30 \rightarrow PPZZZ$	1 pont
12. $N=10, I=30 \rightarrow FPFPPFPPPP$	2 pont

13. $N=10, I= 100 \rightarrow$ FPPFPFPPFP 2 pont
 14. $N=10, I= 1000 \rightarrow$ PZFPPPFPPZ 2 pont
 15. $N=50, I= 1\ 000\ 000 \rightarrow$ 3 pont
 16. $N=50, I= 1\ 000\ 000\ 000 \rightarrow$ 3 pont
 17. $N=50, I= 1\ 000\ 000\ 000\ 000 \rightarrow$ 3 pont
 18. $N=50, I= 1\ 000\ 000\ 000\ 000\ 000 \rightarrow$ 4 pont

2. feladat: Veszélyes városok (30 pont)

Ádám és Éva különböző városokban lakik. Ádám el szeretne utazni Évához. Vannak olyan városok, ahova veszélyes belépni. Ádám és Éva városa biztos nem veszélyes.

Készíts programot, amely megadja, hogy Ádámnak minimálisan hány veszélyes várost kell érinteni, és az ilyen útvonalak közül minimálisan hány közbülső városon kell áthaladnia!

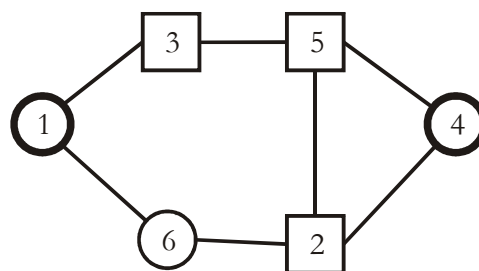
A *standard bemenet* első sorában a városok száma ($1 \leq N \leq 10\ 000$), a közöttük levő közvetlen utak száma ($1 \leq M \leq 1\ 000\ 000$), valamint Ádám és Éva városának sorszáma ($1 \leq A, E \leq N$) van. A következő M sor tartalmazza a közvetlen utak két végpontja sorszámát ($1 \leq U_i \neq V_i \leq N$). Az utolsó sorban a veszélyes városok száma ($0 \leq TV < N-1$), majd a TV darab veszélyes város sorszáma ($1 \leq TVS_i \leq N$) található.

A *standard kimenet* első sorába a minimálisan érintendő veszélyes városok számát kell írni, a második sorba pedig az ennyi veszélyes várost érintő utakon levő városok számát (nem beleértve Ádám és Éva városát)! A második sorba egy, a feltételnek megfelelő útvonalat kell írni, az első szám Ádám városa, az utolsó pedig Éva városa legyen! Több megoldás esetén bármelyik megadható.

Példa:

bemenet
 6 7 1 4
 1 3
 5 3
 5 2
 2 4
 4 5
 1 6
 6 2
 3 5 2 3

kimenet
 1
 2
 1 6 2 4



Korlátok:

Időlimit: 0.1 mp.

Memórialimit: 32 MiB

Pontozás: A tesztek 40%-ában a $N \leq 100$

Értékelés:

1. Egyetlen út van 1+1 pont
 2. Egyetlen út van, csupa veszélyes várossal 1+1 pont
 3. Nincs veszélyes város a legrövidebb úton 1+1 pont
 4. A legrövidebb úton van veszélyes város, de a leghosszabbon nincs 1+1 pont
 5. Minden úton van egy veszélyes város 1+1 pont
 6. Egyes utakon különböző számú veszélyes város van 1+1 pont

- | | |
|--|----------|
| 7. Véletlen közepes teszt (sok város, kevés út, sok veszélyes város) | 1+1 pont |
| 8. Véletlen közepes teszt (sok város, sok út, sok veszélyes város) | 1+1 pont |
| 9. Véletlen közepes teszt (sok város, kevés út, kevés veszélyes város) | 1+1 pont |
| 10. Véletlen közepes teszt (sok város, sok út, kevés veszélyes város) | 1+1 pont |
| 11. Véletlen nagy teszt (sok város, kevés út, sok veszélyes város) | 1+1 pont |
| 12. Véletlen nagy teszt (sok város, sok út, sok veszélyes város) | 1+1 pont |
| 13. Véletlen nagy teszt (sok város, kevés út, kevés veszélyes város) | 1+2 pont |
| 14. Véletlen nagy teszt (sok város, sok út, kevés veszélyes város) | 1+2 pont |

3. feladat: Repülők indítása (30 pont)

Egy repülőtérről N repülőjárat indul. A repülőtársaságok mindegyikre megadtak egy lehetséges indítási idő intervallumot. Az i -edik járat legkorábban az A_i és legkésőbb a B_i időpontban indítható. Ezekon belül az egyes gépek tetszőleges időpontban indíthatók, de a sorrendjük nem változtatható meg. Egy indítási terv a járatok indítási idejének a T_1, \dots, T_N sorozata. Azt az indítási tervet nevezzük legbiztonságosabbnak, amikor az egymás utáni indítások közötti legkisebb idő a lehető legnagyobb.

Készíts programot, amely megadja a legbiztonságosabb indítási tervben az egymás utáni indítások közötti legkisebb időt!

A *standard bemenet* első sorában a repülőjáratok száma van ($2 \leq N \leq 10\,000$). A következő N az egyes repülőgépek legkorábbi és legkésőbbi lehetséges indítása időpontjai szerepelnek ($0 \leq A_i < B_i \leq 10\,000\,000$), indulási idő szerinti sorrendben.

A *standard kimenet* egyetlen sorába a legbiztonságosabb indítási tervben az egymás utáni indítások közötti legkisebb időt kell kiírni! A második sorba egy ilyen legbiztonságosabb indítási tervet kell írni! Több megoldás esetén bármelyik megadható.

Példa:

bemenet

4
0 5
1 4
5 15
5 10

kimenet

3
0 3 7 10



Korlátok:

Időlimit: 0.1 mp.

Memórialimit: 32 MiB

Pontozás: A tesztek 60%-ában a $N \leq 100$

Értékelés:

- | | |
|---|----------|
| 1. Két repülőjárat van | 1+2 pont |
| 2. Mindenki a kezdőidőpontján indul (az utolsót kivéve) | 1+2 pont |
| 3. Az elsőt kivéve mindenki a végidőpontján indul | 1+2 pont |
| 4. Felváltva kezdő-, illetve végidőpontokon indulnak | 1+2 pont |
| 5. A két szélső kivételével nem kezdő- és nem végidőpontokon indulnak | 1+2 pont |
| 6. Van olyan pár, amiket nem lehet messzebbre tenni | 1+2 pont |

- | | |
|---------------------------|----------|
| 7. Véletlen közepes teszt | 1+2 pont |
| 8. Véletlen közepes teszt | 1+2 pont |
| 9. Véletlen nagy teszt | 1+2 pont |
| 10. Véletlen nagy teszt | 1+2 pont |

4. feladat: Építkezés (30 pont)

Egy építkezés terve N munka elvégzését írja elő. Minden munkát pontosan egy nap alatt lehet elvégezni. A terv tartalmaz megelőzési előírásokat $U V$ pár formájában, ami azt jelenti, hogy a V munka csak akkor kezdhető meg, ha az U munkát már elvégezték. Egy vállalkozó nyerte a versenykiírást az építkezés megvalósítására. A vállalkozónak K alkalmazottja van, akik mindegyike egy nap alatt tud egy munkát elvégezni. Tehát a vállalkozó egy nap alatt legfeljebb K munkát tud elvégezni.

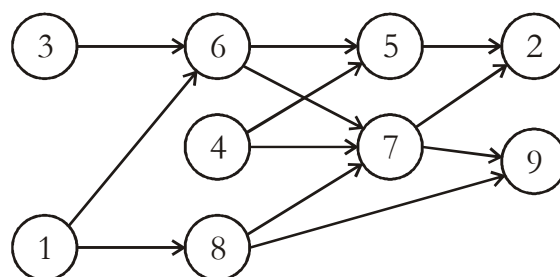
Készíts programot, amely kiszámítja, hogy legkevesebb hány nap alatt valósítható meg az építkezés!

A *standard bemenet* első sorában a munkák száma ($1 \leq N \leq 100\,000$) és az alkalmazottak száma ($1 \leq K \leq N$) és a megelőzési előírások ($1 \leq M \leq 100\,000$) száma van. A további M sor mindegyike egy számpárt tartalmaz ($1 \leq U \neq V \leq N$), ami egy megelőzési előírás.

A *standard kimenet* első sorába az építkezés elvégzéséhez szükséges legkevesebb napok L számát kell írni! A további L sor mindegyike egy-egy napon elvégzendő munkák azonosítóit tartalmazza tetszőleges sorrendben! A sorban a felsorolást a 0 szám zárja! Több megoldás esetén bármelyik megadható.

Példa:

bemenet	kimenet
9 3 12	4
1 6	3 1 4 0
3 6	8 6 0
4 5	7 5 0
5 2	9 2 0
4 7	
6 5	
6 7	
7 9	
8 7	
8 9	
1 8	
7 2	



Korlátok

Időlimit: 0.1 mp.

Memórialimit: 32 MiB

Pontozás: A tesztek 30%-ában a $N \leq 1000$

Értékelés:

- | | |
|-----------------------------|----------|
| 1. Kis bemenet | 1+1 pont |
| 2. Kicsi véletlen | 1+1 pont |
| 3. Kicsi speciális véletlen | 1+1 pont |

4. Közepes bemenet	1+1 pont
5. Véletlen speciális közepes teszt	1+1 pont
6. Véletlen speciális közepes teszt	1+1 pont
7. Véletlen speciális közepes teszt	1+2 pont
8. Véletlen közepes teszt	1+2 pont
9. Véletlen speciális teszt	1+2 pont
10. Véletlen speciális nagy teszt	1+2 pont
11. Véletlen nagy teszt	1+2 pont
12. Véletlen nagy teszt	1+2 pont

5. feladat: Konvex négyszög (30 pont)

Adott a síkon egy P ponthalmaz.

Készíts programot, amely megadja a ponthalmaz olyan négy a, b, c és d pontját, amelyek konvex négyszöget alkotnak!

A *standard bemenet* első sorában a ponthalmaz pontjainak száma ($7 \leq N \leq 100\,000$) van. A további N sor mindegyike egy-egy pont x és y koordinátáját ($-1\,000\,000 \leq x, y \leq 1\,000\,000$) tartalmazza.

A *standard kimenet* első és egyetlen sorába négy olyan pont sorszámát kell írni, amelyek ebben a sorrendben órajárással ellentétes irány szerint konvex négyszöget alkotnak! Ha nincs megoldás, akkor négy 0 számot kell kiírni! Több megoldás esetén bármelyik megadható.

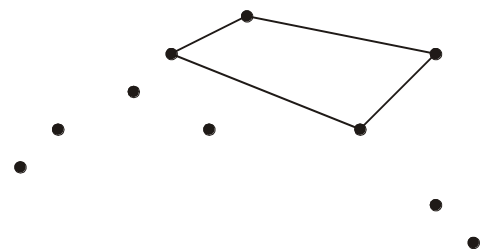
Példa:

bemenet

```
10
4 7
1 5
2 6
5 8
6 6
7 9
12 8
10 6
12 4
13 3
```

kimenet

```
4 8 7 6
```



Korlátok

Időlimit: 0.1 mp.

Memórialimit: 32 MiB

Pontozás: A tesztek 30%-ában a $N \leq 1000$

Értékelés:

1. Kis bemenet	1+1 pont
2. Kicsi véletlen	1+1 pont
3. Kicsi speciális véletlen	1+1 pont
4. Közepes bemenet	1+1 pont

5. Véletlen speciális közepes teszt	1+1 pont
6. Véletlen speciális közepes teszt	1+1 pont
7. Véletlen speciális közepes teszt	1+1 pont
8. Véletlen közepes teszt	1+1 pont
9. Véletlen nagy teszt	1+1 pont
10. Véletlen speciális nagy teszt	1+1 pont
11. Véletlen speciális nagy teszt	1+1 pont
12. Véletlen speciális nagy teszt	1+1 pont
13. Véletlen nagy teszt	1+1 pont
14. Véletlen nagy teszt	1+1 pont
15. Véletlen nagy teszt	1+1 pont

Elérhető összpontszám: 150 pont + 50 pont a 2. fordulóból