

9. feladat: Hálózat (50 pont)

Bergengóciában N város között kell kiépíteni a hálózatot. K héten keresztül kapunk egy-egy újabb árajánlatot két város közötti közvetlen hálózati kapcsolat kiépítésére.

Írj programot (HALOZAT.PAS, HALOZAT.C, ...), amely minden hétre –ha lehetséges– megad az addig beérkezett javaslatok alapján egy tervet, hogy mely város-párok között építsenek ki közvetlen hálózati kapcsolatot, hogy bármely városból bármely másik elérhető legyen közvetlenül vagy más városokon keresztül, de úgy, hogy az építés összköltsége minimális legyen! A terv összekötendő város-párokból és a hálózat kiépítés teljes költségéből áll.

A HALOZAT.BE szöveges állomány első sorában A városok ($1 \leq N \leq 100$) és a hetek ($1 \leq K \leq 10000$) száma van. A következő K sor mindegyike egy hét árajánlatát tartalmazza: két város sorszámot ($1 \leq A \neq B \leq N$) és egy árat ($1 \leq AR \leq 1000$), egy-egy szóközzel elválasztva.

A HALOZAT.KI szöveges állományba pontosan $K+1$ sort kell írni. Az első sorba annak a hétnek a sorszámát, ami után már nem csökken tovább a minimális kiépítési költség. Ha a hálózat soha nem építhető ki, akkor ebbe a sorba 0 kerüljön! Az $i+1$ -edik sorba pedig, ha az i -edik héten még nem lehet olyan kapcsolatrendszer kiépíteni, hogy bármelyik város bárholnan elérhető legyen, akkor 0 kerüljön, egyébként pedig az i -edik heti kiépítés minimális költsége!

Példa:

HALOZAT.BE	HALOZAT.KI
3 5	4
1 3 100	0
2 3 50	150
1 3 60	110
1 2 40	90
1 3 120	90

10. feladat: Dominó (50 pont)

Tekintsük a dominó játéknak a következő kétszemélyes változatát. Először kiraknak egy sorban páros számú dominót, ez a választható dominósor, majd választanak egy kezdő dominót, amit külön helyre leraknak. A két játékos felváltva lép, az első játékos kezd. Minden lépésben az aktuális játékos választ egy dominót a választható dominósor valamelyik végéről és azt a kirakott dominósor valamelyik végéhez illeszti. Az illesztés azt jelenti, hogy a választott dominónak egyik oldalán ugyanannyi pöttynek kell lennie, mint amihez illeszteni akarja. Ha nem tud illeszkedőt választani, akkor le kell vennie a választható dominósor valamelyik szélső dominóját, és félre kell raknia, de ekkor egy büntetőpontot kap. A játék akkor ér véget, ha elfogyott az összes választható dominó. A játékban az nyer, akinek kevesebb büntetőpontja lesz.

Írj programot (DOMINO.PAS, DOMINO.C, ...), amely az első játékos játékát valósítja meg és nyer!

A bemenetekre teljesül, hogy az első játékosnak van nyerő stratégiája, azaz meg tudja venni a második játékost.

KÖNYVTÁR

A játék használatát a **jatek** könyvtár öt művelete biztosítja:

- **Meret**, egyszer kell hívni a program elején, a dominók N számát adja.
- **D1**, **D2**, egy dominó sorszámát kell argumentumként megadni, a visszaadott érték a dominó egyik, illetve másik oldalán lévő pöttyök száma. **D1(0)** és **D2(0)** a kezdő dominót adja, amelynek bal oldalán $D1(0)$, jobb oldalán $D2(0)$ pötty van.
- **EnLep**, Ezzel a művelettel kell közölni az első játékos lépését. Egy argumentuma van, a lépés kódja. A kódok a következők:
 - **1**: A választható dominósor bal szélső dominóját választjuk és azt a kirakott dominósor bal végéhez illesztjük. A választott dominó valamelyik oldalának illeszkednie kell a kirakott dominósor bal végéhez.
 - **2**: A választható dominósor bal szélső dominóját választjuk és azt a kirakott dominósor jobb végéhez illesztjük. A választott dominó valamelyik oldalának illeszkednie kell a kirakott dominósor jobb végéhez.
 - **3**: A választható dominósor jobb szélső dominóját választjuk és azt a kirakott dominósor bal végéhez illesztjük. A választott dominó valamelyik oldalának illeszkednie kell a kirakott dominósor bal végéhez.
 - **4**: A választható dominósor jobb szélső dominóját választjuk és azt a kirakott dominósor jobb végéhez illesztjük. A választott dominó valamelyik oldalának illeszkednie kell a kirakott dominósor jobb végéhez.
 - **5**: Ez a paraméter csak akkor adható meg, ha a választható dominósor egyik végén sincs olyan dominó, amely illeszkedne a kirakott sor valamelyik végéhez. Ekkor a művelet hatása az, hogy eltávolítja a választható dominósor bal szélső dominóját és 1 büntetőpontot ad a műveletet végrehajtó játékosnak.
 - **6**: Ez a paraméter csak akkor adható meg, ha a választható dominósor egyik végén sincs olyan dominó, amely illeszkedne a kirakott sor valamelyik végéhez. Ekkor a művelet hatása az, hogy eltávolítja a választható dominósor jobb szélső dominóját és 1 büntetőpontot ad a műveletet végrehajtó játékosnak.
- **TeLep**, A második játékos lépésének a kódját adja. Ezzel kérdezhető le a második játékos lépése. A második játékos utolsó lépésének a lekérdezése eredményezi a program befejezését, ezért csak az a szabályos játék, amely a második játékos utolsó lépését is lekérdezi.

A jatek könyvtár két szöveges állományt készít: domino.ki és domino.lg. A domino.ki első és egyetlen sora az első, illetve a második játékosnak a játék során szerzett büntetőpontjainak száma tartalmazza. A program és a könyvtár közötti dialógust a domino.lg tartalmazza.

Utasítások Pascal programozóknak: A

```
uses jatek;
```

import utasítás szerepeljen a program első sorában.

Utasítások C/C++ programozóknak: A

```
#include "jatek.h"
```

import utasítás szerepeljen a program első sorában. Készíts egy domino.prj projekt állományt a feladatkönyvtárban és add hozzá a projekthez a domino.c (domino.cpp) és jatek.obj állományokat és compile/make paranccsal végezd a fordítást.

HASZNÁLAT

Készíteni kell egy domino.in szöveges állományt, amelynek első sorában legyen a dominók N száma, ami páros legyen. A második sorban a kezdő dominót megadó két, 0 és 9 közötti egész szám legyen. A további N sorban kell megadni a kezdeti játékállásban szereplő dominókat, soronként egyet, abban a sorrendben, ahogy a táblán vannak.

PÉLDA

DOMINO.IN

```
6
1 2
1 2
3 1
3 4
4 5
2 3
```

FELTÉTELEK

- A megoldás program nem olvashat és nem írhat semmilyen állományt!
- Szabálytalan dialógus a program megszakítását eredményezi.
- A dominók N számára: $2 \leq N \leq 16$
- A dominók sorszámaira: $0 \leq i \leq N$. A kezdő dominó sorszáma 0
- Pascal könyvtárnév: jatek.tpu
- Pascal deklarációk:


```
function Meret: integer;
function D1(i: integer):integer;
function D2(i: integer):integer;
procedure EnLep(i: integer);
function TeLep : integer);
```
- C/C++ könyvtárnevek: jatek.h, jatek.obj
- C/C++ deklarációk:


```
int Meret(void);
int D1(int i);
int D2(int i);
```

```
void EnLep(int i);  
int TeLep(void);
```