

1. feladat: Ház (20 pont)

Egy ház tervrajzát egységnyi négyzetrácsos lapon készítik. Minden szobának téglalap alakúnak kell lenni. Eddig  $N$  szobát rajzoltak fel a tervrajzon. Minden szobát a bal felső és jobb alsó sarkával adnak meg. A négyzetrács egy mezőjét  $x$ -és  $y$ -koordinátájával adják meg, a bal felső mező koordinátái  $(0,0)$ . Az  $x$ -koordináták a vízszintesen, az  $y$ -koordináták függőlegesen nőnek. A tervező ki akarja számítani, hogy hány új téglalap alakú szobát lehet még betenni a tervbe, ha bármely két új szoba bármely két oldalának nem lehet közös része, továbbá mind a négy oldala szomszédos vagy meglévő szobával, vagy a ház oldalával. Eddig betervezett szobák olyanok, hogy minden szabadon maradt terület téglalap alakú.

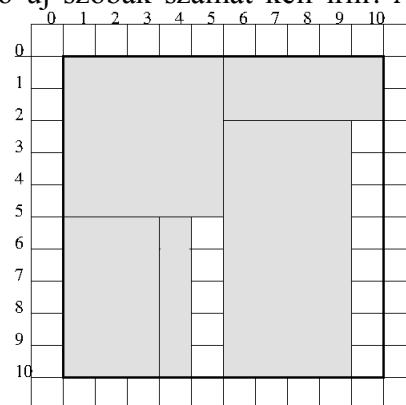
Készíts programot (`haz.pas`, `haz.c`, ...), amely az épület és a tervben meglévő szobák ismeretében megadja, hogy hány téglalap alakú új szobát lehet még betenni a tervbe, valamint mekkora a legnagyobb lehetséges új szoba területe!

A `haz.be` állomány első sorában a tervben meglévő szobák száma ( $1 \leq N \leq 1\,000\,000$ ), valamint az ház bal felső  $(FX,FY)$  és jobb alsó  $(AX,AY)$  sarkának koordinátái vannak ( $0 \leq FX < AX \leq 10\,000$ ,  $0 \leq FY < AY \leq 10\,000$ ), egy-egy szóközzel elválasztva. A következő  $N$  sor mindegyikében egy-egy szoba bal felső  $(BFX_i,BFY_i)$  és jobb alsó  $(JAX_i,JAY_i)$  sarkának koordinátái vannak ( $FX \leq BFX_i < JAX_i \leq AX$ ,  $FY \leq BFY_i < JAY_i \leq AY$ ) egy-egy szóközzel elválasztva.

A `haz.ki` szöveges állomány első sorába a kialakítható új szobák számát kell írni! A második sorba a legnagyobb új szoba területe kerüljön!

Példa:

<code>haz.be</code>	<code>haz.ki</code>
5 1 1 10 10	2
1 1 5 5	8
6 1 10 2	
6 3 9 10	
1 6 3 10	
4 6 4 10	



Értékelés:

- |                                       |          |
|---------------------------------------|----------|
| Nincs szabad terület                  | 1+0 pont |
| Egyetlen szabad téglalap van, belül   | 1+1 pont |
| Van szabad téglalap a bal oldalon is  | 1+1 pont |
| Van szabad téglalap a jobb oldalon is | 1+1 pont |
| Van szabad téglalap alul is           | 1+1 pont |
| Van szabad téglalap felül is          | 1+1 pont |
| Véletlen közepes teszt                | 1+1 pont |
| Véletlen közepes teszt                | 1+1 pont |
| Véletlen nagy teszt                   | 1+1 pont |
| Véletlen nagy teszt                   | 1+2 pont |

2. feladat: Csatorna (30 pont)

Egy szennyvíz csatorna hálózathoz takarító robotot fejlesztettek. A hálózat csomópontokból és közöttük levő kör keresztmetszetű csatorna szakaszokból áll, amelyeknek ismerjük a csőátmérőjét. A robot olyan csövet tud tisztítani, amelynek átmérője nagyobb a robot méreténél. A robot a csatornában mindkét irányban haladhat.

Készíts programot (`csatorna.pas`, `csatorna.c`, ...), amely megadja, hogy adott pontból indítva a robot hány csatorna szakaszt tud kitisztítani, valamint hogy minimum hány további pontból kellene elindítani, hogy az összes olyan csatornát kitisztítsa, ahova befér!

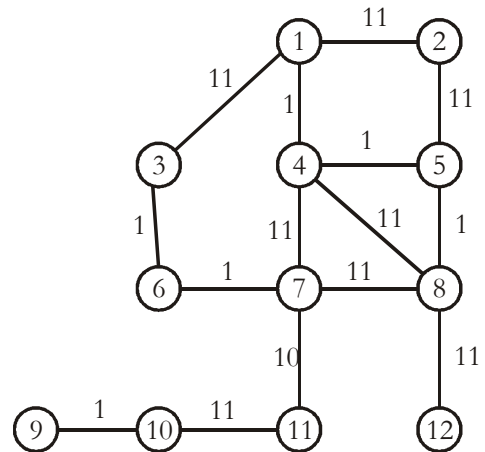
A `csatorna.be` szöveges állomány első sora a csomópontok  $N$  számát ( $1 \leq N \leq 1000$ ), a csatorna szakaszok  $M$  számát ( $1 \leq M \leq 10\ 000$ ), a kiinduló csomópont sorszámát ( $1 \leq S \leq N$ ) és a robot  $R$  méretét ( $1 \leq R \leq 100$ ) tartalmazza, egy-egy szóközzel elválasztva. A következő  $M$  sor mindegyike egy-egy csatorna szakasz két végpontját ( $1 \leq K_i \neq V_i \leq N$ ) és átmérőjét ( $1 \leq A_i \leq 100$ ) tartalmazza.

A `csatorna.ki` szöveges állomány első sorába az  $S$  pontból kitisztítható csatorna szakaszok számát kell írni! A második sorba azon további pontok minimális számát kell írni, amelyekből elindulva az összes olyan csatorna kitisztítható, ahova a robot befér!

Példa:

```
csatorna.be
12 15 4 10
1 2 11
5 8 1
1 3 11
1 4 1
2 5 11
6 7 1
3 6 1
4 7 11
4 8 11
4 5 1
8 7 11
11 7 10
8 12 11
9 10 1
11 10 11
```

```
csatorna.ki
4
2
```



Értékelés:

Egyetlen pontból mindenhova elér	1+0 pont
Fa szerkezet	1+2 pont
A kezdőpontokból minden pont elérhető	1+2 pont
Vannak elérhetetlen pontok	1+2 pont
Vannak elérhetetlen szakaszok	1+2 pont
Olyan helyről indul, ahonnan nem tud semerre menni	1+2 pont
Véletlen közepes teszt	1+2 pont
Véletlen közepes teszt	1+2 pont
Véletlen nagy teszt	2+2 pont
Véletlen nagy teszt	2+2 pont

3. feladat: Poligon (30 pont)

Adott a síkon egy  $N$  csúcsponot tartalmazó zárt, nem metsző törtvonal a csúcsok felsorolásával. Tehát a felsorolásban az  $i$ -edik és  $i+1$ -edik ( $i < N$ ) pont van összekötve egyenes szakasszal, illetve az  $N$ -edik az elsővel.

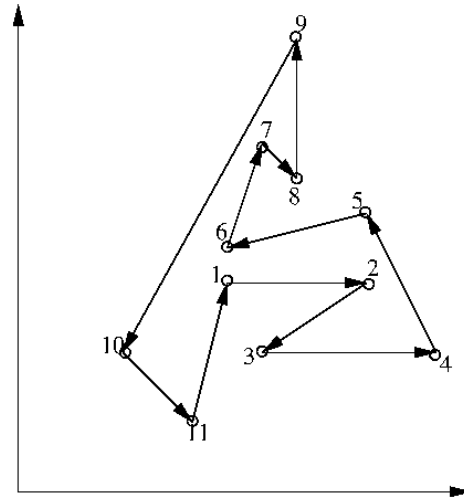
Készíts programot (`poligon.pas`, `poligon.c`, ...), amely megadja azokat az **A**, **B** csúcspárokat, amelyekre teljesül, hogy a törtvonal minden **A**-tól és **B**-től különböző pontja szigorúan balra van, ha **A**-ból **B**-felé nézünk!

A `poligon.be` szöveges állomány első sorában a csúcspontok **N** száma ( $3 \leq N \leq 10\,000$ ) van. A következő **N** sor mindegyike két **X Y** ( $-1\,000\,000 \leq X, Y \leq 1\,000\,000$ ) egész számot tartalmaz, egy csúcspont **x**- és **y**-koordinátáját.

A `poligon.ki` szöveges állomány első sorába azon csúcspárok **M** számát kell írni, amelyekre teljesül a kívánt feltétel. A következő **M** sor mindegyike egy megfelelő csúcspár **A** és **B** sorszámát tartalmazza, egy szóközzel elválasztva. A csúcspárokat tetszőleges sorrendben ki lehet írni. A pontpár **A** és **B** sorrendje lényeges!

Példa:

```
poligon.be      poligon.ki
11              4
6 6             10 11
10 6            11 4
7 4             4 9
12 4            9 10
10 8
6 7
7 10
8 9
8 13
3 4
5 2
```



Értékelés:

Kicsi konvex poligon	1+2 pont
Nincs ilyen pár	1+2 pont
Sarokponttól balra haladó poligon	1+2 pont
Sarokponttól jobbra haladó poligon	1+2 pont
Kis méret, nem a konvex burok a megoldás	1+2 pont
Közepes méret, nem a konvex burok a megoldás	1+2 pont
Nagy méret, nem a konvex burok a megoldás	1+2 pont
Véletlen közepes teszt	1+2 pont
Véletlen nagy teszt	1+2 pont
Véletlen nagy teszt	1+2 pont

4. feladat: Lefedés (30 pont)

Adott **N** pozitív egész szám. Keresünk legfeljebb **K** olyan zárt intervallumot, hogy minden megadott szám benne van valamelyik intervallumban és az intervallumok összhossza a lehető legkisebb. Minden lefedő **[a,b]** intervallumra teljesülni kell, hogy **a < b**. Az intervallum hossza a **b-a** érték.

Készíts programot (`lefed.pas`, `lefed.c`, ...), amely megadja a legkisebb összhosszú lefedő intervallumokat!

Az `lefed.be` szöveges állomány első sorában két egész szám van, a lefedendő számok **N** száma ( $1 \leq N \leq 100\,000$ ) és a lefedésre használható intervallumok számának **K** maximuma ( $1 \leq K \leq N$ ). A második sor pontosan **N** pozitív egész számot tartalmaz (egy-egy szóközzel elválasztva), a lefedendő számokat. A számok nem nagyobbak, mint **2 000 000**.

Az `lefed.ki` szöveges állomány első sorába a lefedő intervallumok összhosszát kell írni. A további legfeljebb  $K$  sorba kell kiírni a lefedő intervallumokat, egy sorba egy intervallum kezdő és végpontját. Az intervallumokat kezdőpontjuk szerint növekvő sorrendben kell kiírni. Több megoldás esetén bármelyik megadható.

Példa:

<code>lefed.be</code>	<code>lefed.ki</code>
7 3	8
3 1 4 11 7 9 15	1 4
	7 11
	15 16

Értékelés:

Kicsi, $K=N$ , a megoldás $N$	1+2 pont
Kicsi, $K=N$ , a megoldás nem $N$	1+2 pont
$K=2$	1+2 pont
$K=N-1$	1+2 pont
Egymást követő számok	1+2 pont
Kis méretű véletlen bemenet	1+2 pont
Közepes méret, véletlen számsor	1+2 pont
Nagy méret, nem véletlen számsor	1+2 pont
Véletlen közepes teszt	1+2 pont
Véletlen nagy teszt	1+2 pont

5. feladat: Játék (40 pont)

Ádám és Éva kétszemélyes játékot játszik egy  $N$  mezőt tartalmazó játéktáblán. A játék során az első mezőről indulva, felváltva lépve, egy bábút mozgatnak a játéktáblán, ugyanarra a mezőre többször is léphetnek. Egy adott mezőről csak szomszédos mezőre lehet lépni egy lépésben. Minden mezőre rá van írva egy pontszám. Ádám kezdi a játékot. Ha Ádám az aktuális lépésében az  $m$  mezőre lép, amire  $p$  pontszám van írva, akkor összpontszáma  $p$  értékkel növekszik. Ha Éva az  $m$  mezőre lép, amire  $p$  pontszám van írva, akkor Ádám összpontszámát csökkentik  $p$  értékkel. Ádám célja, hogy a lehető legtöbb összpontot szerezze, Éva célja pedig az, hogy Ádám a lehető legkevesebb összpontot szerezzen a játék során. Az összpontszám lehet negatív is, ekkor Ádám fizet Évának a játék végén.

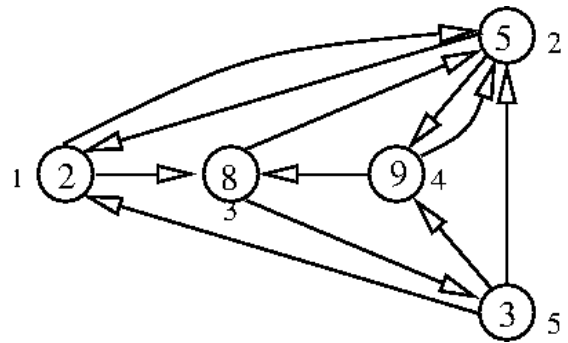
Készíts programot (`jatek.pas`, `jatek.c`, ...), amely kiszámítja, hogy mekkora az a legnagyobb összpontszám, amit Ádám biztosan meg tud szerezni  $K$  lépéses játékban, bárhol is lép Éva!

A `jatek.be` szöveges állomány első sorában két egész szám van egy szóközzel elválasztva, a játéktáblán lévő mezők  $N$  száma ( $1 \leq N \leq 500$ ) és a játékban megteendő lépések  $K$  száma ( $1 \leq K \leq 2000$ ). (Tehát mindkét játékos  $K$  lépést tesz felváltva.) A második sor pontosan  $N$  pozitív egész számot tartalmaz egy-egy szóközzel elválasztva, az  $m$ -edik szám az  $m$  sorozású mezőn lévő pontszám értéke, legfeljebb **1000**. A következő  $N$  sor írja le a mezők szomszédjait, tehát, hogy adott mezőről mely mezőkre lehet lépni közvetlenül. Az állomány  $m+2$ -edik sora az  $m$ -edik mező szomszédjait tartalmazza egy-egy szóközzel elválasztva, 0-val zárva. Minden mezőről legalább egy másik mezőre lehet lépni. Minden mezőnek önmaga is lehet szomszédja.

A `jatek.ki` szöveges állomány első és egyetlen sorába egy egész számot kell írni, a legnagyobb összpontszámot, amit Ádám meg tud szerezni  $K$  lépéses játékban, bárhol is lép Éva!

**Példa:**

```
jatek.be      jatek.ki
5 2           4
2 5 8 9 3
3 2 0
4 1 0
2 5 0
3 2 0
4 2 1 0
```



**Értékelés:**

- Kicsi, mohó is megoldás 4 pont
- Azonos pontok a mezőkön 4 pont
- $N=2$  4 pont
- $K=N-1$  4 pont
- Páros gráf 4 pont
- Fa 4 pont
- Kicsi méret, véletlen bemenet 4 pont
- Közepes méret, nem véletlen 4 pont
- Véletlen közepes teszt 4 pont
- Véletlen nagy teszt 4 pont

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