CEOI 2021

CENTRAL EUROPEAN OLYMPIAD IN INFORMATICS 2021

ZAGREB, CROATIA | SEPTEMBER 1-5

Day 1

September 2nd 2021

Tasks

Task	Time Limit	Memory Limit	Score
Diversity	7 seconds	512 MiB	100
L-triominoes	8 seconds	512 MiB	100
Newspapers	1 second	512 MiB	100
Total			300

REPUBLIC OF CROATIA
Ministry of Science and Education

CROATIAN ASSOCIATION OF TECHNICAL CULTURE

CROATIAN COMPUTER SCIENCE ASSOCIATION

Task: Diversity

Zorán a Zágrábi Állatkert gondozója. Tudományos kutatást végez a látogatók megelégedettsége és az állatok közötti megtett útvonaluk közti kapcsolatról.

A látogatók a teljes állatkertet bejáró útjuk során N élőhelyet tekintenek meg. Az élőhelyek mindegyike egyfajta állatot tartalmaz. Az i. élőhely az a_i fajtájú állatot tartalmazza. A látogatók az élőhelyek sorrendjében haladnak.

Zorán a kutatást az állatok meglévő sorrendjének felmérésével kezdi. Utána a látogatókat kérdezi arról, hogy mennyire vannak megelégedve a látogatásukkal. A kutatás azt eredményezte, hogy a látogatók akkor a legjobban elégedettek, ha a sétájuk *teljes sokszínűsége* olyan alacsony értékű, amennyire az csak lehetséges.

Az élőhelyek egy sorozatának *sokszínűség*e az élőhelyeken levő különböző állatfajták száma. Az élőhelyek sorozatának *teljes sokszínűségé*t az összes, összefüggő részsorozatai sokszínűségének összege adja.

Például az $(1, 1, 2)$ sorozat sokszínűségének az értéke 2, mert kétfajta állat található a három élőhelyen, míg a teljes sokszínűsége 8, mivel a következő összefüggő részsorozatok az (1), (1), (2), $(1, 1)$, $(1, 2)$, $(1, 1, 2)$ sokszínűségének értékei rendre 1, 1, 1, 1, 2, 2.

Zorán tudja az állatfajták eredeti elhelyezkedését az élőhelyeken. Mielőtt átrendezi az állatkertet, **egymástól független** Q kérdést tesz fel. Az i. kérdésben azt kérdezi, hogy mi a lehető legkisebb **teljes sokszínűsége** az l_i. élőhelytől az r_i. élőhelyig tartó összefüggő részsorozatának (beleértve a határokat is), ami az ezeken az élőhelyeken lakó állatok **átrendezésével** elérhető.

Bemenet

Az első sor az N és Q egész számokat tartalmazza.

A második sorban N darab, szóközzel elválasztott $a_1, a_2, \ldots, a_N$ egész szám van. Az i. szám az i. élőhelyen található állat fajtáját adja meg.

A következő Q sorból a i. sor az i. kérdést leíró l_i és r_i $(1 \leq l_i \leq r_i \leq N)$ egész számokat tartalmazza.

Megjegyezzük, hogy a kérdések függetlenek egymástól abban az értelemben, hogy az i. élőhelyen mindig az eredeti elrendezésben megadott a_i fajtájú állattal számolunk.

Kimenet

A kimenet i. sora az i. kérdésre adandó, egyetlen egész szám választ tartalmazza.

Pontozás

Részfeladat	Pontozás	Feltételek
1	4	$1 \leq N \leq 11$, $1 \leq a_i \leq 300\,000$, $Q = 1$, $l_1 = 1$, $r_1 = N$
2	10	$1 \leq N \leq 300\,000$, $1 \leq a_i \leq 11$, $Q = 1$, $l_1 = 1$, $r_1 = N$
3	8	$1 \leq N \leq 300\,000$, $1 \leq a_i \leq 23$, $Q = 1$, $l_1 = 1$, $r_1 = N$
4	16	$1 \leq N \leq 300\,000$, $1 \leq a_i \leq 1\,000$, $Q = 1$, $l_1 = 1$, $r_1 = N$
5	26	$1 \leq N \leq 300\,000$, $1 \leq a_i \leq 300\,000$, $Q = 1$, $l_1 = 1$, $r_1 = N$
6	36	$1 \leq N \leq 300\,000$, $1 \leq a_i \leq 300\,000$, $1 \leq Q \leq 50\,000$

Példák

input

```
3 1
1 2 3
1 3
```

output

```
10
```

input

```
4 2
1 1 1 1
1 2
2 4
```

output

```
3
6
```

input

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

output

```
16
8
4
```

Az első példa értelmezése: Minden átrendezés esetén minden összefüggő részsorozat sokszínűsége megegyezik a sorozat elemeinek számával, tehát a helyes válasz értéke: $1+1+1+2+2+3=10$.

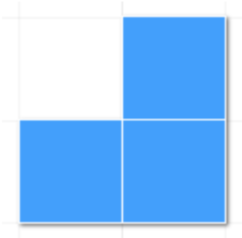
A második példa értelmezése: Minden átrendezés esetén minden összefüggő részsorozat sokszínűségének értéke 1. Tehát minden lekérdezésben a válasz a kérdésben szereplő összefüggő részsorozatainak a száma.

A harmadik példa értelmezése: Az első lekérdezésben az optimális átrendezés $(1,2,2,3)$, amely esetén a teljes sokszínűség értéke $1+1+1+1+2+1+2+2+2+3=16$. A második lekérdezésben az optimális átrendezés $(1,1,2)$, amely esetén a teljes sokszínűség értéke $1+1+1+1+2+2=8$. A harmadik lekérdezésben az optimális átrendezés $(1,3)$, amely esetén a teljes sokszínűség értéke to $1+1+2=4$.

Task: L-triominoes

Luka egy négyzetrácsos táblára talált, amely W szélességű és H hosszúságú, $W \times H$ egységnégyzetre van osztva. Megállapította, hogy pontosan K darab egységnégyzet hiányzik belőle.

Lukanak korlátlan számú, *L-alakú triomino*-ja van. A kérdés, hogy a tábla helyesen lefedhető-e ilyen triominokkal?



A helyes lefedésen azt értjük, hogy a tábla minden meglévő négyzete le van fedve pontosan egy ilyen triomino egy négyzetével, tehát a triominok egyike sem takar hiányzó négyzetet vagy egyik négyzeten sincs egynél több triomino vagy lóg le a tábláról. Természetesen a triominok akárhányszor elforgathatók 90°-kal.

Bemenet

Az első sor a feladatleírásban szereplő W, H és K ($0 \le K \le W \times H$) egész számokat tartalmazza.

A következő K sorból az i. tartalma két egész szám: x_i ($1 \le x_i \le W$) és y_i ($1 \le y_i \le H$), az i. hiányzó négyzet koordinátái. Az adott értékek páronként eltérőek.

Kimenet

Ha Luka helyesen le tudja fedni a táblát, akkor a kimenet első és egyetlen sora a `"YES"` szöveg, egyébként a `"NO"` szöveg.

Pontozás

Részfeladat	Pontozás	Feltételek
1	10	$2 \le W \le 13, 2 \le H \le 1\,000, K \le 250$
2	7	$2 \le W \le 13, 2 \le H \le 10^9, K = 0$
3	11	$2 \le W \le 3, 2 \le H \le 10^9, K \le 250$
4	17	$4 \le W \le 6, 2 \le H \le 10^9, K \le 250$
5	35	$7 \le W \le 13, 2 \le H \le 10^9, K \le 250$
6	20	$2 \le W \le 13, 2 \le H \le 10^9, K \le 250$

Példák

input

```
4 3 3
1 1
1 3
4 3
```

output

```
YES
```

input

```
5 2 4
1 2
2 1
5 1
5 2
```

output

```
NO
```

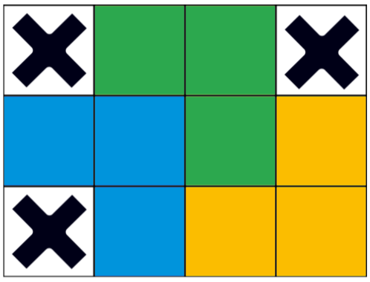
input

```
2 3 0
```

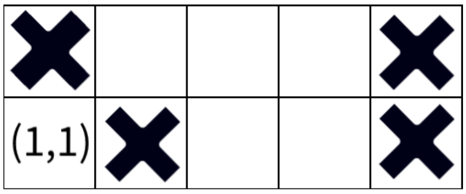
output

```
YES
```

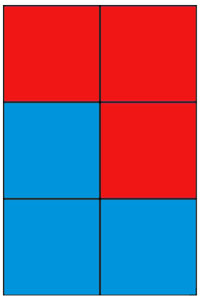
Az első példa értelmezése:



A második példa értelmezése: Luka semmiképpen nem tudja lefedni az $(1,1)$ koordinátájú négyzetet.



A harmadik példa értelmezése:



Task: Newspapers

„*Ulovi me, ulovi me, kupit ću ti novine!*" – a horvát gyerekek kedvenc dala. Magyarra fordítva: *kapj el, kapj el, s veszek neked újságot.*

Ankica és Branko játszanak egy irányítatlan, összefüggő gráfon. Branko mozog a gráfon, Ankica pedig el akarja kapni. A játék több fordulóból áll, egy forduló a következőképpen zajlik:

- **Ankica megtippeli Branko helyét.** Pontosabban azt tippeli meg, hogy Branko a gráf egy adott pontjában van-e. Ha eltalálta, akkor elkapja Brankot, s ezzel vége a játéknak.

  Egyébként

- **Branko az aktuális pontjából a gráf egy szomszédos pontjába megy.** Megjegyezzük, hogy Branko nem maradhat egyhelyben!

A kérdés, hogy adott gráfon Ankicának van-e véges lépésben végrehajtható stratégiája, amellyel mindig elkaphatja Brankot - függetlenül attól, hogy Branko hogyan lép és mi volt a kezdeti helyzete.

Formálisan Ankica stratégiája leírható egy $A = (a_1, a_2, \ldots, a_k)$ sorozattal, ahol a_i Ankica i. fordulóban adott tippje (vagyis az i. fordulóban azt tippeli, hogy Branko a gráf a_i pontjában van).

Hasonlóan Branko helyzetét formálisan a $B = (b_1, b_2, \ldots, b_k)$ sorozattal adhatjuk meg, ahol b_i a gráf azon pontja, ahol Branko tartózkodik az i. forduló előtt. Természetesen a sorozat két egymást követő b_i és b_{i+1} $(1 \leq i < k)$ elemeire teljesül, hogy van köztük él a gráfban. Megjegyezzük, hogy az A sorozatra nincsen semmilyen feltétel.

Akkor mondjuk, hogy Ankica stratégiája nyerő - azaz legfeljebb k fordulóban elkapja Brankot - ha minden érvényes k elemű B sorozatra teljesül, hogy van olyan i $(1 \leq i \leq k)$, melyre $a_i = b_i$, azaz valamelyik fordulóban helyesen tippelt Ankica.

Ha létezik ilyen stratégia, akkor a lehetséges legkisebb k értéket keressük.

Akkor is szerezhetsz pontot, ha helyes, de nem a legrövidebb nyerő stratégiát találod meg, azaz ahol k nem minimális. Lásd a *Pontozás* részt.

Bemenet

A bemenet első sora az N és M $(N - 1 \leq M \leq \frac{N(N-1)}{2})$ egész számokat tartalmazza, N a gráf pontjainak és M a gráf éleinek a száma. A gráf pontjait az $1...N$ számokkal jelöljük.

A következő M sorból az i. két, szóközzel elválasztott egész számot tartalmaz: u_i és v_i $(1 \leq u_i, v_i \leq N, u_i \neq v_i)$, ami azt jelenti, hogy a gráfban az u_i és v_i pontok közt van él.

Minden él pontosan egyszer szerepel a bemenetben.

Kimenet

Ha nincs Ankica számára nyerő stratégia, akkor csak a `"NO"` szöveg kerüljön a kimenet első és egyetlen sorába.

Egyébként az első sorba a `"YES"` szöveget kell írni.

A második sorba a stratégia fordulóinak k számát kell írni.

A harmadik sor tartalmazza a nyerő stratégiát megadó k elemű sorozatot: a_1, a_2, $\ldots$, a_k.

Pontozás

Részfeladat	Pontozás	Feltételek
1	12	$1 \leq N \leq 20$
2	8	$1 \leq N \leq 1\,000$, $M = N - 1$, a gráf élei az u és $u+1$ pontpárok (minden $u = 1, \ldots, N-1$-re)
3	80	$1 \leq N \leq 1\,000$

Ha a programod helyesen kiírja a `"YES"` szót az első sorba az adott tesztesetre, de nem adja meg a nyerő stratégiát, akkor a tesztesetre járó pont 50% -át kapod.

Ha a programod helyesen kiírja a `"YES"` szót az első sorba az adott tesztesetre, és megad egy helyes, de nem optimális nyerő stratégiát, akkor a tesztesetre járó pont 75% -át kapod. Csak akkor jár a pont, ha a stratégiádban megadott sorozat hossza (k) legfeljebb $5N$. Bizonyítható, hogy minden optimális nyerő stratégia hossza legfeljebb $5N$.

Példák

input

```
7 6
1 2
1 3
1 4
1 5
1 6
1 7
```

output

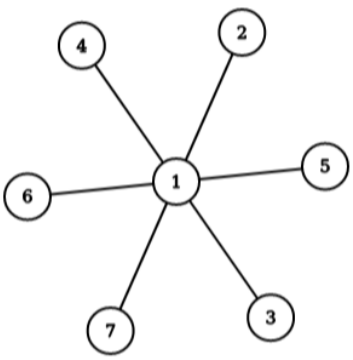
```
YES
2
1 1
```

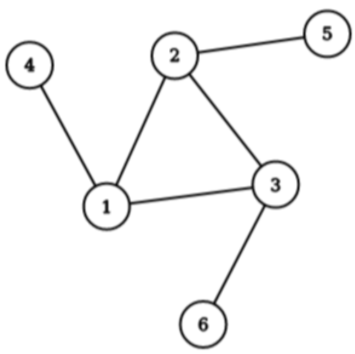
input

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

output

```
NO
```





Az első példa értelmezése: Ha Branko kezdetben az 1 pontban van, akkor Ankica az első fordulóban elkapja, egyébként a másodikban.

A második példa értelmezése: Tegyük fel, hogy Branko kezdeti helyzete az 1, 2 vagy a 3, és különbözik a_1-től. Ezen pontok mindegyike két másikkal kapcsolódik, így Branko minden fordulóban választhat a kettő közül. Tehát Ankicanak nincs nyerő stratégiája.