# 1. Jancsi és a tömbök

* **időkorlát:** 2 másodperc/teszt
* **bemenet:** jancsi.in
* **kimenet:** jancsi.out

Jancsinak van egy *n* elemű, egész számokból álló tömbje, *𝑎*, és egy 𝑚 elemű egész számokból álló tömbje, 𝑏 (𝑚 ≤ n).

Jancsi akkor tekint **jó tömbnek** egy 𝑚 hosszúságú 𝑐 tömböt, ha a 𝑐 tömb elemeit úgy lehet átrendezni, hogy legalább 𝑘 belőlük megegyezik a 𝑏 tömb elemeivel.

Például, ha 𝑏=[1, 2, 3, 4] és 𝑘=3, akkor

* a [4, 1, 2, 3] és [2, 3, 4, 5] tömbök **jók** (mert úgy átrendezhetők, hogy [1, 2, 3, 4] és [5, 2, 3, 4])
* a [3, 4, 5, 6] és [3, 4, 3, 4] tömbök **nem jók**..

Jancsi az *𝑎* tömb minden egyes részsorozatát kiválasztja 𝑚 hosszúságú 𝑐 tömbként. Segíts Jancsinak megszámolni, hány kiválasztott tömb lesz **jó tömb**.

Más szóval, keressük meg azon 𝑙 pozíciók számát, ahol 1 ≤ 𝑙 ≤ *n*−𝑚+1, és az *𝑎𝑙, 𝑎𝑙+1,…,𝑙+𝑚−1* elemek egy **jó tömböt** alkotnak.

**Bemenet**

* Az első sor egy 𝑡 egész számot tartalmaz (1 ≤ 𝑡 ≤ 104) - a tesztesetek számát.
* Minden teszteset első sorában három egész szám található: *n*, 𝑚 és 𝑘 (1 ≤ 𝑘 ≤ 𝑚 ≤ *n* ≤ 2⋅105) - az *𝑎* és 𝑏 tömbök elemeinek száma, a megfelelő számú illeszkedő elem.
* Minden teszteset második sorában *n* egész szám áll: 𝑎1, 𝑎2,…, 𝑎n (1 ≤ *𝑎*𝑖 ≤ 106) - az *𝑎* tömb elemei. **Az** *a* **tömb elemei nem feltétlenül egyediek.**
* Minden teszteset harmadik sorában 𝑚 egész szám található: 𝑏1, 𝑏2,…,*b*𝑚 (1 ≤ 𝑏𝑖 ≤ 106) - a 𝑏 tömb elemei. **A 𝑏 tömb elemei nem feltétlenül egyediek.**

**Kimenet**

* Minden teszteset esetén külön sorba írd ki az *𝑎* tömb **jó tömbjeinek** számát.

**Példa**

|  |  |
| --- | --- |
| **Bemenet** | **Kimenet** |
| **5**  7 4 2  4 1 2 3 4 5 6  1 2 3 4  7 4 3  4 1 2 3 4 5 6  1 2 3 4  7 4 4  4 1 2 3 4 5 6  1 2 3 4  11 5 3  9 9 2 2 10 9 7 6 3 6 3  6 9 7 8 10  4 1 1  4 1 5 6  6 | 4  3  2  4  1 |

**Megjegyzések**

* Az első példában az összes résztömb jó.
* A második példában a jó résztömbök az 1., 2. és 3. pozícióban kezdődnek.
* A harmadik példában a jó résztömbök az 1. és 2. pozícióban kezdődnek.

# 2. Túravonalak

* **időkorlát:** 2 másodperc/teszt
* **bemenet:** tura.in
* **kimenet:** tura.out

Egy N×M méretű egész értékeket tartalmazó mátrix egy terepet modellez (1<N, M<19). Az elemek értékei a [-10,10] intervallumból vannak és a megfelelő pozíciójú pont tengerszint feletti/alatti magasságát/mélységét jelenti. Az északi oldalon lépünk be a területre (első sor), minden lépésben egy sornyit haladunk úgy, hogy egyet lépünk dél, dél-nyugat vagy dél-kelet irányba, majd a déli oldalon (utolsó sor) hagyjuk el a területet. Hány útvonal közül választhatunk, ha tengerszint alá nem akarunk „süllyedni” (a megfelelő elem negatív) és maximális számú nyeregponton szeretnénk áthaladni. Nyeregpontnak tekintjük azokat az elemeket, amelyek nem negatívak, szigorúan nagyobbak az északi és déli szomszédjaiknál, és szigorúan kisebbek a nyugati és keleti szomszédjaiknál.

**Bemenet**

* Az első sor egy t egész számot tartalmaz (0 < t < 9) - a tesztesetek számát.
* Minden teszteset első sorában két egész szám található, az N és M értékek.
* Minden teszteset következő N sorában a mátrix sorai találhatók, minden sorban M egész szám egy-egy szóközzel elválasztva.

**Kimenet**

* Minden teszteset esetén külön sorba írd ki az útvonalak számát.

**Példa** (egy teszteset; a nyeregpontokat aláhúztuk)

|  |  |
| --- | --- |
| **Bemenet** | **Kimenet** |
| 1  7 5  -1 2 3 3 -1  5 3 4 4 -1  4 2 5 1 2  1 1 5 4 5  4 2 3 1 4  4 3 4 2 3  2 2 5 1 1 | 24 |

# 3. Medvebiztos legrövidebb út

* **időkorlát:** 2 másodperc/teszt
* **bemenet:** medve.in
* **kimenet:** medve.out

Képzeljünk el egy n szintes „hegyet” (i=1,2,…,n), amelynek szintjeit mxm méretű bináris mátrixok kódolnak. Mindenik mátrixban 1-esek jelzik a hegyhez tartozó pontokat, és ismert, hogy az 1-esek is, és a 0-sok is összefüggő részt alkotnak, amennyiben szomszédosnak akkor tekintünk két elemet, ha van közös oldaluk. További megkötés, hogy a hegy-szeletek „koncentrikusak” abban az értelemben, hogy az i. szinti (i=2,…,n) 1-esek pozícióiban az (i-1). szinten is 1-esek vannak. Az utolsó szinti mátrixban egyetlen 1-es található, amely a csúcsot képviseli. Mivel minden szint belső pontjai számítanak medve-veszélyesnek, ezért az alábbi módszerrel szeretnénk feljutni a csúcsra: (i) a 0. szint bármelyik perem-pontjától indulhatunk; (ii) minden szint peremétől a következő szint pereméig minimális lépésből szeretnénk eljutni, a legfelső szinten a csúcspontig; (iii) a felkapaszkodás a következő szintre, illetve valamely szint peremén való haladás veszélytelennek számít (ezeket a részeket kerülik a medvék, merthogy tériszonyuk van). Amennyiben a fentebb vázolt stratégiát követjük, hány lépést teszünk veszélyes területen az alaptól a csúcsig?

A **bement** első sora a t értéket tartalmazza, ami a tesztesetek számát jelöli. Minden tesztesethez: (i) az első sor tartalmazza a n és m értékeket, azaz a szintek számát és a négyzetes mátrixok méretét (1 < n < 100, 3 < m < 100); (ii) a következő n · m sor mindegyike tartalmaz m bináris értéket (0/1), egy szóközzel elválasztva, megadva a mátrixok elemeit.

Minden teszteset kapcsán egyetlen sort kell kiírni a **kimenetre**, ami tartalmazza a veszélyes területen megtett lépések számát.

Példa bement

|  |  |
| --- | --- |
| **Bemenet** | **Kimenet** |
| 1  3 10  **1 1 1 1 1 1 1 1 1 1**  **1** 1 1 1 1 1 1 1 1 **1**  **1** 1 1 1 1 1 1 1 **1** 0  **1** 1 1 1 1 1 1 1 1 **1**  **1** 1 1 1 1 1 1 1 1 **1**  **1** 1 1 1 1 1 1 1 1 **1**  **1** 1 1 1 1 1 1 1 1 **1**  **1** 1 1 1 1 1 1 1 1 **1**  **1** 1 1 1 1 1 1 1 1 **1**  **1 1 1 1 1 1 1 1 1 1**  0 0 0 0 0 0 0 0 0 0  0 0 0 0 0 0 0 0 0 0  0 0 0 0 **1 1 1 1** 0 0  0 0 0 **1** 1 1 1 **1** 0 0  0 0 0 0 **1** 1 1 **1** 0 0  0 0 0 **1** 1 1 1 **1** 0 0  0 0 0 **1** **1** **1** **1** 0 0 0  0 0 0 0 0 0 0 0 0 0  0 0 0 0 0 0 0 0 0 0  0 0 0 0 0 0 0 0 0 0  0 0 0 0 0 0 0 0 0 0  0 0 0 0 0 0 0 0 0 0  0 0 0 0 0 0 0 0 0 0  0 0 0 0 0 0 0 0 0 0  0 0 0 0 0 **1** 0 0 0 0  0 0 0 0 0 0 0 0 0 0  0 0 0 0 0 0 0 0 0 0  0 0 0 0 0 0 0 0 0 0  0 0 0 0 0 0 0 0 0 0  0 0 0 0 0 0 0 0 0 0 | 2 |

**Magyarázat**

A példa bemenet egyetlen tesztesetből (t = 1) áll, ami egy 3 szintből álló hegyet reprezentál, amelyet 10 × 10-es mátrixok kódolnak. A vastagított 1-ek a perempontokat jelölik. Az aláhúzott 1-ek az optimális út veszélyes pontjait ábrázolják. Az 1. szinten, a 2. szint perempontjaihoz 1 belsőponti, azaz veszélyes lépés vezet ([3,9]–[3,8]), és a 2. szinten, a 3. szinti csúcsnak a pozíciójához is 1 belsőponti, vagyis veszélyes lépés vezet ([5,5)–[5,6]).