Feladat Detonator

Bemenet	`detonator.in`
Kimenet	`detonator.out`

Roman felügyelő egy robbanó szerkezet előtt áll, ami gyakorlatilag egy N szintű piramis, amelynek szintjeit 1-től N-ig számozzuk. Minden i szinten i bomba található, melyeket $B_{i,j}$ jelöl, ahol j az adott szinten található bomba sorszáma. Minden $B_{i,j}$ bomba esetén tudjuk, hogy a kezdeti időponttól számítva $T_{i,j}$ másodperc után bekövetkezik a robbanás. Roman feladata az, hogy hatékonyan hatástalanítsa a bombákat, a következő szabályok szerint:

1. ábra. Példa $N = 4 - re$. A $T_{i,j}$ értékek a bombákra vannak írva.

1. Egy bomba hatástalanításának ideje 1 másodperc;
2. Ahhoz, hogy hatástalaníthassuk a $B_{i,j}$ bombát, előbb hatástalanítani kell az alatta levő két bombát, amelyeken ez fekszik, vagyis $B_{i+1,j}$ és $B_{i+1,j+1}$. Az N-ik szinten levő bombák alatt nincsenek más bombák, tehát nem teljesítik ezt a szabályt.

A szerkezet akkor lesz hatástalanítva, ha minden bomba hatástalanítva van. Roman nem szeretne kapkodni, ezért tudni szeretné, hogy mennyi a maximális X idő másodpercben mérve úgy, hogy ha a hatástalanítási művelet X másodpercet késik, a szerkezet még mindig hatástalanítható legyen. Más szóval mondva, mennyi X legnyagyobb értéke, amelyre a hatástalanítás még megvalósítható, ha $T_{i,j}$ helyett $T_{i,j} - X$ értéket rakunk minden $1 \leq j \leq i \leq N$ esetben. Még az is előfordulhat, hogy X egy negatív szám, ha Roman túl későn érkezett a szerkezethez. Például, ha a szerkezet nem hatástalanítható az adott esetben, de ha Roman egy másodperccel hamarabb érkezett volna, akkor a hatástalanítás sikeres lenne, akkor$X = -1$. Ha a hatástalanítás egy másodperccel hamarabb nem sikerült volna, de két másodperccel hamarabb sikerült volna, akkor $X = -2$, és így tovább.

Követelmény

Adott Q teszt, mindegyikről ismerjük N-et és a $T_{i,j}$ értékeket minden $1 \leq j \leq i \leq N$ esetén, és ki kell számoljuk X értékét.

Bemeneti adatok

A `detonator.in` bemeneti állomány első sora tartalmazza Q-t, a tesztek számát. Következik a Q teszt leírása. Minden teszt esetében az első sor tartalmazza N értékét. A következő N sor mindegyike, 1-től N-ig számozva az adott szerkezet egy szintjét egyetlen sorban, természetes számokat, ahol a j-edik szám az i-edik sorból $T_{i,j}$. Vegyétek észre, hogy az i-ik sor i elemet tartalmaz.

Kimeneti adatok

A `detonator.out` kimeneti állomány Q sort fog tartalmazni, minden teszt esetében az X számot, sorrendben.

Korlátok

- $1 \leq Q \leq 5$
- $1 \leq N \leq 1\,000$
- $1 \leq T_{i,j} \leq 10^9$, minden $1 \leq j \leq i \leq N$.

#	Pontszám	Korlátok
1	7	A $T_{i,j}$ értékek minden $1 \leq j \leq i \leq N$ esetén egyenlőek (minden bomba ugyanabban a pillanatban fog robbanni).
2	13	$N \leq 5, \lvert X \rvert \leq 10$
3	9	$N \leq 5$
4	14	$N \leq 50, \lvert X \rvert \leq 10$ si $T_{i,j} \geq \max\{T_{i+1,j}, T_{i+1,j+1}\}$ minden $1 \leq j \leq i < N$
5	13	$N \leq 50, \lvert X \rvert \leq 10$
6	2	$N \leq 50$
7	9	$N \leq 200, \lvert X \rvert \leq 10$
8	2	$N \leq 200$
9	19	$N \leq 500$
10	12	Más megszorítások nincsenek.

Példák

detonator.in	detonator.out	Magyarázat
4 2 10 10 10 4 10 7 9 4 6 8 1 3 2 5 3 9 9 9 1 1 1 3 6 5 3 4 4 4	7 0 -2 0	Az első tesztben $N = 2$, és van $N \cdot (N + 1)/2 = 3$ bombánk. Minden bomba 10 másodperc után robban. Tehát Roman legtöbb 7 másodpercet várhat, amikor még hatástalaníthatja a bombákat a 8-ik, 9-ik illetve 10-ik másodpercben. A második teszt az 1. ábrán látható. Ebben az esetben Roman azonnal el kell kezdje a bombák hatástalanítását, ugyanis a $B_{4,1}$ bomba egy másodperc után robban. Az egyetlen helyes hatástalanítási sorrend a bombák $B_{i,j}$ értéke pontosan ebben a sorrendben $1, 2, \ldots, 10$. A harmadik teszt esetében létezik három bomba, amelyik azonnal robban, tehát Roman 2 másodperccel hamarabb kellett volna jöjjön, hogy sikeresen hatástalanítsa a szerkezetet.