Fénykép probléma

Egy rendezvényre n adott számú vendéget hívtak meg. Minden vendég előre jelezte, hogy mettől meddig lesz jelen. A szervezők fényképeken akarják megörökíteni a rendezvényen résztvevőket. Azt tervezik, hogy kiválasztanak k időpontot és minden kiválasztott időpontban az akkor éppen jelenlevőkről csoportképet készítenek. Az a céljuk, hogy a lehető legkevesebb képet kelljen készíteni, de mindenki rajta legyen legalább egy képen.

Adjunk meg egy olyan megoldást, amely kiszámítja, hogy **legkevesebb hány fényképet kell készíteni**, és megadja azokat az **időpontokat** is, amikor csoportképet kell készíteni!

Bemenet: A `fenykep.be` szöveges állomány első sorában a vendégek $\boldsymbol{n}$ száma van ($1 \leq n \leq 3000$). A következő $\boldsymbol{n}$ sor mindegyike két egész számot tartalmaz egy szóközzel elválasztva, egy vendég $\boldsymbol{e}$ érkezési és $\boldsymbol{t}$ távozási időpontját ($1 \leq e < t \leq 1000$). Ha egy fényképet az x időpontban készítik és $e \leq x < t$, akkor azon a fényképen rajta lesz az e időben érkező és t időben távozó vendég.

Kimenet: A `fenykep.ki` szöveges állomány első sorába a készítendő fényképek $\boldsymbol{k}$ számát kell írni! A második sor pontosan $\boldsymbol{k}$ egész számot tartalmazzon egy-egy szóközzel elválasztva, azon időpontokat (tetszőleges sorrendben), amikor a csoportképeket készíteni kell.

Megoldás: Akkor fényképezzünk, amikor feltétlenül szükséges! Ez azt jelenti, hogy amikor elmenne az első ember, aki még nem volt rajta egy fényképen sem, akkor fényképeznünk kell.

Példa: A vendégek száma 6 ($n = 6$).

```
bemenet
6
2 4
1 4
2 7
7 13
5 10
3 9
```

A távozási idő (t) alapján sorba rendezzük a vendégeket:

```
1 4
2 4
2 7
3 9
5 10
7 13
```

A bemeneti intervallum:

$$I = \{[e_1, t_1), \dots, [e_n, t_n)\} = \{[1,4), [2,4), [2,7), [3,9), [5,10), [7,13)\}$$

Az I probléma megoldása a fényképek készítésének időpontjainak (f) halmaza:

$M = \{f_1, \dots, f_k\}$, ahol $f_1 < \dots < f_k$ és
minden i-re ($i = 1, \dots, n$) van olyan $f \in M$, hogy $e_i \leq f < t_i$

Az M első elemét az I problémában lévő első vendég távozási idejéből számítjuk ki:

$$f_1 = t_1 - 1 = 4 - 1 = \mathbf{3}$$

(A -1 azért kell, mivel a vendég 3-kor már távozik és csak egész értékekkel dolgozunk.)

Ezután töröljük I-ből minden olyan intervallumot, amelyben benne van az $f_1 = 3$.

$$I' = I - \{[e_i, t_i) :\ e_i \leq f_1 < t_i\} = \{[e_5, t_5), [e_6, t_6)\} = \{[5,10), [7,13)\}$$

Az I' probléma megoldása:

$$M' = \{f_2, \dots, f_k\}$$

Az M megoldáshalmaz második elemét az I' problémában lévő első vendég távozási idejéből számítjuk ki:

$$f_2 = t_5 - 1 = 10 - 1 = \mathbf{9}$$

Ezután töröljünk I'-ből minden olyan intervallumot, amelyben benne van az $f_2 = 9$.

$I'' = \emptyset$, tehát $M = \{3, 9\}$ és $k = 2$

```
kimenet
2
3 9
```

Előadás szervezési probléma

Egy termék bemutatóra több meghívott vendég várható, de mindenki más időpontban érkezik, és más időpontban távozik. A rendezvényen egy terméket szeretnének népszerűsíteni, de a bemutató előadást a lehető legkevesebbszer szeretnék előadni. A következő adatok ismertek: az előadások hossza, az előadások között tartott szünetek minimális hossza, a vendégek száma, valamint a vendégek érkezési és távozási időpontjai.

Adjunk meg egy olyan megoldást, amely kiszámítja, hogy legkevesebb **hány előadást kell tartani**, és megadja az **előadások kezdési időpontjait** is!

Bemenet: Az `eloadas.be` szöveges állomány első sorában az előadások $\boldsymbol{h}$ hossza ($0 < h < 60$), az előadások közötti szünetek $\boldsymbol{s}$ hossza ($0 < s < 60$), valamint a vendégek $\boldsymbol{n}$ száma van ($1 \leq n \leq 3000$). A következő $\boldsymbol{n}$ sor mindegyike két egész számot tartalmaz egy szóközzel elválasztva, egy vendég $\boldsymbol{e}$ érkezési és $\boldsymbol{t}$ távozási időpontját ($0 \leq e < t \leq 23$).

Kimenet: Az `eloadas.ki` szöveges állomány első sorába a megtartandó előadások $\boldsymbol{k}$ számát kell írni! A második sor pontosan $\boldsymbol{k}$ egész számot tartalmazzon egy-egy szóközzel elválasztva, azon időpontokat, amikor az előadásokat kezdeni kell.

Példa: Az előadások 50 perc hosszúak ($h = 50$), az előadások között 10 perc szünet van ($s = 10$) és a vendégek száma 10 ($n = 10$).

```
    bemenet
    50 10 10
 1. 10 12
 2. 11 13
 3. 12 14
 4. 10 13
 5. 13 14
 6. 14 17
 7. 12 15
 8. 11 12
 9. 13 15
10. 15 17
```

1) A távozási idő (t) alapján sorba rendezzük a vendégeket:

```
 1. 10 12  → 11:00-11:50
 8. 11 12
 4. 10 13
 2. 11 13
 3. 12 14  → 13:00-13:50
 5. 13 14
 7. 12 15
 9. 13 15
 6. 14 17  → 16:00-16:50
10. 15 17
```

2) Vegyük sorra a vendégeket és tegyük be az első előadás intervallumba, amibe belefér. Ha nincs ilyen intervallum, akkor létrehozunk egy olyat, aminek még azelőtt van vége, mielőtt a vendég távozik.

Megoldás:

1. előadás - 11:00-11:50 {$1.\ [10,12),\ 8.\ [11,12),\ 4.\ [10,13),\ 2.\ [11,13)$ }
2. előadás - 13:00-13:50 {$3.\ [10,14),\ 5.\ [13,14),\ 7.\ [12,15),\ 9.\ [13,15)$ }
3. előadás - 16:00-16:50 {$6.\ [14,17),\ 10.\ [15,17)$ }

```
kimenet
3
11 13 16
```

Teremfoglalási probléma

Egy rendezvényen n számú előadást szeretnének tartani. Minden előadó megadta, hogy az előadását mettől meddig tartaná. A szervezők célja, hogy mindenki megtarthassa az előadását, de minél kevesebb termet szeretnének kibérelni.

Adjunk meg egy olyan megoldást, amely kiszámítja, hogy **legkevesebb hány termet** kell biztosítani az előadásoknak, hogy mindegyiket megtarthassák, valamint megad egy lehetséges **terembeosztást** is!

Bemenet: A `terem.be` szöveges állomány első sorában az előadások $\boldsymbol{n}$ száma van ($1 \leq n \leq 3000$). A következő $\boldsymbol{n}$ sor mindegyike két egész számot tartalmaz egy szóközzel elválasztva, egy előadás $\boldsymbol{k}$ kezdetének és $\boldsymbol{v}$ végének időpontját ($0 \leq k < v \leq 23$). Az előadások a kezdő időpontjuk szerint nem csökkenő sorrendben vannak.

Kimenet: A `termek.ki` szöveges állomány első sorába az előadások beosztásához minimálisan szükséges termek $\boldsymbol{t}$ számát kell írni. A további $\boldsymbol{t}$ sor azt adja meg, hogy az előadásokat mely termekbe osztottuk be. Az állomány $i+1$-edik sorában azoknak az előadásoknak a sorszámai vannak felsorolva időrendi sorrendben, amelyeket az i-edik terembe osztottunk be. Több megoldás esetén bármelyik megadható.

Példa: Az előadások száma 10 ($n = 10$).

```
    bemenet
    10
 1. 1  3
 2. 2  4
 3. 2  5
 4. 2  4
 5. 3  6
 6. 4  7
 7. 4  9
 8. 5  7
 9. 6  9
10. 10 12
```

1) A kezdési idő (k) alapján sorba rendezzük az előadásokat
   (ha az adatokat eleve rendezve kapjuk, akkor ez a lépés kihagyható):

```
 1. 1  3   → 1
 2. 2  4   → 2
 3. 2  5   → 3
 4. 2  4   → 4
 5. 3  6   → 1
 6. 4  7   → 2
 7. 4  9   → 4
 8. 5  7   → 3
 9. 6  9   → 1
10. 10 12  → 1
```

2) Vegyük sorra az előadásokat és tegyük be az első terembe, ahova (ütközés nélkül) betehetők! Ha mindegyik terem foglalt, akkor új termet kell kezdenünk!

Megoldás:

1. terem $\{1.\ [1,3),\ 5.\ [3,6),\ 9.\ [6,9),\ 10.\ [10,12)\ \}$
2. terem $\{2.\ [2,4),\ 6.\ [4,7)\ \}$
3. terem $\{3.\ [2,5),\ 8.\ [5,7)\ \}$
4. terem $\{4.\ [2,4),\ 7.\ [4,7)\ \}$

```
kimenet
4
1 5 9 10
2 6
3 8
4 7
```