



## Leállt az értékelő szerver? (balance)

Mozgalmas első versenynap telt el... Miközben a Tudományos Bizottság szűk körben elhárította az értékelő szerver elleni hackertámadást, attól tartanak, hogy ez hatással volt a beküldések pontozására. Csak egy lehetőség van: minden beküldést újra kell értékelni!

Az osztályozó szerver  $N$  processzormaggal rendelkezik. A bizottság már hozzárendelt minden maghoz egy listát  $S$  beküldésből, ahol minden beküldés a verseny  $T$  feladatainak egyikéhez tartozik (a következőként számozva:  $1, \dots, T$ ). A bizottság gondoskodott arról, hogy  $S$  a kettő hatványa legyen.\* Most, a következő  $S$  percben minden mag percenként pontosan egy beküldést fog értékelni a listájáról. Sajnos a feladatok adatait tartalmazó adatbázis meglehetősen sérülékeny és bármikor összeomolhat, amennyiben az egyidejűleg az egy feladat adataihoz intézett kérések száma nagymértékben változik. A bizottság úgy kívánja rendezni az egyes magokhoz történő beküldéseit, hogy az újbóli elbírálás során az egy feladat esetében az egyidejűleg kiértékelendő beküldések maximális és minimális száma legfeljebb eggyel térjen el egymástól.

Írj programot, amely kiszámítja a beküldések magokhoz való rendezett hozzárendelését.

### Bemenet

A bemenet első sora a fent leírt három egész számot tartalmazza:  $N$ ,  $S$  és  $T$ .

Ezután  $N$  sor következik, amely a magokhoz rendelt beküldéseket írja le. A  $i$ -edik sor  $S$  darab egész számot tartalmaz  $t_1, \dots, t_S$  ( $1 \leq t_j \leq T$ ), ami azt jelenti, hogy az  $i$ -edik maghoz a  $t_1, \dots, t_S$  feladathoz rendelt beküldések tartoznak.

### Kimenet

A programnak  $N$  sort kell kiírnia, amelyek a beküldések magokhoz való rendezett hozzárendelését írják le úgy, hogy az egyidejűleg kiértékelt beküldések maximális és minimális száma bármely feladat esetében legfeljebb eggyel térjen el egymástól: ezen sorok közül az  $i$ -ediknek  $S$  darab egész számot  $r_1, \dots, r_S$  kell tartalmaznia, ami azt jelenti, hogy az  $i$ -edik mag a  $j$ -edik percben értékeli ki az  $r_j$  feladatra vonatkozó beküldéseket. Garantált, hogy minden egyes tesztesethez létezik ilyen hozzárendelés.

### Korlátok és értékelés

Minden esetben adott, hogy  $S = 2^k$  néhány pozitív egész  $k$ -ra,  $1 \leq N, S, T \leq 100\,000$  és  $N \cdot S \leq 500\,000$ .

**Részfeladat 1 (10 pont).**  $S = 2$  és  $N, T \leq 20$

**Részfeladat 2 (15 + 5 + 5 pont).**  $S = 2$

**Részfeladat 3 (15 + 5 + 5 pont).**  $N \cdot S \leq 10\,000$

**Részfeladat 4 (20 + 10 + 10 pont).** Nincs további korlátozás.

**Részpontozás.** A 2., a 3. és a 4. részfeladatnál a zárójelben megadott részpontszámokat kaphatod:

- ▶ Az első szám azt a pontszámot jelöli, amelyet akkor kapsz meg, hogy ha minden olyan tesztesetet megoldasz, amelyben  $T \leq N$  és az egyes feladatokhoz benyújtott beküldések száma osztható  $S$ -el.

\* Különben egy hiba miatt† a bírálati rendszer összeomlana, ami az összes tűzfalat lekapcsolná és potenciálisan érzékeny információkat hozna nyilvánosságra!

† Egyetlen munkád volt, Wolfgang!



- ▶ A második szám a további pontok számát jelzi, amelyeket akkor kapsz meg, hogy ha általánosságban véve minden olyan tesztesetet megoldasz, amelyben  $T \leq N$ .
- ▶ A harmadik szám az összes teszteset megoldása esetén kapott többletpontok számát jelöli.

A webes felületen ez a megfelelő részfeladat "Group 1," "Group 2," és "Group 3" számával jelenik meg.

## Példák

Bemenet	Kimenet
3 2 3 1 2 2 3 2 3	2 1 3 2 2 3
3 4 3 2 3 2 2 2 3 3 2 2 2 3 2	2 2 2 3 3 2 3 2 2 3 2 2

Az első példa kimenetében az egyidejűleg kiértékelt beküldések maximális és minimális száma közötti különbség az 1. és a 2. feladatok esetében egy, míg a 3. feladat esetében nulla. Másrészt, ha a beküldéseket a bemenetnek megfelelően rendeznénk, nem lenne érvényes kimenet, mivel a 3. feladatnál az egyidejűleg kiértékelt beküldések maximális és minimális száma közötti különbség kettő. A második példa kimenetében az egyidejűleg kiértékelt beküldések maximális és minimális száma közötti különbség mindhárom feladat esetében nulla.

## Határok

Idő: 2 s

Memória: 1024 MiB