



Börtön

 2 sec.  1024 MB

Alice-t és Bobot igazságtalanul bezárták egy szigorúan őrzött börtönbe. Most a szökésüket tervezik. Ehhez a lehető leghatékonyabban kell kommunikálniuk (pontosabban, Alice minden nap információt szeretne küldeni Bobnak). Sajnos nem tudnak találkozni, így csak szalvétákra írt számokkal tudnak kommunikálni. Alice minden nap egy új információt szeretne küldeni Bobnak – egy számot 0 és $N - 1$ között. Minden délutén Alice fog három szalvétát és mindháromra ír egy-egy számot 0 és $M - 1$ között (a számok lehetnek egyformák), majd otthagyja őket egy széken. Ezután az ellenségük, Charly megsemmisíti az egyik szalvétát és összekeveri a maradék kettőt. Végül Bob elolvassa a maradék két szalvétára írt számokat. Bobnak pontosan vissza kell állítania az eredeti számot, amit Alice küld neki. A szalvétákon kevés a hely, így M rögzített. Alice és Bob célja, hogy maximalizálják az átadható információ mennyiségét, így a lehető legnagyobb N -t választhatják. Segíts Alice-nak és Bobnak azzal, hogy megvalósítasz egy stratégiát mindkettejük számára, mely maximalizálja N értékét.



Megvalósítás

Mivel ez egy kommunikációs probléma, a kódodat két részletben futtatjuk (egyszer mint Alice és egyszer mint Bob). A futtatások nem oszthatnak meg egymással adatot és nem kommunikálhatnak semmilyen más módon, az eddig ismertetett módszeren kívül. Három függvényt kell megvalósítanod:

```
int setup(int M);
```

Ezt a függvényt egyszer hívja meg a program Alice futtatásának kezdetén, egyszer pedig Bob futtatásának kezdetén. A függvény bemenete M , és a kívánt N -t kell visszaadnia. A `setup` függvény mindkét hívásának ugyanazt az N -t kell visszaadnia.

```
std::vector<int> encode(int A);
```

Ez valósítja meg Alice stratégiáját. Bemenete az elkódolandó szám A ($0 \leq A < N$) és három számot kell visszaadnia W_1, W_2, W_3 ($0 \leq W_i < M$) melyek az A számot kódolják. Ez a függvény T alkalommal lesz meghívva – minden naphoz egyszer (ugyanaz A érték több napon is előfordulhat).

```
int decode(int X, int Y);
```

Ez valósítja meg Bob stratégiáját. A bemenete az `encode` függvény által előállított három szám közül kettő, valamilyen sorrendben. Azt az A számot kell visszaadnia, amelyiket az `encode` megkapta. Ez a függvény is T -szer lesz meghívva – az `encode` mind a T hívásához egyszer, azonos sorrendben. Minden `encode` hívás előbb történik meg, mint bármelyik



decode hívás.



Korlátok

- $M \leq 4300$
- $T = 5000$



Pontozás

Egy adott részfeladat esetében a kapott pontok S hányada attól függ, hogy mi a legkisebb N , amit a `setup` függvény az adott részfeladat bármelyik tesztjénél visszaad. Függ továbbá az N^* értéktől is, amely az N célértéke, amire a részfeladat teljes pontszámának megszerzéséhez szükség van:

- Ha a megoldásod bármelyik teszten rossz eredményt ad, akkor $S = 0$.
- Ha $N \geq N^*$, akkor $S = 1.0$.
- Ha $N < N^*$, akkor $S = \max \left(0.35 \max \left(\frac{\log(N) - 0.985 \log(M)}{\log(N^*) - 0.985 \log(M)}, 0.0 \right)^{0.3} + 0.65 \left(\frac{N}{N^*} \right)^{2.4}, 0.01 \right)$.



Részfeladatok

Részfeladat	Pontszám	M	N^*
1	10	700	82017
2	10	1100	202217
3	10	1500	375751
4	10	1900	602617
5	10	2300	882817
6	10	2700	1216351
7	10	3100	1603217
8	10	3500	2043417
9	10	3900	2536951
10	10	4300	3083817



Példa

Tekintsük a következő példát $T = 5$ értékkel. Tekintsünk egy olyan kódolási sémát, amelyben Alice három egyenlő számot küld a 0 kódolásához, vagy három különböző számot az 1 kódolásához. Vegyük észre, hogy Bob az eredeti számot az Alice által küldött három szám közül bármelyikből kettőből dekódolni tudja.



Futtatás	Függvényhívás	Visszatérési érték
Alice	setup(10)	2
Bob	setup(10)	2
Alice	encode(0)	{5, 5, 5}
Alice	encode(1)	{8, 3, 7}
Alice	encode(1)	{0, 3, 1}
Alice	encode(0)	{7, 7, 7}
Alice	encode(1)	{6, 2, 0}
Bob	decode(5, 5)	0
Bob	decode(8, 7)	1
Bob	decode(3, 0)	1
Bob	decode(7, 7)	0
Bob	decode(2, 0)	1



Mintaértékelő

A mintaértékelőben minden encode és decode hívás egy programfuttatásban történik. Ebben a setup függvény csak egyszer kerül meghívásra (ellentétben az értékelő rendszerrel, ahol egyszer-egyszer mindkét program futtatásakor).

A bemenet egyetlen egész szám - M . A mintaértékelő először kiírja az N értéket, amit a setup visszaadott. Ezután T alkalommal meghívja az encode és decode függvényeket véletlenszerűen generált számokkal 0 és $N - 1$ között, és véletlenszerűen generált választásokkal arra vonatkozóan, hogy az encode által generált három szám közül melyik kettőt adja a decode-nak (és milyen sorrendben). A program hibaüzenetet ír ki, ha a megoldásod hibás.