



Task Prison

2 sec. 1024 MB

Այիսն ու Բորը անարդար կերպով դատապարտվել են շատ խիստ պահվող բանտում: Այժմ նրանք պետք է պլանավորեն իրենց փախուստը: Դրա համար նրանք պետք է կարողանան հաղորդակցվել հնարավորինս արդյունավետ (մասնավորապես, Այիսը պետք է ամեն օր տեղեկություններ ուղարկի Բորին): Բայց նրանք չեն կարող հանդիպել և միայն կարող են տեղեկություն փոխանակել անձեռոցիկների վրա գրավոր նշումների միջոցով: Ամեն օր Այիսը ուզում է ուղարկել նոր տեղեկություն Բորին՝ 0 - ի և $N - 1$ - ի միջև թիվ: Ամեն ճաշի ժամանակ Այիսը ստանում է երեք անձեռոցիկներ և յուրաքանչյուր անձեռոցիկի վրա գրում է 0 - ի և $M - 1$ - ի միջև մեկ թիվ (կարող են լինել կրկնություններ) և թողնում դրանք իր նստարանին: Այնուհետև, նրանք թշնամին՝ Չարլին, ոչնչացնում է մեկ անձեռոցիկը և խառնում մնացած երկուսը: Վերջում Բորը գտնում է երկու մնացած անձեռոցիկները և կարդում դրանց վրա գրված թվերը: Նա պետք է ճշգրիտ պարզի անհրաժեշտ տեղեկատվությունը, որը Այիսը ցանկանում էր ուղարկել նրան: Անձեռոցիկների վրա տեղը սահմանափակ է, ուստի M -ը ֆիքսված է: Բայց Այիսի ու Բորի նպատակն է մաքսիմիզացնել ուղարկվող տեղեկատվությունը, այնպես որ նրանք ազատ են ընտրելու N -ը որքան կարող են մեծ: Օգնեք Այիսին և Բորին իրականացնելու այնպիսի ռազմավարություն յուրաքանչյուրի համար, որպեսզի N -ի արժեքը հնարավորինս մեծ լինի:



Իրականացման մանրամասներ

Քանի որ սա հաղորդակցման խնդիր է, ձեր ծրագիրը կաշխատի երկու առանձին գործարկումներով (մեկը Այիսի համար և մեկը Բորի համար), որոնք չեն կարող ունենալ ընդհանուր տվյալներ կամ հաղորդակցվել որևէ այլ կերպ, բացի այստեղ նկարագրվածից: Դուք պետք է իրականացնեք երեք ֆունկցիա.

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

Սա կանչվելու է մեկ անգամ, նախքան ձեր ծրագիրը կաշխատացվի Այիսի համար և մեկ անգամ, նախքան ձեր ծրագիրը կաշխատացվի Բորի համար: Այս ֆունկցիային տրվում է M -ը և ինքը պետք է վերադարձնի ցանկալի N -ը: `setup`-ի երկու կանչերն էլ պետք է վերադարձնեն միևնույն N -ը:

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

Սա իրականացնում է Այիսի ռազմավարությունը: Այն պետք է կոդավորի A ($0 \leq A < N$) թիվը և պետք է վերադարձնի երեք թիվ՝ W_1, W_2, W_3 ($0 \leq W_i < M$), որոնք կոդավորում են A -ն: Այս ֆունկցիան կանչվելու է ընտրամենը T անգամ՝ ամեն օր մեկ անգամ (A -ի արժեքները ինչ որ օրերում կարող են կրկնվել):

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



Սա իրականացնում է Բորի ռազմավարությունը: Այն կանչվելու է encode-ի վերադարձրած թվերից երկուսի վրա, որոնք տրվելու են կամայական հերթականությամբ: Այն պետք է վերադարձնի նույն A արժեքը, որը encode-ը ստացել է: Այս ֆունկցիան նույնպես կանչվելու է T անգամ՝ encode-ի T կանչերին համապատասխան, նրանք լինելու են նույն հերթականությամբ: encode-ի բոլոր կանչերը նախորդելու են decode-ի բոլոր կանչերին:



Սահմանափակումներ

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



Գնահատումը

Յուրաքանչյուր ենթախնդրի համար, դուք կստանաք որպես միավոր մի S կոտորակ, որը կախված է setup-ի կողմից այդ ենթախնդրի բոլոր թեստերի համար վերադարձված փոքրագույն N -ից: Այն նաև կախված է N^* -ից, որը N -ի նպատակային արժեքն է, որի դեպքում տրվում է տվյալ ենթախնդրի համար նախատեսված միավորն ամբողջությամբ:

- Եթե ձեր ծրագիրը որևէ թեստի համար սխալ է աշխատում, ապա $S = 0$:
- Եթե $N \geq N^*$, ապա $S = 1.0$:
- Եթե $N < N^*$, ապա $S = \max\left(0.35 \max\left(\frac{\log(N) - 0.985 \log(M)}{\log(N^*) - 0.985 \log(M)}, 0.0\right)^{0.3}, 0.65 \left(\frac{N}{N^*}\right)^{2.4}, 0.01\right)$:



Ենթախնդիրներ

Ենթախնդիր	Միավոր	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



Օրինակ

Դիտարկենք հետևյալ օրինակը, որտեղ $T = 5$: Այստեղ մենք ունենք կոդավորման այսպիսի սխեմա. Եթե Ալիսն ուղարկում է երեք հատ նույն թիվը, ապա նա կոդավորում է 0 թիվը, իսկ եթե երեք տարբեր թվեր, ապա նա կոդավորում է 1 թիվը: Նկատենք, որ Բոբը կարող է ապակոդավորել սկզբնական թիվը Ալիսի ուղարկած երեք թվերից ցանկացած երկուսի միջոցով:

Execution	Function call	Return value
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

Գրեյդերի նմուշը

Գրեյդերի նմուշի համար encode-ի և decode-ի բոլոր կանչերը լինելու են ձեր ծրագրի միևնույն կատարման ժամանակ: Նաև setup-ը կանչվելու է միայն մեկ անգամ (հակառակ իսկական գրեյդերի, որի դեպքում յուրաքանչյուր կատարման համար մեկ անգամ է կանչվում):

Մուտքում տրվում է միայն մեկ թիվ՝ M -ը: Ապա այն կտալի ձեր setup-ի վերադարձրած N -ը: Հետո այն կանչելու է encode և decode ֆունկցիաները այս հերթականությամբ T անգամ պատահական գեներացված 0-ից $N - 1$ սահմաններում գտնվող թվերով, նաև պատահական է որոշվում, թե encode-ի տված երեք թվերից որ երկուսը և ինչ հերթականությամբ տրվեն decode-ին: Այն կտալի հաղորդագրությունը սխալի մասին, եթե ձեր ծրագիրը սխալ աշխատի: