



## Задача Prison

 2 sec.  1024 MB

Алис и Боб са несправедливо осъдени на затвор с максимална сигурност. Сега те трябва да планират бягството си. За да направят това, те трябва да могат да комуникират възможно най-ефективно (по-специално, Алис трябва да изпраща ежедневна информация на Боб). Те обаче не могат да се срещат и могат да обменят информация само чрез бележки, написани на салфетки. Всеки ден Алис иска да изпрати нова информация на Боб - число между 0 и  $N - 1$ . На всеки обяд Алис взема три салфетки и пише число между 0 и  $M - 1$  на всяка салфетка (възможно е да има повторения) и ги оставя на стола си. След това техният враг, Чарли, унищожава една от салфетките и евентуално разбърква другите две. Накрая Боб намира останалите две салфетки и прочита числата върху тях. Той трябва точно да декодира оригиналното число, което Алис е искала да му изпрати. Мястото върху салфетките е ограничено, така че  $M$  е фиксирано. Целта на Алис и Боб е да увеличат максимално количеството информация, което могат да си предават, така че те са свободни да избират  $N$  колкото е възможно по-голямо. Помогнете на Алис и Боб, като приложите стратегия за всеки от тях, опитвайки се да увеличите максимално стойността на  $N$ .



### Детайли по имплементацията

Тъй като това е интерактивна задача, програмата ви ще бъде изпълнена в два отделни процеса (един за Алис и един за Боб), които не могат да споделят данни или да комуникират по друг начин освен описания тук. Трябва да имплементирате три функции:

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

Тази функция ще бъде извикана веднъж в началото на изпълнението на вашата програма от Алис и веднъж в началото на изпълнението на Боб. Дадено е  $M$  и трябва да върнете желаното  $N$ . И двете извиквания на `setup` трябва да върнат едно и също  $N$ .

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

Тази функция реализира стратегията на Алис. Тя ще бъде извикана с числото за кодиране, което е  $A$  ( $0 \leq A < N$ ) и трябва да върне три числа  $W_1, W_2, W_3$  ( $0 \leq W_i < M$ ), които кодират  $A$ . Тази функция ще бъде извикана общо  $T$  пъти - веднъж на ден (стойностите на  $A$  могат да се повтарят между дните).

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



Тази функция реализира стратегията на Боб. Ще бъде извикана с две от трите числа, върнати от `encode`, в произволен ред. Трябва да върне същата стойност  $A$ , която е получила `encode`. Тази функция ще бъде извикана  $T$  пъти - отговарящи на вече направените  $T$  извиквания на `encode`, като те ще бъдат в същия ред. Всички извиквания на `encode` ще се случат преди всички извиквания на `decode`.



## Ограничения

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



## Оценяване

За дадена подзадача, частта  $S$  от точките, които получавате, зависи от най-малката стойност на  $N$ , върната от `setup` за всеки тест в тази подзадача. Тя зависи също от  $N^*$ , което е най-малката стойност на  $N$ , необходима за получаване на пълния брой точки за подзадачата:

- Ако решението ви не е успешно за някой от тестовете, то  $S = 0$ .
- Ако  $N \geq N^*$ , то  $S = 1.0$ .
- Ако  $N < N^*$ , то  $S = \max \left( 0.35 \max \left( \frac{\log(N) - 0.985 \log(M)}{\log(N^*) - 0.985 \log(M)}, 0.0 \right)^{0.3} + 0.65 \left( \frac{N}{N^*} \right)^{2.4}, 0.01 \right)$ .



## Подзадачи

Подзадача	Точки	$M$	$N^*$
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



## Пример

Да разгледаме следния пример за  $T = 5$ . Разглеждаме схема на кодиране, при която Алис изпраща три равни числа, за да кодира 0, или три различни числа, за да кодира 1. Обърнете внимание, че Боб може да декодира оригиналното число от всеки две от трите числа (без значение на реда им), изпратени от Алис.

Изпълнение от	Извикана функция	Върната стойност
Алис	<code>setup(10)</code>	2
Боб	<code>setup(10)</code>	2
Алис	<code>encode(0)</code>	{5, 5, 5}
Алис	<code>encode(1)</code>	{8, 3, 7}
Алис	<code>encode(1)</code>	{0, 3, 1}
Алис	<code>encode(0)</code>	{7, 7, 7}
Алис	<code>encode(1)</code>	{6, 2, 0}
Боб	<code>decode(5, 5)</code>	0
Боб	<code>decode(8, 7)</code>	1
Боб	<code>decode(3, 0)</code>	1
Боб	<code>decode(7, 7)</code>	0
Боб	<code>decode(2, 0)</code>	1



## Примерен грейдър

За примерния грейдър, всички извиквания на `encode` и `decode` ще бъдат в едно и също изпълнение на вашата програма. Освен това, `setup` ще бъде извикана само веднъж (за разлика от двата пъти, по веднъж за изпълнение на процесите на Алис и на Боб, както е в системата за оценяване).

Входът е единствено цяло число –  $M$ . След това ще се отпечата  $N$ , върнато от вашата функция `setup`. След това ще се извикат `encode` и `decode`  $T$  пъти със случайно генерирани числа от 0 до  $N-1$  и случайно генерирани избори кои две от трите числа, върнати от `encode` да даде на `decode` (и в какъв ред). Ще отпечата съобщение за грешка, ако решението ви е неуспешно.