



ამოცანა ციხე

 2 sec.  1024 MB

ალისა და ბობი უსამართლოდ გამოამწყვდიეს მაქსიმალურად დაცულ ციხეში. ახლა მათ უნდა შეადგინონ გაქცევის გეგმა. ამის მისაღწევად მათ უნდა მოახერხონ ერთმანეთთან რაც შეიძლება ეფექტური კომუნიკაციის დამყარება (კონკრეტულად, ალისამ უნდა გაუგზავნოს ბობს ინფორმაცია ყოველდღიურად). მათ არ შეუძლიათ ერთმანეთთან შეხვედრა და ინფორმაციას ცვლიან მხოლოდ ხელსახოცებზე დაწერილი მესიჯების გამოყენებით. ალისას უნდა, რომ ბობს ყოველ დღე გაუგზავნოს ახალი ინფორმაცია – რიცხვი 0 -დან $N - 1$ -მდე. ყოველ სადილზე ალისა ღებულობს 3 ახალ ხელსახოცს და წერს თითოეულ მათგანზე 0 -დან $M - 1$ -მდე რიცხვს (არა აუცილებლად განსხვავებულ რიცხვებს) და ტოვებს ხელსახოცებს თავის სკამზე. შემდეგ მათი მოწინააღმდეგე, ჩარლი, ანადგურებს ხელსახოცებიდან ერთ-ერთს და ერთმანეთში ურევს დარჩენილ ორს. საბოლოოდ, ბობი პოულობს სკამზე 2 ხელსახოცს და კითხულობს მათზე რიცხვებს. მან ზუსტად უნდა გაარკვიოს იმ რიცხვის მნიშვნელობა, რომლის გაგზავნაც ალისას სურდა. ხელსახოცებზე სივრცე შეზღუდულია, ამიტომ M არის ფიქსირებული. თუმცა, ალისას და ბობის მიზანია ინფორმაციის გამტარუნარიანობის მაქსიმიზაცია, რომ მათ შეძლონ რაც შეიძლება დიდი N -ის გაგზავნა. დაეხმარეთ ალისას და ბობს თითოეული მათგანისთვის სტრატეგიის იმპლემენტაციაში, რათა მაქსიმალურად გაზარდონ N -ის მნიშვნელობა.

იმპლემენტაციის დეტალები

რადგან ეს კომუნიკაციის ამოცანაა, თქვენი პროგრამა შესრულდება ორ ცალკეულ გამოცახებაში (ერთი ალისასთვის და მეორე ბობისთვის), რომლებსაც არ შეუძლიათ მონაცემების გაზიარება ან კომუნიკაცია ქვემოთ აღწერილის გარდა სხვა გზით. თქვენ იმპლემენტაცია უნდა გაუკეთოთ სამ ფუნქციას:

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

ეს ფუნქცია გამოიძახება ერთხელ ალისას სტრატეგიის შესრულებამდე და ერთხელ ბობის სტრატეგიის შესრულებამდე. მას ეძლევა M და უნდა დააბრუნოს სასურველი N , რომლის გადაცემასაც შეეცდება ალისა ბობისთვის. `setup`-ის ორივე გამოცახებამ უნდა დააბრუნოს ერთი და იგივე N .

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

ეს ფუნქცია იმპლემენტაციას უკეთებს ალისას სტრატეგიას. ის გამოიძახება, ბობისთვის A ($0 \leq A < N$) რიცხვის გადასაგზავნად და უნდა დააბრუნოს სამი რიცხვი W_1, W_2, W_3 ($0 \leq W_i < M$), რომლებშიც კოდირებულია A რიცხვი. ეს ფუნქცია გამოიძახება სულ T -ჯერ – დღეში ერთხელ (A -ს მნიშვნელობები შეიძლება განმეორდეს დღეებს შორის).



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

ეს ფუნქცია იმპლემენტაციას უკეთებს ბობის სტრატეგიას. გამოძახებისას ამ ფუნქციას გადმოეცემა encode-ს მიერ დაბრუნებული სამი რიცხვიდან ორი, რაღაც თანმიმდევრობით. მან უნდა დააბრუნოს იგივე მნიშვნელობა A , რაც encode-მა მიიღო. ეს ფუნქცია ასევე გამოიძახება T -ჯერ – encode-ის T ცალი გამოძახების შესაბამისად და იმავე თანმიმდევრობით. encode-ის ყველა გამოძახება მოხდება decode-ის ყველა გამოძახებამდე.

შეზღუდვები

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

შეფასება

კონკრეტული ქვეამოცანისთვის, მიღებული ქულების S წილი დამოკიდებულია setup-ის მიერ ამ ქვეამოცანაში ყველა ტესტს შორის დაბრუნებულ უმცირეს N -ზე. ეს ასევე დამოკიდებულია N^* -ზე, რომელიც წარმოადგენს N -ის სამიზნე მნიშვნელობას, რომელიც გჭირდებათ ქვეამოცანისთვის სრული ქულების მისაღებად:

- თუ თქვენი ამოხსნა ვერ შეასრულებს რომელიმე ტესტს, მაშინ $S = 0$.
- თუ $N \geq N^*$, მაშინ $S = 1.0$.
- თუ $N < N^*$, მაშინ $S = \max\left(0.35 \max\left(\frac{\log(N) - 0.985 \log(M)}{\log(N^*) - 0.985 \log(M)}, 0.0\right)^{0.3} + 0.65 \left(\frac{N}{N^*}\right)^{2.4}, 0.01\right)$.

ქვეამოცანები

ქვეამოცანა	ქულა	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



მაგალითი

განვიხილოთ შემდეგი მაგალითი, სადაც $T = 5$. აქ გვაქვს კოდირების სქემა, სადაც ალისა 0-ის კოდირებისთვის სამ თანაბარ რიცხვს აგზავნის ან 1-ის კოდირებისთვის სამ განსხვავებულ რიცხვს. შევნიშნოთ, რომ ბობს შეუძლია ორიგინალი რიცხვის გაშიფვრა ალისის მიერ გაგზავნილი სამი რიცხვიდან ნებისმიერი ორიდან.

შესრულებელი	ფუნქციის გამოძახება	დაბრუნებული მნიშვნელობა
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



სანიმუშო გრეიდერი

სანიმუშო გრეიდერისთვის, encode-სა და decode-ს ყველა გამოძახება თქვენი პროგრამის ერთსა და იმავე შესრულებაში განხორციელდება. გარდა ამისა, setup გამოიძახება მხოლოდ ერთხელ (განსხვავებით ორჯერ, ერთხელ შესრულებისას, როგორც შეფასების სისტემაში).

შესატანი მონაცემი არის მხოლოდ ერთი მთელი რიცხვი - M . შემდეგ ის დაბეჭდავს თქვენი setup-ის მიერ დაბრუნებულ N -ს. შემდეგ ის გამოიძახებს encode და decode ფუნქციებს ამ თანმიმდევრობით T ჯერ, შემთხვევით გენერირებული რიცხვებით 0-დან $N-1$ -მდე და შემთხვევითად ირჩევს, encode-ს დაბრუნებული სამი რიცხვიდან რომელი ორი უნდა გადასცეს decode-ს (და რა თანმიმდევრობით). ის დაბეჭდავს შეცდომის შეტყობინებას, თუ თქვენი ამოხსნა ვერ ამოხსნის მოცემულ ტესტს.