

Zadatak 3. Zec

Ludi mađioničar je upravo izgubio svog najdražeg zeca (Bijelog Zeca, naravno) negdje u redu od N kaveza i pokušava ga pronaći. Kavezi su numerisane cijelim brojevima od 1 do N . U početku zec je u nekom nepoznatom kavezu i svaka sekunda mađioničara izgleda ovako:

1. Prvo izabere jedan kavez u nizu i provjeri ga. Ovo nazivamo **provjereni kavez**. Ako je zec tu, potraga prestaje.
2. Nakon toga, zec bira da ostaje u istom kavezu, ili da ode do sljedećeg kaveza (u lijevu ili desnu stranu). **Primijetite da je moguće da zec ode do provjerenog kaveza, ako su jedan do drugog. To ne zaustavlja pretragu!**

Izbor zeca je determinističan na osnovu njegovog raspoloženja. Tačnije, zec ima dva raspoloženja:

1. **Prestrašen** – kada je zec ovog raspoloženja, on se kreće **dalje od provjerenog kaveza**. Ako se ne može kretati dalje (tj, zec je u ćeliji 1 ili N), zec ostaje u istom mjestu.
2. **Znatiželjan** – u ovom raspoloženju zec se kreće **prema provjerenom kavezu**. Primijetite da je uvijek moguće da se zec približuje.

Također, zec reaguje samo na zadnji provjereni kavez, ne zanima ga stanje prošlih provjerenih kaveza.

Pošto je to mađioničarev najdraži zec, on veoma dobro zna njegovo raspoloženje. Tačnije, on zna da zec mijenja raspoloženje svakih S sekundi iz *prestrašen* and C sekundi iz *znatiželjan*. Na primjer, ako je $S = 2$ i $C = 1$, raspoloženje zeca bi bila sekvenca:

[*prestrašen, prestrašen, znatiželjan, prestrašen, prestrašen, znatiželjan ...*].

Mađioničar je veoma zabrinut za svog zeca i zamolio vas je da napišete program `rabbit.cpp` koji računa sekvencu kaveza za provjeriti, tako da zec, bez obzira na njegovu inicijalnu poziciju, je garantovan da bude nađen.

Ulaz

S prve linije standardnog ulaza vaš program treba pročitati 3 broja: N , S i C , koji predstavljaju broj kaveza i ponašanje raspoloženja zeca.

Izlaz

Na prvoj liniji standardnog izlaza vaš program treba ispisati K , broj sekundi koliko vaša sekvenca traje. Na drugoj liniji vaš program treba ispisati K brojeva u intervalu $[1, N]$, ispisujući sve kaveze koji se trebaju provjeriti svake sekunde. Primijetite da smijete više puta provjeriti isti kavez.

Bodovanje

Nek K bude broj sekundi u vašoj sekvenci. Ako pokušate provjeriti nepostojeći kavez (tj, van intervala $[1, N]$), ili vaša sekvenca ne bi našla zeca uvijek, dobićete 0 bodova za taj odgovor i *Wrong Answer* kao objašnjenje. U suprotnom, ako test dadne R bodova, dobićete pR bodova gdje:

- $p = 0$, ako je $K > N$
- $p = 1$, ako je $K \leq T$ (T je dato na sljedećoj strani)
- $p = 0.3 \left(\frac{T}{K}\right)^2$, u suprotnom (T je dato na sljedećoj strani)

Gdje je: $T = \frac{N(S+C)}{S+2 \max(S,C)} + 3 \max(S, C)$

Ograničenja

$$2 \leq N \leq 10^4$$

$$0 \leq S, C \leq 50$$

Informacije o testovima

- Za 8% testova $S = 0, C = 1$
- Za 12% testova $S = 1, C = 0$
- Za 8% testova $S = 1, C = 1$

Primjer

Ulaz	Izlaz
12 2 1	14
	2 5 3 2 6 1 2 11 12 12 8 10 12 6

Objašnjenje

Možemo to testirati za bilo koju poziciju ova sekvenca će uvijek naći zeca. Na primjer, uzmimo primjer gdje zec kreće od kaveza 8. Pretraga će ići ovako:

Sekunda	Provjereni kavez	Raspoloženje zeca (Prije kretanja)	Kretanje zeca
1	2	Prestrašen	8 -> 9
2	5	Prestrašen	9 -> 10
3	3	Znatiželjan	10 -> 9
4	2	Prestrašen	10 -> 11
5	6	Prestrašen	11 -> 12
6	1	Znatiželjan	12 -> 11
7	2	Prestrašen	11 -> 12
8	11	Prestrašen	12 -> 12
9	12	Znatiželjan	Nađen

Za ovo rješenje imamo $K > T$, jer $K = 14$ i $T = 12$.

Tako da, za ovaj dio, broj bodova za ovaj test bi bio $p \approx 0.22$.