

COVID testovi

U Fabijanovoj školi došlo je do novog vala COVID epidemije. Kako bi se spriječilo daljnje širenje, škola je odlučila testirati sve učenike korištenjem antigenskih testova s uzorcima sline učenika.

Budući da su svi učitelji davno zaboravili kako koristiti te testove, Fabijan se prijavio kao volonter za pomoć pri testiranju.

Dobio je uzorke sline od N učenika (i možda koje voditeljice ekipe, no to je sad irelevantno za ovaj zadatak). Zbog GDPR-a, učenici su označeni brojevima od 0 do $N - 1$ i njegov zadatak je odrediti koji su uzorci pozitivni. Iz lokalnih statistika zna da je vjerojatnost da je neki učenik pozitivan jednaka P , a da su vjerojatnosti za različite učenike međusobno neovisne (pogledajte odjeljak Ulazni podatci za formalniji opis). Nažalost, već je bilo prekasno kad je shvatio da je testiranje svih učenika izuzetno dug i dosadan zadatak. Međutim, shvatio je da proces testiranja može obaviti na pametniji način nego testiranjem uzoraka redom. Ako pomiješa podskup uzoraka i testira tu smjesu, saznat će jesu li svi uzorci u toj smjesi negativni ili je barem jedan od njih pozitivan. To bi mogao iskoristiti kako bi smanjio broj potrebnih testova!

Budući da ima dovoljno sline u svakom uzorku, može testirati uzorak koliko god puta želi. Nadalje, testovi su potpuno precizni, tako da se nikad ne događa da različiti testovi daju različite rezultate za istu osobu.

Pod tim uvjetima, želio bi optimizirati proces kako bi iskoristio što manje testova. Međutim, on je zauzet testiranjem, pa je optimizacija procesa na vama.

Interakcija

Ovaj zadatak je interaktivan.

Vaš program će se izvršavati na određenom broju testnih slučajeva. Kao dio jednog testnog slučaja, i stoga tijekom jednog izvršavanja vašeg programa, morat ćete riješiti T različitih problema. Vrijednosti N i P su iste za sve probleme, ali koji su učenici pozitivni bit će (vjerojatno) različiti u svakom problemu.

Možete implementirati potrebni protokol sami ili koristiti predložak. Predložak možete pronaći u CMS-u kao privatak zadatka pod nazivom `template.cpp`.

Protokol

Prvo, vaš program treba pročitati redak sa standardnog unosa koji sadrži cijeli broj N , realni broj P i cijeli broj T odvojene razmacima — redom broj ljudi, vjerojatnost pozitivnog uzorka i broj problema.

Nakon toga, program može ispisivati upite na standardni izlaz. Svaki upit treba biti jedan redak koji sadrži Q , razmak i niz s duljine N , gdje je $s_i = 1$ ako Fabijan treba dodati i -tu osobu u test, a 0 inače. Nakon ispisa ovog retka, program treba isprazniti standardni izlaz i zatim pročitati jedan znak u jednom retku koji će biti P ako je barem jedna osoba iz testirane skupine pozitivna, a N inače.

Program također mora ispisati odgovor kao jedan redak na standardni izlaz koji sadrži A , razmak i niz s duljine N , gdje je $s_i = 1$ ako je i -ta osoba pozitivna, a 0 inače. Nakon ispisa ovog retka, program treba isprazniti standardni izlaz i zatim pročitati jedan znak u jednom retku.

Ako redak sadrži C , tada je vaš odgovor bio točan. U tom slučaju, program može početi s izvođenjem upita o sljedećem problemu ili ako je ovo bio vaš T -ti odgovor, završiti.

Ako redak sadrži W , tada vaš odgovor nije bio točan. U tom slučaju program treba odmah završiti.

Molimo imajte na umu da je završavanje nakon W važno za dobivanje točne povratne informacije od CMS-a. Ako vaš program nastavi s radom, može se srušiti ili dobiti neku drugu neuspješnu presudu.

Predložak

Ako koristite implementaciju protokola u `template.cpp`, trebate implementirati funkciju `std::vector<bool> find_positive()`. Ova funkcija će biti pozvana jednom za svaki problem. Mora vratiti vektor Booleovih vrijednosti duljine N , gdje je i -ti element `true` ako je i -ti učenik pozitivan.

Možete koristiti funkciju `bool test_students(std::vector<bool> mask)` da biste odredili tko je pozitivan. Ova funkcija obavlja test na podskupu uzoraka. Njezin jedini argument je vektor `bool` vrijednosti duljine N , gdje je i -ti element `true` ako i -ti uzorak treba biti dodan u smjesu. Vraća `true` ako (i samo ako) je barem jedan od uzoraka u smjesi pozitivan.

Također možete koristiti globalne varijable N i P koje sadrže N i P iz teksta zadatka. Možete obaviti bilo kakvu inicijalizaciju u funkciji `main` nakon prvog poziva `scanf`.

Ulazni podatci

Sudac za zadatak nije adaptivan, što znači da se pozitivnost pojedinih učenika određuje prije pokretanja vašeg programa. Nadalje, hoće li bilo koji učenik biti pozitivan, određeno je neovisno s vjerojatnošću P koristeći poštenu generator slučajnih brojeva.

Podzadaci i bodovanje

Postoje dva podzadatka.

Prvi podzadatak (10 bodova)

- $N = 1\,000$
- $T = 1$
- $0 \leq P \leq 1$

Rješenje je prihvaćeno ako odgovori točno i koristi najviše $2 \cdot N$ upita u svakom testnom slučaju.

Drugi podzadatak (90 bodova)

- $N = 1\,000$
- $T = 300$
- $0.001 \leq P \leq 0.2$

Ovaj podzadatak koristi parcijalno bodovanje.

Ako je vaš odgovor na bilo koji problem netočan, dobit ćete nula bodova. Inače, broj bodova za određeni testni slučaj bit će određen na temelju prosječnog broja upita po problemu. Općenito, manji broj upita donosi veći broj bodova. Neka Q označava prosječni broj upita koje je vaš program koristio u svim problemima, zaokružen na jedno decimalno mjesto prema dolje. Za svaki testni slučaj, izračunali smo vrijednost F (vidi dolje). Broj bodova za određeni testni slučaj bit će izračunat prema sljedećim pravilima:

- Ako $Q > 10 \cdot F$ dobit ćete 0 bodova (pogrešan odgovor).
- Ako je $F < Q \leq 10 \cdot F$, broj bodova će biti određen sljedećom formulom:

$$90 \cdot \frac{F}{F + 4 \cdot (Q - F)}$$

- Ako je $Q \leq F$, dobit ćete punih 90 bodova.

Vaše rješenje će biti ocjenjivano na nekoliko testnih slučajeva s različitim vrijednostima P . Ukupan broj bodova koje ćete dobiti bit će minimalni broj bodova u svim testnim slučajevima (tj. kroz sve vjerojatnosti P).

Postoje sljedeći testni slučajevi:

| P | F |
|----------|-------|
| 0.001 | 15.1 |
| 0.005256 | 51.1 |
| 0.011546 | 94.9 |
| 0.028545 | 191.5 |
| 0.039856 | 246.3 |
| 0.068648 | 366.2 |
| 0.104571 | 490.3 |
| 0.158765 | 639.1 |
| 0.2 | 731.4 |

Sustav za ocjenjivanje će pružiti povratne informacije za svaki testni slučaj. Ove povratne informacije će uključivati vrijednost Q vašeg rješenja za svaki testni slučaj gdje ste dobili broj bodova različit od nula.

Primjer interakcije

Evo uzorka interakcije s ocjenjivačem. Ne zaboravite isprazniti izlaz nakon svakog retka.

| Vaš unos | Vaš izlaz |
|----------|--------------|
| 10 0.4 2 | |
| | Q 1000000000 |
| P | |
| | Q 0000001000 |
| P | |
| | Q 0000000001 |
| P | |
| | Q 0111110110 |
| N | |
| | A 1000001001 |
| C | |
| | A 0000000000 |
| W | |

Program je točno riješio prvi problem, ali ne i drugi, jer je točno rješenje bilo 1100010010 (što program nije mogao znati, jer nije izveo nikakve upite). Čak i ako je postojao još jedan problem, program bi trebao odmah završiti.