

Sõnum

Aisha ja Basma on kaks sõpra, kes suhtlevad omavahel kirja teel. Aishal on sõnum M , mis on S bitist (s.o nullidest või ühtedest) koosnev jada, mida ta soovib Basmale saata. Aisha suhtleb Basmaga, saates talle **pakette**. Pakett on 31 bitist koosnev jada, mille indeksid on vahemikus 0 kuni 30. Aisha soovib saata sõnumi M Basmale, saates talle teatud arvu pakette.

Kahjuks segab Cleopatra Aisha ja Basma suhtlusele vahele ning suudab pakette **rikkuda**. See tähendab, et igas pakettis võib Cleopatra muuta täpselt 15 bitikohta. Täpsemalt on olemas massiiv C pikkusega 31, kus iga element on kas 0 või 1:

- $C[i] = 1$ näitab, et Cleopatra saab muuta bitti indeksiga i . Neid indekseid nimetame Cleopatra poolt **kontrollitavateks**.
- $C[i] = 0$ näitab, et bitti indeksiga i Cleopatra muuta ei saa.

Massiivis C on täpselt 15 ühte ja 16 nulli. Sõnumi M saatmisel jääb Cleopatra kontrollitavate indeksite komplekt kõigi pakettide jaoks samaks. Aisha teab täpselt, millised 15 indeksit on Cleopatra kontrolli all. Basma teab ainult seda, et Cleopatra kontrollib 15 indeksit, kuid ta ei tea, millised need on.

Olgu A pakett, mille Aisha otsustab saata (nimetame seda **algseks paketiks**). Olgu B pakett, mille Basma saab (nimetame seda **rikutud paketiks**). Iga i puhul, kus $0 \leq i < 31$:

- kui Cleopatra ei kontrolli bitti indeksiga i ($C[i] = 0$), saab Basma biti i nii, nagu Aisha selle saatis ($B[i] = A[i]$),
- kui Cleopatra kontrollib bitti indeksiga i ($C[i] = 1$), otsustab Cleopatra biti i väärtuse.

Kohe pärast iga paketi saatmist saab Aisha teada, milline vastav rikutud pakett on.

Pärast seda, kui Aisha on kõik paketid saatnud, saab Basma kõik rikutud paketid **samas järjekorras, nagu need saadeti**, ja peab taastama algse sõnumi M .

Sinu ülesandeks on välja töötada ja rakendada strateegia, mis võimaldaks Aishal saata sõnumi M Basmale nii, et Basma suudab rikutud pakettidest M taastada. Täpsemalt peaksid sa kirjutama kaks funktsiooni. Esimene funktsioon teostab Aisha tegevusi. Sellele antakse sõnum M ja massiiv C ning see peaks saatma paketid, et edastada sõnum Basmale. Teine funktsioon teostab Basma tegevusi. Sellele antakse rikutud paketid ja see peaks taastama algse sõnumi M .

Realisatsioon

Esimene funktsioon, mille sa peaksid kirjutama, on:

```
void send_message(std::vector<bool> M, std::vector<bool> C)
```

- M on S -elemendiline massiiv, milles on Aisha sõnum, mida ta tahab Basmale saata.
- C on 31-elemendiline massiiv, mis näitab indekseid, mida Cleopatra kontrollib.
- Seda funktsiooni võidakse igas testis välja kutsuda **ülimalt** 2100 korda.

See funktsioon peaks pakettide saatmiseks kasutama järgmist funktsiooni:

```
std::vector<bool> send_packet(std::vector<bool> A)
```

- A on algne pakett (31-elemendiline massiiv), mis sisaldab Aisha saadetud bitte.
- See funktsioon tagastab rikutud paketi B , mis sisaldab Basma saadud bitte.
- Seda funktsiooni võib igal `send_message` väljakutsel kasutada ülimalt 100 korda.

Teine funktsioon, mille sa peaksid kirjutama, on:

```
std::vector<bool> receive_message(std::vector<std::vector<bool>> R)
```

- R on massiiv, mis kirjeldab rikutud pakette. Paketid pärinevad ühe `send_message` väljakutse raames Aisha saadetud pakettidest ja need antakse **samas järjekorras, milles Aisha need saatis**. Iga R element on 31-elemendiline massiiv, mis esintab üht rikutud paketti.
- See funktsioon peaks tagastama S -elemendilise massiivi, mis on võrdne algse sõnumiga M .
- Seda funktsiooni võidakse igas testis välja kutsuda **mitu korda, täpselt üks kord** iga `send_message` väljakutse kohta. Funktsiooni `receive_message` **väljakutsumiste järjekord** ei tarvitse olla sama, mis vastavate `send_message` väljakutsumiste järjekord.

Pane tähele, et hindamissüsteemis kutsutakse `send_message` ja `receive_message` funktsioone välja kahest eraldiseisvast programmist.

Piirangud

- $1 \leq S \leq 1024$,
- C sisaldab täpselt 31 elementi, millest 16 on väärtusega 0 ja 15 on väärtusega 1.

Alamülesanded ja hindamine

Kui mõnes testis ei vasta `send_packet` funktsiooni väljakutsed ülaltoodud reeglitele või kui mõni `receive_message` tagastatud väärtus on vale, on selle testi skoor 0.

Vastasel juhul, olgu Q maksimaalne `send_packet` funktsiooni väljakutsumiste arv üle kõigi `send_message` väljakutsete kõigis testides. Samuti olgu X võrdne:

- 1, kui $Q \leq 66$,
- 0.95^{Q-66} , kui $66 < Q \leq 100$

Seejärel arvutatakse tulemus järgmiselt:

Alamülesanne	Tulemus	Lisapiirangud
1	$10 \cdot X$	$S \leq 64$.
2	$90 \cdot X$	Lisapiirangud puuduvad.

Pange tähele, et mõnel juhul võib hindaja käitumine olla **adaptiivne**. See tähendab, et `send_packet` poolt tagastatud väärtused sõltuvad mitte ainult selle sisendargumentidest, vaid ka paljudest muudest asjadest, sealhulgas varasemad selle funktsiooni väljakutsete sisendid ja väljundid ning hindaja poolt genereeritud pseudojuhuslikud arvud. Hindaja on **deterministlik** selles mõttes, et kui seda kaks korda välja kutsuda ja mõlemal juhul samad paketid saata, tehakse nendesse samasugused muudatused.

Näide

Vaatame järgmist väljakutset.

```
send_message([0, 1, 1, 0],  
             [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,  
             1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1])
```

Sõnum, mida Aisha püüab Basmale saata, on $[0, 1, 1, 0]$. Cleopatra ei saa muuta bitti indeksiga 0 kuni 15, samas kui ta saab muuta bitti indeksiga 16 kuni 30.

Näite huvides oletame, et Cleopatra täidab järjestikused kontrollitavad bitid vahelduvate nullide ja ühtedega, s.t. ta määrab nulli esimesele kontrollitavale indeksile (meie näites indeks 16), ühe teisele kontrollitavale indeksile (indeks 17) ja nii edasi.

Aisha võib otsustada saata kaks algse sõnumi bitti ühes pakettis järgmiselt: ta täidab esimesed 8 enda kontrollitavat bitti esimese saadetava bitiga ning järgmised 8 enda kontrollitavat bitti teise saadetava bitiga.

Seejärel saadab Aisha järgmise paketi:

```
send_packet([0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1,  
            0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0])
```

Pange tähele, et Cleopatra võib muuta bittide viimaste 15 indeksiga, seega võib Aisha need suvaliselt määrata, kuna need võidakse üle kirjutada. Cleopatra eeldatava strateegia järgi tagastab funktsioon järgmise massiivi:

[0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0].

Aisha otsustab saata sõnumi M viimased kaks bitti teises pakettis samamoodi nagu enne:

```
send_packet([1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0,
             0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0])
```

Cleopatra eeldatava strateegia järgi tagastab funktsioon järgmise massiivi:

[1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0].

Aisha võib saata rohkem pakette, kuid ta otsustab seda mitte teha.

Hindaja kutsub seejärel funktsiooni `receive_message` välja järgmiselt:

```
receive_message([[0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1,
                  0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0],
                 [1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0,
                  0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0]])
```

Basma taastab sõnumi M järgmiselt. Iga paketi puhul võtab ta esimese biti, mis esineb kaks korda järjest, ja viimase biti, mis esineb kaks korda järjest. See tähendab, et esimesest pakettist võtab ta bitid [0, 1] ja teisest pakettist bitid [1, 0]. Kokku pannes taastab ta sõnumi [0, 1, 1, 0], mis on õige vastus selle `receive_message` väljakutse jaoks.

Saab näidata, et Cleopatra eeldatava strateegia korral ja sõnumite pikkusega 4 suudab Basma sõnumi M selle lähenemisviisiga õigesti taastada, sõltumata C väärtusest. Samas kõigil juhtudel ei ole see lähenemine korrektne.

Näidishindaja

Näidishindaja ei ole adaptiivne. Selle asemel täidab Cleopatra järjestikused kontrollitavad bitid vahelduvate nullide ja ühtedega, nagu eespool toodud näites kirjeldatud.

Sisendi vorming: **Esimene rida sisendist sisaldab täisarvu T , mis määrab stsenaariumide arvu.** Järgnevad T stsenaariumit. Igaüks neist on esitatud järgmises vormingus:

```
S
M[0]  M[1]  ...  M[S-1]
C[0]  C[1]  ...  C[30]
```

Väljundi vorming: Näidishindaja kirjutab kõigi T stsenaariumi tulemused samas järjekorras, nagu need on sisendis antud, järgmises vormingus:

```
K L
D[0]  D[1]  ...  D[L-1]
```

Siin on K send_packet väljakutsete arv, D on receive_message poolt tagastatud sõnum ja L on selle pikkus.