

## Mesaj

Aişə və Bəsmə bir-biri ilə yaxşı yola gedən iki dostdur. Aişənin Bəsməyə göndərmək istədiyi  $S$  sayda bit (yəni, sıfır və ya bir) ardıcılığından ibarət  $M$  mesajı var. Aişə Bəsmə ilə **paketlər** göndərərək ünsiyyət qurur. Paket 0-dan 30-a qədər indekslənmiş 31 bit ardıcılığıdır. Aişə Bəsməyə bir neçə paket göndərməklə  $M$  mesajını göndərmək istəyir.

Təəssüf ki, Kleopatra Aişə ilə Bəsmə arasındakı əlaqəyə müdaxilə etdi və paketləri **zədələmək** qabiliyyətini əldə etdi. Yəni, hər bir paketdə Kleopatra tam olaraq 15 indeksdə olan bitləri dəyişdirə bilər. Konkret olaraq, 31 uzunluğunda  $C$  massivi var, burada hər bir element 0 və ya 1-dir və aşağıdakı mənaya gəlir:

- $C[i] = 1$   $i$  indeksli bitin Kleopatra tərəfindən dəyişdirilə biləcəyini göstərir. Biz bu indeksləri Kleopatranın **nəzarətində** olan indekslər adlandırırıq.
- $C[i] = 0$   $i$  indeksli bitin Kleopatra tərəfindən dəyişdirilə bilməyəcəyini göstərir.

$C$  massivi dəqiq olaraq 15 bir və 16 sıfırdan ibarətdir.  $M$  mesajı göndərilərkən, Kleopatranın nəzarətində olan indekslər dəsti bütün paketlər üçün eynidir. Aişə hansı 15 indeksin Kleopatranın nəzarətində olduğunu dəqiq bilir. Bəsmə yalnız 15 indeksin Kleopatranın nəzarətində olduğunu bilir, lakin onların hansı indekslər olduğunu bilmir.

$A$  Aişənin göndərməyə qərar verdiyi paket olsun (biz bunu **orijinal paket** adlandırırıq).  $B$  Bəsmə tərəfindən qəbul edilən paket olsun (biz bunu **zədələnmiş paket** adlandırırıq).  $0 \leq i < 31$  olan hər bir  $i$  üçün:

- əgər Kleopatra  $i$  indeksli bitə nəzarət etmirsə ( $C[i] = 0$ ), Bəsmə Aişənin göndərdiyi  $i$  indeksli bit olduğu kimi alır ( $B[i] = A[i]$ ),
- əks halda, əgər Kleopatra  $i$  indeksli bitə nəzarət edirsə ( $C[i] = 1$ ),  $B[i]$  dəyəri Kleopatra tərəfindən müəyyən edilir.

Hər bir paketi göndərdikdən dərhal sonra Aişə müvafiq zədələnmiş paketin nə olduğunu öyrənir.

Aişə bütün paketləri göndərdikdən sonra Bəsmə bütün zədələnmiş paketləri **göndərildikləri ardıcılıqla** alır və orijinal mesajı, yəni  $M$ -i bərpa etməlidir.

Sizin tapşırığınız Aişəyə  $M$  mesajını Bəsməyə göndərməyə imkan verən, Bəsmənin zədələnmiş paketlərdən  $M$ -i bərpa edə biləcəyi strategiyani hazırlamaq və həyata keçirməkdir. Konkret olaraq, iki proseduru icra etməlisiniz. Birinci prosedur Aişənin hərəkətlərini yerinə yetirir. Ona  $M$  mesajı və  $C$  massivi verilir və o, mesajı Bəsməyə ötürmək üçün bəzi paketlər göndərməlidir. İkinci prosedur

Bəsmənin hərəkətlərini yerinə yetirir. Ona zədələnmiş paketlər verilir və o, orijinal mesajı, yəni  $M$ -i bərpa etməlidir.

## İcra Təfərrüatları

İcra etməli olduğunuz birinci prosedur:

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

- $M$ : Aişənin Bəsməyə göndərmək istədiyi mesajı təsvir edən  $S$  uzunluğunda massiv.
- $C$ : Kleopatranın nəzarətində olan bitlərin indekslərini bildirən 31 uzunluğunda massiv.
- Bu prosedur hər bir testdə **ən çox 2100 dəfə** çağırıla bilər.

Bu prosedur paket göndərmək üçün aşağıdakı proseduru çağırmalıdır:

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

- $A$ : Aişə tərəfindən göndərilən bitləri təmsil edən orijinal paket (uzunluğu 31 olan massiv).
- Bu prosedur Bəsmə tərəfindən qəbul ediləcək bitləri təmsil edən zədələnmiş  $B$  paketini qaytarır.
- Bu prosedur `send_message`-in hər çağırışında **ən çox 100 dəfə** çağırıla bilər.

İcra etməli olduğunuz ikinci prosedur:

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

- $R$ : zədələnmiş paketləri təsvir edən massiv. Paketlər Aişə tərəfindən bir `send_message` çağırışında göndərilən paketlərdən yaranır və Aişə tərəfindən **göndərildiyi sıraya görə** verilir.  $R$ -nin hər bir elementi zədələnmiş bir paketi təmsil edən 31 uzunluğunda massivdir.
- Bu prosedur  $M$  orijinal mesajına bərabər olan  $S$  sayda bitdən ibarət massiv qaytarmalıdır.
- Bu prosedur hər bir testdə **bir neçə dəfə**, hər bir müvafiq `send_message` çağırışı üçün **tam olaraq bir dəfə** çağırıla bilər. `receive_message` **proseduruna olan çağırışların sırası**, müvafiq `send_message` çağırışlarının sırası ilə mütləq eyni deyil.

Qeyd edək ki, qiymətləndirmə sistemində `send_message` və `receive_message` prosedurları **iki ayrı proqramda** çağırılır.

## Məhdudiyyətlər

- $1 \leq S \leq 1024$
- $C$  tam olaraq 31 elementdən ibarətdir, bunlardan 16-sı 0-a və 15-i 1-ə bərabərdir.

## Alt Tapşırıqlar və Qiymətləndirmə

Testlərin hər hansı birində send\_packet proseduruna edilən çağırışlar yuxarıda qeyd olunan qaydalara uyğun gəlmirsə və ya receive\_message proseduruna edilən çağırışların hər hansı birinin qaytardığı dəyər yanlışdırsa, həllinizin balı bu test üçün 0 olacaq.

Əks halda, gəlin  $Q$  bütün testlər üzrə bütün send\_message çağırışları arasında send\_packet proseduruna edilən çağırışların maksimum sayı olsun. Həmçinin  $X$  bərabər olsun:

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

Sonra bal aşağıdakı kimi hesablanır:

Alt Tapşırıq	Bal	Əlavə Məhdudiyyətlər
1	$10 \cdot X$	$S \leq 64$
2	$90 \cdot X$	Əlavə məhdudiyyət yoxdur.

Qeyd edək ki, bəzi hallarda qiymətləndiricinin davranışı **adaptiv** ola bilər. Bu o deməkdir ki, send\_packet tərəfindən qaytarılan dəyərlər təkcə onun giriş parametrlərindən deyil, həm də bir çox başqa şeylərdən, o cümlədən bu prosedura əvvəlki çağırışların giriş və qaytarılan dəyərlərindən və qiymətləndirici tərəfindən yaradılan psevdo-təsadüfi ədədlərdən asılı ola bilər. Qiymətləndirici **deterministikdir** yəni ki, onu iki dəfə işlətsəniz və hər iki işləyişdə eyni paketləri göndərsəniz, o, onlara eyni dəyişiklikləri edəcək.

## Nümunə

Aşağıdakı çağırışı nəzərdən keçirin.

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

Aişənin Bəsməyə göndərməyə çalışdığı mesaj  $[0, 1, 1, 0]$ -dir. 0-dan 15-ə qədər indeksləri olan bitləri Kleopatra dəyişdirə bilməz, 16-dan 30-a qədər olan bitləri isə Kleopatra dəyişdirə bilər.

Bu nümunə üçün, fərz edək ki, Kleopatra idarə etdiyi ardıcıl bitləri növbəli olaraq 0 və 1 ilə doldurur, yəni nəzarət etdiyi birinci indeksə 0 (bizim nümunədə 16), ikinci indeksə 1 (17-ci indeks), üçüncüyə 0 (18-ci indeks) və sairə təyin edir.

Aişə orijinal mesajdan iki biti bir paketdə aşağıdakı şəkildə göndərmək qərarına gələ bilər: o, birinci biti idarə etdiyi ilk 8 indeksə, ikinci biti isə idarə etdiyi növbəti 8 indeksə göndərəcək.

Bundan sonra Aişə aşağıdakı paketi göndərməyi seçir:

```
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])
```

Qeyd edək ki, Kleopatra son 15 indeksdəki bitləri dəyişə bilər, ona görə də Aişə onları istədiyi kimi təyin edə bilər, çünki onların dəyərləri dəyişdirilə bilər. Kleopatranın fərz edilən strategiyası ilə prosedur bu massivi geri qaytarır:  $[0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0]$ .

Aişə  $M$ -in son iki bitini ikinci paketdə əvvəlki kimi göndərmək qərarına gəlir:

```
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, 0, 0])
```

Kleopatranın fərz edilən strategiyası ilə prosedur bu massivi geri qaytarır:  $[1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0]$ .

Aişə daha çox paket göndərə bilər, lakin o, göndərməmək qərarına gəlir.

Sonra qiymətləndirici aşağıdakı proseduru çağırır:

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

Bəsmə  $M$  mesajını aşağıdakı kimi bərpa edir. Hər paketdən o, ardıcıl iki dəfə var olan ilk bitini və ardıcıl olaraq iki dəfə var olan sonuncu bitini götürür. Yəni birinci paketdən o,  $[0, 1]$  bitlərini, ikinci paketdən isə  $[1, 0]$  bitlərini götürür. Onları bir araya gətirməklə o, bu `receive_message` çağırışı üçün düzgün geri qaytarma dəyəri olan  $[0, 1, 1, 0]$  mesajını bərpa edir.

Göstərilə bilər ki, Kleopatranın fərz edilən strategiyası və uzunluğu 4 olan mesajlar üçün Bəsmənin bu yanaşması  $C$  dəyərindən asılı olmayaraq  $M$  mesajını düzgün şəkildə bərpa edir. Ancaq ümumi halda bu düzgün deyil.

## Nümunə Qiymətləndirici

Nümunə qiymətləndirici adaptiv deyil. Bunun əvəzinə, Kleopatra yuxarıdakı nümunədə təsvir olunduğu kimi, idarə etdiyi ardıcıl bitləri növbəli olaraq 0 və 1 bitləri ilə doldurur.

Giriş formatı: **Girişin birinci sətirində ssenarilərin sayını göstərən  $T$  tam ədədi var.  $T$  ssenariləri növbəti sətirlərdə verilir. Onların hər biri aşağıdakı formatda verilir:**

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

Çıxış formatı: Nümunə qiymətləndirici  $T$  ssenarinin hər birinin nəticəsini aşağıdakı formatda girişdə təqdim edildiyi ardıcılıqla yazır:

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

Burada,  $K$  send\_packet çağırışlarının sayıdır,  $D$  receive\_message tərəfindən geri qaytarılan mesajdır və  $L$  onun uzunluğudur.