

Message

Aisha dhe Basma janë dy shoqe që përshtaten me njëra tjetër. Aisha ka një mesazh M , e cila ka një sekuençë S bits (prsh. zero ose një) dhe do të dëshironte t'ia dërgonte Basmasë. Aisha komunikon me Basmanë duke i dërguar **paketa**. Një paketë është një sekuençë me 31 bite të indeksuar nga 0 deri 30. Aisha dëshiron të dërgoj mesazhin M tek Basma duke i dërguar asaj disa numra nga paketa.

Fatkeqësisht, Kleopatra rrezikon komunikimin midis Aishes dhe Basmasë dhe është në gjendje që të **modifikojë** paketat. Kjo do të thotë që Kleopatra mund të modifikojë bite në 15 indekse. Konkretisht, është një matricë C me gjatësi 31, tek i cili secili element është 0 ose 1, me kuptimin si më poshtë:

- $C[i] = 1$ kjo tregon se biti me indeks i mund të ndryshohet nga Kleopatra. Këto indekse mund ti quajm të **kontrolluara** nga Kleopatra.
- $C[i] = 0$ kjo tregon se biti me indeks i nuk mund të ndryshohet nga Kleopatra.

Matrica C përmban saktësisht 15 njësha dhe 16 zero. Gjatë dërgimit të një mesazhi, grupin e indekseve të kontrolluar nga Kleopatra qëndron njësoj për të gjitha paketat. Aisha e di saktësisht se cilat janë 15 indekset që kontrollon Kleopatra. Basma di vetëm që 15 indekse kontrollohen nga Kleopatra, por nuk di se cilat indekse kontrollohen.

A është një pako që Aisha vendos ta dërgojë (që ne e quajm **paketa origjinale**). B është paketa që marrë nga Basma (që ne e quajmë **paketa të ndryshuara**). Për secilën i , të tilla që $0 \leq i < 31$:

- Nëse Kleopatra nuk kontrollon bitin me indeksin i ($C[i] = 0$), Basma merr bitet siç i ka dërguar Aisha ($B[i] = A[i]$),
- përndryshe, nëse Kleopatra kontrollon bitin me indeks i ($C[i] = 1$), vlera e $B[i]$ vendoset nga Kleopatra.

Menjëherë pas dërgimit të secilës paketë, Aisha mëson se cila është paketa e nacmuar.

Pasi Basma i merr të gjitha paketat e ngacmuara sipas renditjes që ishin dërguar duhet të rindërtojë mesazhin origjinal M .

Detyra është të krijoni dhe të implementoni një strategji që i lejon Aishës të dërgoj mesazhin M tek Basma, kështu Basma mund të rimarrë mesazhin M nga paketa e ngacmuar. Konkretisht, ju duhet të implementoni dy procedura. Procedura e parë tregon veprimet e Aishës. Jep një mesazh M dhe

matrica C , dhe dërgon disa paketa që të dërgoj mesazhin tek Basma. Procedura e dytë tregon veprimet që bën Basma. Jepet paketa e ngacmuar dhe do të rimarrim paketën origjinale M .

Implementation Details

Procedura e parë që duhet të zbatoni është:

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

- M : një matricë me gjatësi S jep mesazhin që Aisha do të dërgoj tek Basma.
- C : një matricë me gjatësi 31 tregon bitet e kontrolluara nga CKleopatra.
- Kjo procedurë mund të thërret **2100 times** për çdo rast testimi.

Kjo procedurë do thërrasi procedurën më poshtë për të dërguar një paketë:

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

- A : një paketë origjinale (një matricë me gjatësi 31) përfaqëson bitet të dërguara nga Aisha.
- Kjo procedurë tregon paketat e ngacmuara B duke përfaqësuar bitet që do të marri Basma.
- Kjo procedurë mund të thërret 100 herë në cdo thirrje send_message.

Procedura e dytë që duhet të zbatoni është:

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

- R : matrica përshkon paketën e ndryshuar. Paketat dalin nga paketat që Aisha ka dërguar nga një telefonst send_message **ne menyrën që i ka dërguar** . Çdo element R është një matricë me gjatësi 31, përfaqëson një paket të ndryshuar .
- Kjo procedurë do të afishoj matricën me S bits e cila është e barabart me mesazhin orgjinal M . **Order of receive_message procedure calls** nuk është domosdoshmërisht e njëjtë me send_message .

Vini re se në sistemin e gradimit send_message dhe receive_message procedura thërren **separate programs**.

Constraints

- $1 \leq S \leq 1024$
- C ka fiks 31 element, ku 16 janë të barabarta me 0 dhe 15 janë të barabarta me 1.

Subtasks and Scoring

Nëse në ndonjë nga rastet e testimit, thirrjet në procedurë `send_packet` nuk janë në përputhje me rregullat e përmendura më lart, ose vlerën e afishimit të ndonjë prej thirrjeve në procedurë `receive_message` është gabim rezultati i zgjidhjes suaj për atë rast testimi do të jetë 0.

Përndryshe, Q të jetë numri maksimal i thirrjeve në procedurë `send_packet` ndër të gjitha thirrjet të `send_message` gjatë ratseve të testimit. Gjithashtu X be equal to:

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

Më pas, rezultati llogaritet si më poshtë:

Subtask	Score	Additional Constraints
1	$10 \cdot X$	$S \leq 64$
2	$90 \cdot X$	No additional constraints.

Vini re se në disa raste kemi **adaptive**. Kjo do të thotë që vlera afishohet nga `send_packet` mund të varet nga argumentet e tij hyrëse dhe vlera e kthimit të thirrjeve të mëparshme në këtë procedurë.

Example

Consider the following call.

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

Mesazhi që Aisha mundohet të dërgoj tek Basma është $[0, 1, 1, 0]$. Bitet me indeks nga 0 deri 15 nuk mund të ndryshojn nga Kleopatra, por bitet me indeks nga 16 deri 30 mund të ndryshojn nga Kleopatra.

Për këtë shembull , le të supozojmë se Kleopatra plotëson pjesët e njëpasnjëshme që ajo kontrollon me alternim 0 dhe 1, i.e. ajo cakton 0 indeksi i parë që ajo kontrollon (index 16 është në rastin tonë), 1 tindeksi i dytë që ajo kontrollon (index 17), 0 indeksi i tretë që ajo kontrollon (index 18), the vazhdon.

Aisha mund të vendosë të dërgojë dy bit nga mesazhi origjinal në një paketë si më poshtë: ajo do të dërgojë bitin e parë në fillim 8 në indekset që ajo kontrollon

dhe biti i dytë në vijim 8 në indekset që ajo kontrollon.

Aisha then chooses to send the following packet:

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

Shëno që Kleopatra mund të ndryshoj bitin e fundit 15, kështu që Aisha mund t'i vendosë ato në mënyrë arbitrare, pasi ato mund të mbishkruhen.

Me strategjinë e supozuar të Kleopatrës, procedura është: $[0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0]$.

Aisha dëshironte dërgoj dy bitet e fundit të M në paketën e dytë në një mënyrë të ngjashme si më parë:

```
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])
```

Me strategjinë e supozuar të Kleopatrës, procedura është: $[1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0]$.

Aisha mund të dërgojë më shumë pako, por ajo zgjodhi mos të dërgoj.

The grader then makes the following procedure call:

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

Basma rimerr mesazhin M si më poshtë. Nga çdo paketë ajo merr bitin e parë që ndodh dy herë radhazi, dhe biti i fundit që ndodh dy herë radhazi.

Kjo është nga paketa e parë ajo merr $[0, 1]$, dhe nga dyta merr bitet $[1, 0]$. Duke i bashkuar, ajo rimerr mesazhin $[0, 1, 1, 0]$, e cila është vlera e saktë për këtë thirrje `receive_message`.

Mund të tregohet se me strategjinë e supozuar të Kleopatrës dhe mesazhet me gjatësi gjata 4, kjo qasje e Basmasë rimëkëmb saktë M , pavarësisht nga vlera e C . Megjithatë, nuk është e saktë në rastin e përgjithshëm.

Sample Grader

The sample grader is not adaptive. Sjellja e Kleopatrës është vendimtaree, dhe ajo i plotëson pjesët e njëpasnjëshme që ajo kontrollon me alternim 0 dhe 1 bits siç përshkruhet në shembullin e mësipërm.

Input format: **Rreshti i parë i hyrjes përmban një numër të plotë T , duke specifikuar numrin e skenarëve** . T scenarios follow. Secila prej tyre ofrohet në formatin e mëposhtëm :

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

Output format: The sample grader writes the result of each of the T scenarios in the same order as they are provided in the input in the following format:

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

Ketu, K është numri i thirrjeve në send_packet, D mesazhi qe afishohet receive_message dhe L gjatësia.