Mi lehet a legjobb ajándék? Két tömb!

Miután Péter visszatért a felfedezőútjáról, kiderült, hogy ma van a születésnapja. A születésnapi ajándéka nem egy tömb volt, hanem kettő! Mindkét tömb nemnegatív egész számokból állt, és azonos N hosszúságúak voltak. Jelöljük az első tömböt A-val, a második tömböt pedig B-vel! Mindkét tömb sorra 1-től N-ig van indexelve. Mivel Péter nagy rajongója a bitműveleteknek, kitalálta, hogyan mérje össze ezen tömbök értékét bitműveletek segítségével. Az L-től R-ig terjedő indexű elemek által alkotott résztömb értékét így definiálja: $(A_L\ xor\ A_{L+1}\ xor\ ...\ xor\ A_R) \cdot (B_L\ and\ B_{L+1}\ and\ ...\ and\ B_R)$, ahol a xor a bitenkénti kizáró vagy műveletet, az and a bitenkénti logikai konjunkció műveletet, a $\cdot$ pedig a szorzást jelenti. A feladatotok az, hogy megtaláljátok a legnagyobb értékű résztömböt – pontosabban, az összes résztömb közül a legnagyobb értékűt.

A bemenet leírása

A szabványos bemenet első sorában egy N egész szám áll, amely a két tömb hosszát jelöli. A szabványos bemenet második sorában N egész szám áll, ahol az i-dik szám az A_i-t jelöli. A szabványos bemenet harmadik és egyben utolsó sorában N egész szám áll, ahol az i-dik szám a B_i-t jelöli.

A kimenet leírása

A szabványos kimenet első és egyben egyetlen sorában a keresett legnagyobb értéket kell kiíratni.

1. Példa

Bemenet

```
3
4 5 2
0 1 3
```

Kimenet

```
7
```

A példa magyarázata

Az $L = 2$, $R = 3$ résztömbre érhető el a legmagasabb érték. $A_2\ xor\ A_3 = 7$ és $B_2\ and\ B_3 = 1$, így ennek a résztömbnek az értéke: $7 \cdot 1 = 7$.

Korlátozások

- $1 \leq N \leq 2 \cdot 10^5$
- $0 \leq A_i, B_i \leq 10^9$

A tesztpéldák hat diszjunkt csoportba vannak sorolva:

- A 4 pontot érő tesztpéldákban: $N \leq 2000$.
- A 4 pontot érő tesztpéldákban: $A_i = 1$.

- A 12 pontot érő tesztpéldákban: $A_i \leq 500$.
- A 20 pontot érő tesztpéldákban: $B_i = 1$.
- A 24 pontot érő tesztpéldákban: $B_i \leq 3$.
- A 36 pontot érő tesztpéldákban: nincsenek további korlátozások.

Megjegyzés

A kizáró vagy jelölésére Pascalban a `xor`-t használják, míg C++-ban a `^` szimbólummal írjuk le. Az x xor y műveletet az x, y nemnegatív számok esetén a folytatásban leírt módon definiálhatjuk. Először a számokat bináris formában kell felírni. Ha egy szám rövidebb, mint a másik, akkor azt vezető nullákkal kell kiegészíteni mindaddig, amíg a bináris számjegyek száma nem lesz megegyező. Így a bináris számok két sorozatát kapjuk, amelyekre $a_1, \ldots, a_k$ és $b_1, \ldots b_k$ jelölést használjuk. Majd minden $i \in 1, \ldots, k$ pozícióra kiszámoljuk a c_i-t az alábbi szabályok mentén:

- Ha $a_i = 0, b_i = 0$, akkor $c_i = 0$
- Ha $a_i = 0, b_i = 1$, akkor $c_i = 1$
- Ha $a_i = 1, b_i = 0$, akkor $c_i = 1$
- Ha $a_i = 1, b_i = 1$, akkor $c_i = 0$

A $c_1, \ldots, c_k$ bináris számjegyekből alkotott sorozat (amely tartalmazhat vezető nullákat is) az eredmény bináris leírása, vagyis az x xor y számé.

A n elem $(x_1, x_2, ..., x_n)$ közötti bitenkénti konjunkció az alábbi módon van meghatározva: x_1 and x_2 and ... and $x_n = (...(((x_1$ and $x_2)$ and $x_3)$ and $x_4)...)$ and x_n.