

BitBeat (A)

Limit pamięci: 256 MB

Limit czasu: 1.00 s

BitBeat to rytmiczna gra komputerowa, w której gracz układa sekwencje dźwięków. Do dyspozycji ma N dźwięków ponumerowanych liczbami od 1 do N . Musi ułożyć z nich utwór rytmiczny, a zarazem nowatorski i różnorodny pod względem użytych tonów.

Każdy dźwięk ma przypisane dwie liczby: dla i -tego dźwięku są to A_i oraz B_i . Oznaczają one odpowiednio jego wysokość i punktację. Powiemy, że ciąg dźwięków d_1, \dots, d_k jest rytmiczny, jeśli $A_{d_i} \text{ XOR }^1 A_{d_{i+1}} > \max(A_{d_i}, A_{d_{i+1}})$ dla każdego $i < k$. Gracz dostaje punkty za **różne**² dźwięki występujące w ciągu. Konkretnie, jeśli w ciągu pojawia się dźwięk x , to gracz dodaje do swojego wyniku liczbę B_x . Za kolejne wystąpienia tego samego dźwięku nie otrzymuje dodatkowych punktów.

Jaś, który dostał tę wspaniałą grę za darmo na Epicu, zastanawia się teraz, od jakiego dźwięku powinien rozpocząć swój ciąg rytmiczny. Pomóż mu i napisz program, który dla każdego i od 1 do N wypisze maksymalną liczbę punktów, jaką może uzyskać, rozpoczynając swój ciąg od dźwięku i .

Wejście

W pierwszym wierszu wejścia znajduje się jedna liczba całkowita N oznaczająca liczbę dźwięków.

W drugim wierszu wejścia znajduje się N liczb całkowitych A_i , oddzielonych pojedynczymi odstępami, będących wysokościami kolejnych dźwięków.

W trzecim wierszu wejścia znajduje się N liczb całkowitych B_i , oddzielonych pojedynczymi odstępami, będących punktacją za kolejne dźwięki.

Wyjście

Wyjście powinno składać się z N linii.

W i -tej z nich powinna znaleźć się maksymalna liczba punktów możliwa do uzyskania z ciągu rytmicznego rozpoczynającego się od dźwięku i .

Ograniczenia

$$1 \leq N \leq 100\,000.$$

$$1 \leq A_i \leq 2^{30} - 1.$$

$$1 \leq B_i \leq 10^9.$$

Podzadania

Podzadanie	Warunki	Punkty
1	$N \leq 200$	13
2	$N \leq 2000$	18
3	$A_1 = A_2 = \dots = A_n$	4
4	Liczby A_i są potęgami dwójki.	7
5	$A_i \leq 2^{12} - 1$	19
6	Bez dodatkowych ograniczeń	39

Przykład

Wejście

Wyjście

Wyjaśnienie

¹Operacja XOR jest opisana tutaj: XOR Wikipedia. W C++ i Pythonie symbol operacji XOR to \wedge .

²Dźwięki są różne, jeśli są oznaczone różnymi liczbami. Różne dźwięki mogą mieć tę samą wysokość oraz punktację.

3	60
2 1 1	60
20 30 10	60

W tym przykładzie rytmiczne ciągi to dokładnie takie, w których dźwięk 1 może sąsiadować z dźwiękiem 2 (bo $2 \text{ XOR } 1 = 3 > \max(2, 1) = 2$), oraz dźwięk 1 może sąsiadować z dźwiękiem 3 (z tego samego powodu). Dźwięk 2 nie może sąsiadować z dźwiękiem 3, bo $1 \text{ XOR } 1 = 0 < 1$. Ciągi maksymalizujące wynik to np: 1213, 213, 312. Ciągi te zawierają wszystkie dźwięki, więc dają maksymalny wynik.

Wejście

4
5 4 4 5
10 20 30 40

Wyjście

10
20
30
40

Wyjaśnienie

W tym przykładzie nie istnieją rytmiczne ciągi o długości większej niż jeden.

Wejście

5
1 2 1 7 11
20 10 30 100 100

Wyjście

60
60
60
200
200