

Diamentowy szyfr

Dostępna pamięć: 32 MB.

Renomowana firma jubilerska zamówiła u Ciebie oprogramowanie zabezpieczające sejfy. Firma produkuje dwa rodzaje sejfów do przechowywania diamentów — jeden z 20 przełącznikami, a drugi z 200 przełącznikami. Aby otworzyć sejf potrzebujesz hasła, które składa się z ciągu cyfr. Opracuj program, który na podstawie hasła wygeneruje sposób ustawienia przełączników. Przełączniki ponumerowano od 0 do $N - 1$, i -tej pozycji przyporządkowano wartość 3^i . Pojedynczy przełącznik ma 3 stany (Góra, Zero i Dół). Sejf otwiera się jeżeli suma wartości przyporządkowanych przełącznikom ustawionym do góry odjąć suma wartości przyporządkowanych przełącznikom ustawionym na dół równa się wartości hasła.

Zadanie

Opracuj program, który:

- wczyta ze standardowego wejścia hasła do sejfów,
- dla każdego hasła wygeneruje ustawienie przełączników,
- wypisze wynik na standardowe wyjście.

Wejście

W pierwszym wierszu zapisano wartość N oznaczającą liczbę haseł ($1 \leq N \leq 250$). W każdym z następnych N wierszy znajduje się jedna liczba bez zer znaczących oznaczająca hasło do sejfów. 50% testów zawiera wyłącznie hasła odpowiadające sejfom z 20 przełącznikami.

Wyjście

Dla każdego hasła wypisz stan przełączników otwierających sejf w dwóch wierszach. W pierwszym wierszu wypisz liczbę przełączników ustawionych do góry oraz ich numery, w drugim wierszu liczbę przełączników ustawionych w dół oraz ich numery. Wszystkie liczby w obrębie jednego wiersza należy oddzielać pojedynczym znakiem spacji.

Przykład

Dla danych wejściowych:

```
2
5
10
```

poprawnym wynikiem jest:

```
1 2
2 0 1
2 0 2
0
```

Rozwiązanie

Problem układania diamentowego szyfru z przełączników dodających lub odejmujących wartości różnych potęg liczby 3 jest ciekawym sposobem przedstawienia trójkowego systemu liczbowego. Zadanie polega na reprezentowaniu dowolnej liczby x podanej jako hasło do sejfów za pomocą dokładnie dwóch liczb g, d takich, że przy reprezentacji w systemie trójkowym obie składają się z samych zer i jedynek oraz spełniających $x = g - d$. Stany przełączników sejfów odpowiadają wtedy miejscom występowania jedynek w obu liczbach, tj. do góry przestawione są te, które odpowiadają jedynkom w g , do dołu przestawione są odpowiadające jedynkom w d .

Przekształćmy równanie z poprzedniego akapitu:

$$g = x + d \quad (1)$$

oraz oznaczmy cyfry na i -tej pozycji w reprezentacji trójkowej tych liczb przez g_i, d_i, x_i . Ponieważ żaden przełącznik nie może być równocześnie odwrócony w górę i w dół, otrzymujemy warunek

$$g_i + d_i \neq 2 \quad (2)$$

Z powyższych obserwacji dla każdego i wynika

$$g_i = x_i + d_i \quad (3)$$

Możemy teraz skonstruować algorytm obliczający g i d mając jako daną liczbę x w systemie trójkowym. Badamy każdą cyfrę x w kolejności od najmniej znaczących. Mamy trzy możliwe przypadki:

1. $x_i = 0$

Z równania (3) wynika $g_i = d_i$, a warunek (2) wymusza, aby obie te liczby były równe zero.

2. $x_i = 1$

Z równania (3) wynika $g_i = 1 + d_i$, a warunek (2) wymusza $d_i = 0$ i $g_i = 1$.

3. $x_i = 2$

Z równania (3) wynika $g_i = 2 + d_i$. Pozornie otrzymujemy sprzeczność z warunkami zadania, musiałyby bowiem zachodzić $g_i = 2$ i $d_i = 0$, ale ponieważ poruszamy się w obrębie arytmetyki o podstawie 3, skorzystajmy z następującego faktu:

$$10_3 = 2_3 + 1_3 \quad (4)$$

Otrzymujemy z tego wartości przełączników spełniające warunek (2): $g_i = 0, d_i = 1$. Oczywiście nie możemy zapomnieć o dodatkowej jedynce w następnej kolumnie: $x_{i+1} = x_{i+1} + 1$.

Powyższy algorytm potrzebuje wejściowej liczby x przedstawionej w systemie trójkowym, należy więc sprowadzić podane hasło do takiej postaci. W przypadku sejfów o 20 przełącznikach wystarczą do tego liczby 32-bitowe. Wartości liczbowe przełączników o wyższych numerach wymuszają użycie własnej arytmetyki dużych liczb.