

# Prawie Hanoi

XX OIJ, zawody III stopnia  
18 kwietnia 2026

Kod zadania: **phn**  
Limit czasu: **10s**  
Limit pamięci: **256MB**  
Maksymalna liczba punktów: **100**



**Uwaga:** To jest zadanie z otwartym sprawdzaniem. Swój wynik punktowy możesz zobaczyć niedługo po wystaniu swojego programu.

Bajtek dostał ostatnio na urodziny wieżę Hanoi. Ponieważ jest jeszcze małym chłopcem, a klasyczne zasady gry wydały mu się zbyt trudne, postanowił wymyśleć własne.

Zestaw składa się z trzech pól numerowanych od 1 do 3 oraz z  $N$  klocków o wielkościach  $1, 2, \dots, N$ . Początkowo wszystkie klocki leżą jeden na drugim na pierwszym polu w pewnej kolejności (niekoniecznie ułożone zgodnie z regułami klasycznej wieży Hanoi!). W jednym ruchu Bajtek może wziąć z góry dowolnego pola pewną liczbę klocków – pod warunkiem, że są one posortowane rosnąco (od góry do dołu) – i przełożyć je na inne pole. W szczególności zawsze może wziąć jeden klocek z góry dowolnego pola. Jego celem jest przeniesienie wszystkich klocków na drugie pole w taki sposób, aby były one posortowane rosnąco (od dołu do góry).

Bajtek zafascynował się nową zabawką, jednak szybko zauważył, że nie zawsze potrafi znaleźć odpowiednią sekwencję ruchów. Zwrócił się więc do Ciebie o pomoc.

## Wejście

W pierwszym wierszu wejścia znajdują się dwie liczby całkowite  $N$  i  $K$  ( $1 \leq N \leq 100\,000$ ,  $10 \cdot N \leq K \leq 1\,000\,000$ ), oznaczające odpowiednio liczbę klocków oraz maksymalną liczbę ruchów, które można wykonać. W drugim (i ostatnim) wierszu znajduje się ciąg  $N$  różnych liczb całkowitych  $A_1, A_2, \dots, A_N$  ( $1 \leq A_i \leq N$ ), określający początkową kolejność klocków na pierwszym polu (od dołu do góry).

## Wyjście

W pierwszym wierszu wyjścia należy wypisać jedną liczbę całkowitą  $M$  ( $0 \leq M \leq K$ ), oznaczającą liczbę ruchów wykonanych przez Bajtkę. Rozwiązanie wymagające więcej niż  $K$  ruchów zostanie uznane za niepoprawne.

W kolejnych  $M$  wierszach należy wypisać po trzy liczby całkowite dodatnie  $A$ ,  $B$  i  $X$  ( $1 \leq A, B \leq 3$ ,  $A \neq B$ ,  $1 \leq X \leq N$ ), oznaczające, że Bajtek przenosi  $X$  klocków z góry pola  $A$  na pole  $B$ . Aby taki ruch był dozwolony, przed jego wykonaniem na polu  $A$  musi znajdować się co najmniej  $X$  klocków, a  $X$  najwyższych z nich musi być posortowanych rosnąco (od góry do dołu). Po wykonaniu wszystkich  $M$  ruchów klocki na drugim polu powinny być ułożone rosnąco od dołu do góry.

Można udowodnić, że w warunkach zadania zawsze istnieje rozwiązanie składające się z co najwyżej  $K$  ruchów.

## Ocenianie

Zestaw testów dzieli się na następujące podzadania. Testy do każdego podzadania składają się z jednej lub większej liczby osobnych grup testów. Rozwiązanie podzadania zdobywa przypisaną mu liczbę punktów.

Dodatkowe ograniczenia	Liczba punktów
Klocki są początkowo posortowane malejąco (od dołu do góry)	16
Klocki są początkowo posortowane rosnąco (od dołu do góry)	15
$N \leq 3$	12
$N \leq 300$ , $K = 1\,000\,000$	18
$N \leq 1000$	23
Brak dodatkowych ograniczeń	16

## Przykłady

Wejście dla testu phn0a:

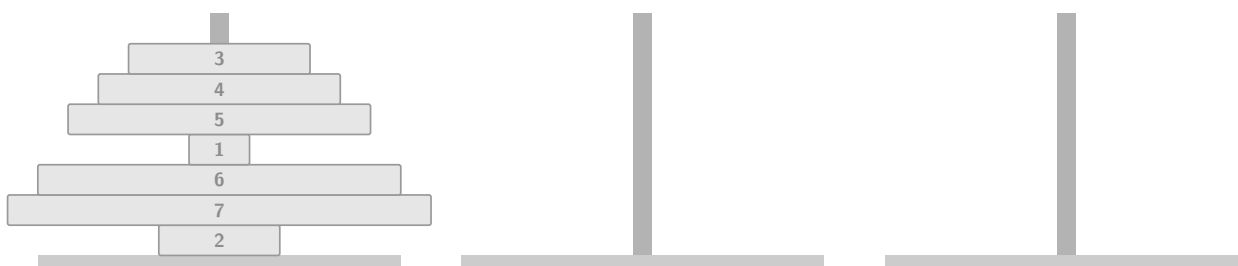
```
7 70
2 7 6 1 5 4 3
```

Wyjście dla testu phn0a:

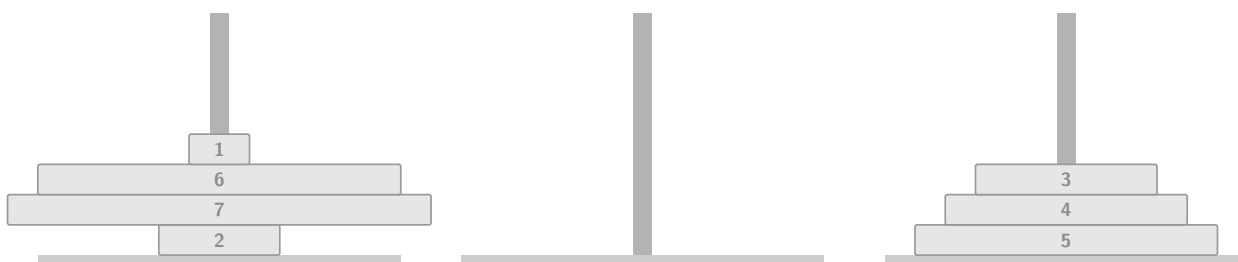
```
24
1 3 1
3 2 1
1 3 1
2 3 1
3 2 2
1 3 1
2 3 2
1 3 1
3 2 4
1 3 1
2 3 4
3 2 5
1 3 1
2 3 5
3 2 1
1 3 1
2 3 1
3 2 1
3 2 1
3 2 1
3 2 1
3 2 1
3 2 1
3 2 1
```

**Wyjaśnienie do przykładu:** Klocki na polach po kolejnych ruchach wyglądają następująco:

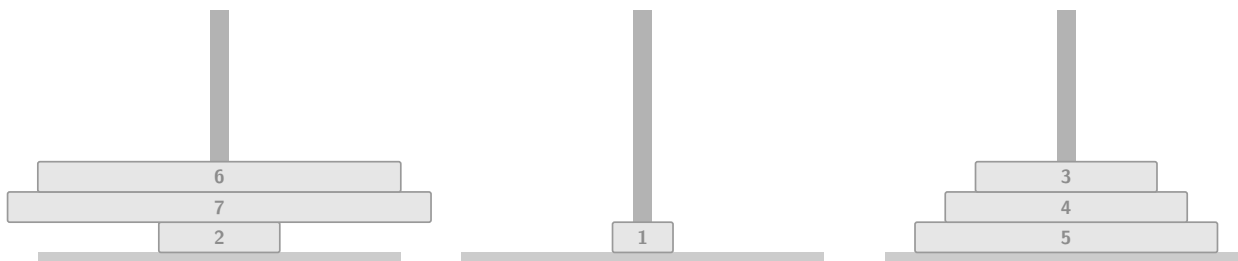
- Początkowa sytuacja:



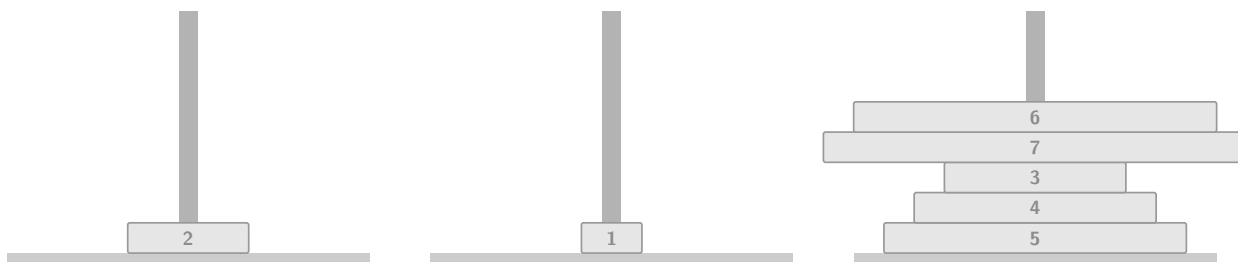
- Przenosimy 3 klocki z pola 1 na pole 3:



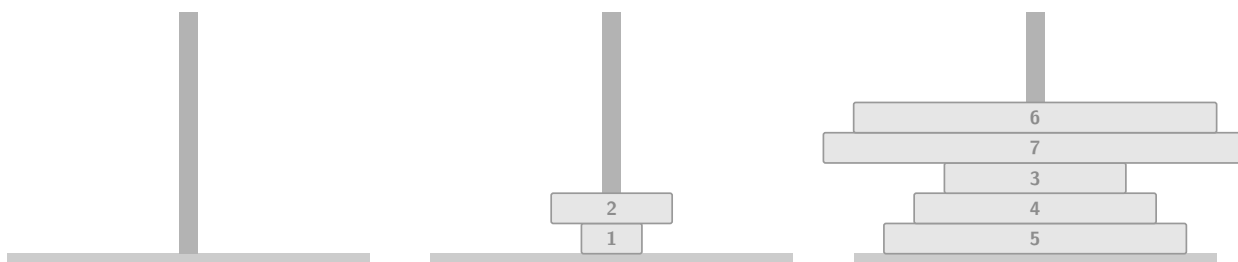
- Przenosimy 1 klocek z pola 1 na pole 2:



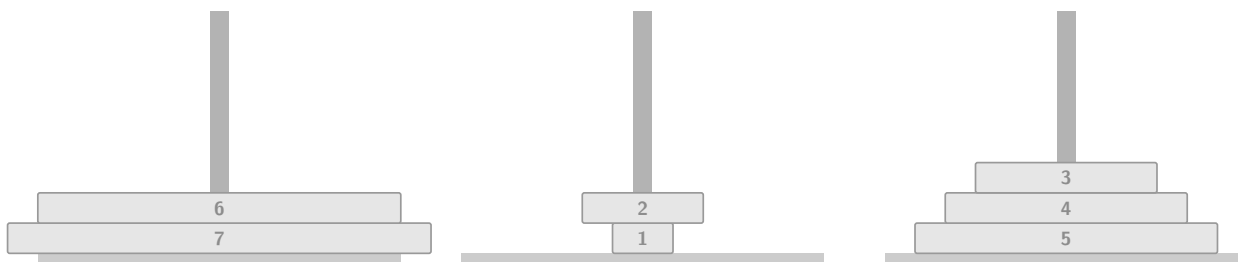
- Przenosimy 2 klocki z pola 1 na pole 3:



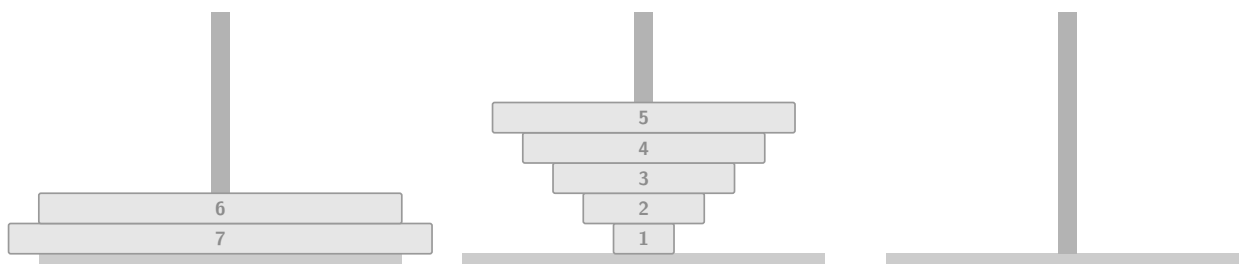
- Przenosimy 1 klocek z pola 1 na pole 2:



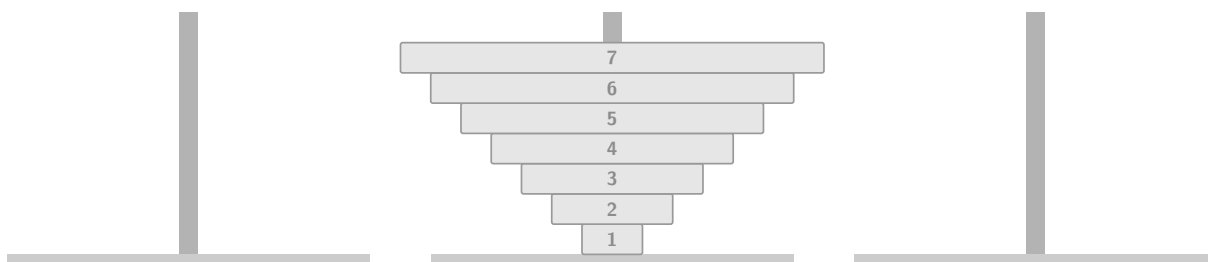
- Przenosimy 2 klocki z pola 3 na pole 1:



- Przenosimy 1 klocek z pola 3 na pole 2, trzy razy:



- Przenosimy 1 klocek z pola 1 na pole 2, dwa razy:



Klocki na polu 2 są teraz posortowane rosnąco (od dołu do góry).

Wejście dla testu phn0b:

```
3 30
3 1 2
```

Wyjście dla testu phn0b:

```
8
1 3 1
1 3 1
3 2 2
1 3 1
2 3 2
3 2 1
3 2 1
3 2 1
```

Wejście dla testu `phn0c`:

```
7 70
4 3 2 1 7 6 5
```

Wyjście dla testu `phn0c`:

```
24
1 3 1
3 2 1
1 3 1
2 3 1
3 2 2
1 3 1
2 3 2
1 3 1
3 2 1
1 3 1
2 3 1
3 2 2
1 3 1
2 3 2
3 2 3
1 3 1
2 3 3
3 2 1
3 2 1
3 2 1
3 2 1
3 2 1
3 2 1
3 2 1
3 2 1
```

