Zadanie: PER
Permutacja

Etap III. Dzień drugi. Plik źródłowy `per.*` *03.04.2008*
Dostępna pamięć: 32 MB.

Multizbiorem nazywamy obiekt matematyczny podobny do zbioru, w którym jednak ten sam element może występować wielokrotnie. Podobnie jak w przypadku zbioru, także dla multizbioru wszystkie elementy można ustawić w ciąg i to zazwyczaj na wiele sposobów; każde takie ustawienie nazywamy *permutacją* multizbioru. Dla przykładu, permutacjami multizbioru $\{1,1,2,3,3,3,7,8\}$ są między innymi $(2,3,1,3,3,7,1,8)$ oraz $(8,7,3,3,3,2,1,1)$.

Powiemy, że jedna permutacja danego multizbioru jest mniejsza (w porządku leksykograficznym) od drugiej, jeżeli na pierwszej pozycji, na której te permutacje się różnią, pierwsza z nich zawiera element mniejszy niż druga. Wszystkie permutacje **danego multizbioru** można ponumerować (zaczynając od jedynki) w kolejności od najmniejszej do największej.

Zadanie

Napisz program, który:

- wczyta ze standardowego wejścia opis permutacji pewnego multizbioru oraz dodatnią liczbę m,
- wyznaczy numer wczytanej permutacji w porządku leksykograficznym, a dokładniej jego resztę z dzielenia przez m,
- wypisze wynik na standardowe wyjście.

Wejście

Pierwszy wiersz wejścia zawiera dwie liczby całkowite n oraz m ($1 \leq n \leq 300\,000$, $2 \leq m \leq 1\,000\,000\,000$), oddzielone pojedynczym odstępem. Oznaczają one odpowiednio liczbę elementów multizbioru oraz ... liczbę m. Drugi wiersz wejścia zawiera n dodatnich liczb całkowitych a_i ($1 \leq a_i \leq 300\,000$), pooddzielanych pojedynczymi odstępami i oznaczających kolejne elementy danej permutacji multizbioru.

Wyjście

Pierwszy i jedyny wiersz wyjścia powinien zawierać jedną liczbę całkowitą, oznaczającą resztę z dzielenia przez m numeru podanej permutacji w porządku leksykograficznym.

Przykład

Dla danych wejściowych:
```
4 1000
2 1 10 2
```
poprawnym wynikiem jest:
```
5
```

Wszystkie permutacje mniejsze od zadanej to (w kolejności leksykograficznej): $(1,2,2,10)$, $(1,2,10,2)$, $(1,10,2,2)$ oraz $(2,1,2,10)$.