

Mã bài: sorting

(Sắp xếp)

(CEOI 2009)

*Output-only problem*

Trong bài toán này, bạn cần so sánh hiệu năng của hai thuật toán Insertion Sort và Quick Sort. Lưu ý, có nhiều biến thể khác nhau của các thuật toán, và trong bài toán này, chúng ta sử dụng đoạn mã giả như sau:

```
procedure insertionSort(int N, array A[1..N]) defined as:
  A[0] := -Infinity
  for i := 2 to N do:
    j := i
    Increment(comparison_count)
    while A[j - 1] > A[j] do:
      SWAP(A[j - 1], A[j])
      j := j - 1
      Increment(comparison_count)
    end while
  end for
```

```
procedure quickSort(list A) defined as:
  list less, greater
  if length(A) <= 1 then
    return A

  pivot := A[1]
  for i := 2 to length(A) do:
    Increment(comparison_count)
    if A[i] < pivot then append A[i] to less
    else append A[i] to greater
  end if
end for
return concatenate(quickSort(less), pivot, quickSort(greater))
```

Như bạn thấy, yếu tố được sử dụng so sánh là *số phép so sánh cơ bản*.

Ví dụ, với hoán vị {3, 1, 4, 2}, Insertion Sort thực hiện 6 phép so sánh và Quick Sort thực hiện 4 phép so sánh.

Bài toán được đặt ra như sau:

Cho hai số nguyên  $N$  và  $K$ , tìm số hoán vị  $P$  của  $\{1, 2, \dots, N\}$  sao cho: số phép so sánh của Insertion Sort( $P$ ) không lớn hơn  $K$  lần số phép so sánh của Quick Sort( $P$ ). Xuất kết quả lấy modulo 1234567.

Giới hạn:

- $1 < N \leq 32$
- $1 \leq K \leq N^2$

Input/Output:

Trong bài toán này, bạn được cho 10 file input x-sorting.in với x từ 0 đến 9. Mỗi input chứa hai số nguyên N và K được cách nhau bởi 1 dấu cách. Bạn cần tính kết quả ra file x—sorting.out tương ứng.

- Lời giải của BTC chạy toàn bộ 10 test này trong thời gian không quá 6 phút.

Ví dụ

Input	Output
6 2	719
21 3	660773