

Bậc palindrome (File chương trình: DPALIN.*)

Palindrome là chuỗi ký tự mà đọc từ trái qua phải giống như đọc từ phải qua trái, ví dụ chuỗi 'abba' hoặc 'madam'.

Với chuỗi s bất kỳ, ta có thể tiến hành *chia đôi* nó như sau:

- Nếu s có độ dài bằng 1 hoặc s không phải là palindrome thì không thể chia đôi nó.
- Nếu s là palindrome độ dài n ($n > 1$) thì có thể chia đôi s để nhận được chuỗi gồm $(n+1)$ **div** 2 ký tự đầu của nó.

Ví dụ, các chuỗi 'informatics' và 'a' là không chia đôi được, chia đôi 'abba' được 'ab', chia đôi 'madam' được 'mad', chia đôi 'aaa' được 'aa', chia đôi tiếp được 'a'.

Bậc palindrome (gọi tắt là *bậc*) của chuỗi s , là số lần tối đa có thể áp dụng liên tiếp phép chia đôi nó. Ví dụ, các chuỗi 'informatics' và 'a' có bậc bằng 0 vì không thể áp dụng phép chia đôi một lần nào; các chuỗi 'abba', 'madam' có bậc bằng 1; chuỗi 'aaa' có bậc bằng 2, còn chuỗi 'totottotot' có bậc bằng 3: 'totottotot' \rightarrow 'totot' \rightarrow 'tot' \rightarrow 'to'.

Yêu cầu: Cho trước chuỗi s (chỉ chứa các chữ cái la tinh thường từ 'a' đến 'z'), có độ dài n , xét từ điển $\Omega(s)$ gồm tất cả các chuỗi có cùng độ dài và cùng bậc palindrome với s , các chuỗi của $\Omega(s)$ được đánh số thứ tự từ 1 trở đi. Hãy xác định bậc p và số thứ tự k của s trong $\Omega(s)$. Dữ liệu luôn đảm bảo rằng: $1 \leq n \leq 200$; $0 \leq p \leq 8$; $1 \leq k < 10^{18}$.

Dữ liệu: Vào từ file văn bản DPALIN.INP gồm nhiều dòng, mỗi dòng (là một test) chứa một chuỗi s như trong yêu cầu trên.

Kết quả: Ghi ra file văn bản DPALIN.OUT, gồm nhiều dòng mà dòng thứ i là kết quả tìm được tương ứng với chuỗi thứ i trong file dữ liệu vào, gồm lần lượt 3 số n, p, k (cách nhau bởi ít nhất một dấu cách).

Ví dụ:

DPALIN.INP	DPALIN.OUT
abba	4 1 1
madam	5 1 7804
ahaha	5 2 7
olymp	5 0 6597777
lhp	3 0 7340
totottotot	10 3 490
minh	4 0 216345
byebye	6 0 22919105

Ghi chú:

- Với mỗi bộ 3 số (n, p, k) cần tìm, điểm được chia theo tỉ lệ 1:3:6.
- Giới hạn thời gian: 01 giây cho mỗi file dữ liệu vào.
- 40% số file dữ liệu vào đảm bảo $k < 100.000$.
- 60% số file dữ liệu vào đảm bảo $k < 1000.000$.
- 80% số file dữ liệu vào đảm bảo $k < 1.000.000.000$.