Mã bài: orderenc

(Mã hóa theo thứ tự)

(IFMO Training 24 – DP Selection)

*Time limit: 2s/test*

Hẳn bạn đã nghe đến các thuật toán nén dữ liệu, trong đó phổ biến là kĩ thuật *mã hóa theo tiền tố*. Nếu chưa từng gặp khái niệm này bao giờ, bạn có thể tham khảo hướng dẫn dưới đây:

Mã hóa có thể coi như một ánh xạ F: {c1,…,cn} -> {S1,S2,…,Sn}, trong đó:

1. C1,…,cn ứng với bảng chữ cái
2. S1,…,Sn ứng với các xâu được mã hóa
3. F(ci) = Si với i = 1,…,n
4. Các xâu S1,…,Sn có dạng nhị phân và không có xâu nào là tiền tố của một xâu khác (Điều này đảm bảo tính duy nhất của mã hóa, và khi giải mã, bạn chỉ cần dò từ trái qua phải cho khi gặp được đoạn mã thích hợp)

Có nhiều thuật toán mã hóa theo tiền tố hợp lệ, trong đó, nổi tiếng nhất là thuật toán mã hóa Huffman, cho phép tạo ra đoạn mã với độ dài ngắn nhất có thể. Tuy nhiên, thuật toán Huffman không đảm bảo được thứ tự của các đoạn mã. Cụ thể hơn, bảng chữ cái có thể được sắp xếp theo thứ tự từ điển, trong khi các mã tương ứng thì không. Do đó, trong bài toán này, bạn cần bổ sung thêm một điều kiện mã hóa:

1. Các xâu S1, …, Sn phải có thứ tự từ điển tăng dần

Hiển nhiên, trong phép mã hóa này, chúng ta không cần quan tâm đến nội dung văn bản; thay vào đó chúng ta chỉ cần quan tâm đến tần suất xuất hiện của các kí tự.

Input: 1 số nguyên n ứng với số kí tự xuất hiện, tiếp theo là n số nguyên p1,…,pn (0 < pi <= 10^9) là số lần xuất hiện của kí tự thứ i

Output: n dòng ứng với đoạn mã của từng kí tự trong phương án tối ưu.

Ràng buộc (dễ): 2 <= n <= 200

Ràng buộc (khó): 2 <= n <= 2000

Ví dụ

|  |  |
| --- | --- |
| 5  1 8 2 3 1 | 00  01  10  110  111 |