# Gợi ý giải bài

it.kma

**Problem:**

Liệt kê tất cả các cách phân tích số nguyên dương n thành tổng các số nguyên dương, hai cách phân tích là hoán vị của nhau chỉ tính là một cách.

Test:

| INPUT | OUTPUT |
|---|---|
| 4 | 3+1<br>2+1+1<br>2+2 |

**Solve:**

Đây là một dạng bài tập yêu cầu liệt kê tất cả các cấu hình. Để ý rằng có thể đưa bài toán với đầu vào n về bài toán với đầu vào n-x với một cách phân tích (x,n-x), tương tự, (x,(y,n-x-y)). Quá trình phân tích sẽ dừng lại khi n-x-y-...=1, tức cấu hình cuối cùng là (1,1,..,1). Rõ ràng sẽ tồn tại một cách phân tích trùng nhau ở dạng (x,n-x) và (n-x,n-(n-x)), để tránh điều này, ta duy trì một thứ tự trong cấu hình để đảm báo tính duy nhất.

Để chứng minh một cấu hình có tính thứ tự là duy nhất, ta giả sử tồn tại hai cấu hình có thự tự và cùng phần tử A=(a1,a2,a3…) và B=(b1,b2,b3,..), nếu có một cặp phần tử mà tại đó ai=bi, a[i+1] khác b[i+1], ta có b[i+1]<a[i], mặt khác, a[i+1]<a[i], vậy a[i+1]> b[i+1] hoặc ngược lại. Suy ra tồn tại các xếp các phần tử mà tại đó có đoạn a[i],a[i+1],b[i+1] hoặc a[i],b[i+1],a[i]. Sự tồn tại của một trong hai cách xếp đó chứng tỏ một trong hai cấu hình A=(a1,a2,a3…) hoặc B=(b1,b2,b3,..) không tồn tại, bởi chúng bị sắp xếp sai. Điều này mâu thuẫn với giả thiết, do đó một cấu hình có thứ tự là duy nhất.

Để duyệt qua tất cả các cấu hình từ đầu vào n, ta duyệt các cấu hình (x,n-x) với x chạy từ n-1→n/2+1 (Tại sao? xin để lại như một bài tập nhỏ), tại từng bước, nếu cấu hình đó thỏa mãn sự giảm dần, ta lấy cấu hình đó và tiếp tục duyệt với đầu vào lúc này là x, để được cấu hình con ((y,x-y),n-x)… Mặt khác, nếu cấu hình đang duyệt tới không thỏa mãn sự giảm dần hoặc chạm đến cấu hình cuối cùng, ta dừng và trở về cấu hình cha.

Ví dụ với n=5, màu đỏ biểu thị các cấu hình không thỏa mãn: