

Bài 1. Ảnh đẹp

Yêu cầu bài toán: Tìm đoạn con tổng lớn nhất có độ dài chẵn và độ dài ≥ 4 .

Nhận xét: Giả sử đoạn tổng lớn nhất là đoạn $[u, v]$ ta có tổng

$$a_u + a_{u+1} + \dots + a_v = S_v - S_{u-1}$$

Với $S_i = a_1 + a_2 + \dots + a_i$.

Đặt $p = u - 1$. Khi đó, với mỗi v , ta có giá trị S_p là giá trị nhỏ nhất trong tất cả các $S_p, S_{p-2}, S_{p-4}, S_{p-6}, \dots$

Giải thuật: Gọi f_i là giá trị nhỏ nhất trong tất cả các giá trị $S_i, S_{i-2}, S_{i-4}, S_{i-6}, \dots$

Dễ dàng tính $f_i = \min(f_{i-2}, S_i)$.

Từ đó, với mỗi giá trị v , đoạn con có tổng lớn nhất thỏa mãn có giá trị là $S_v - f_{v-4}$.

Độ phức tạp: $O(n)$



Bài 2. Đội hình thi đấu

Giải thuật: Nguyên lý cực trị

Sắp xếp dữ liệu theo chiều tăng dần của cặp giá trị (b_i, a_i) , $i = 1 \div n$.

Xét tiềm năng của đội k người. Gọi x là sức mạnh của đội trưởng và y – tổng độ dẻo dai của các người còn lại.

Dễ dàng thấy rằng chỉ có thể có 2 cách chọn:

-  Chọn k người đầu tiên trong dãy đã sắp xếp,
-  Chọn $k-1$ người đầu tiên trong dãy đã sắp xếp và một người trong số $n-k$ người (từ vị trí $k+1$ đến n) còn lại làm đội trưởng.

Từ 2 cách chọn trên rút ra đội có tiềm năng nhỏ hơn.

Gọi ia là chỉ số nhỏ nhất thỏa mãn điều kiện $a_{ia} \leq a_i$ với $i = 1 \div n$ trong dãy đã sắp xếp.

Nếu $ia > k$ thì đội cần chọn sẽ bao gồm $k-1$ người đầu tiên trong dãy đã sắp xếp làm thành viên và người thứ ia làm đội trưởng.

Nếu $ia \leq k$ – đội cần chọn sẽ gồm k người đầu tiên. Vấn đề còn lại – xác định đội trưởng.

Gọi s là tổng độ dẻo dai của k người đầu tiên.

$$s = \sum_{i=1}^k b_i$$

$$d = \max\{b_i - a_i, i = 1 \div k\}$$

Tiềm năng cần tìm sẽ là $s - d$.

Các giá trị s và d được cập nhật với chi phí $O(1)$ khi chuyển từ k sang $k+1$.

Tổ chức dữ liệu

- Mảng `vector<pll>` `ba(n)` – lưu các cặp giá trị nguyên 64 bits ($\mathbf{b_i}, \mathbf{a_i}$),
- Mảng `vector<int64_t>` `f(n)` – lưu kết quả cần tìm.

Độ phức tạp của giải thuật: $O(n \log n)$.

Bài 3. Pha lê

Giải thuật:

Xét tất cả các ước của n .

Một số kỹ thuật đếm số lượng phân tử khác nhau trong một đoạn:

- Sort đoạn tăng dần. Đếm số lượng cặp cạnh khác nhau.
- Sử dụng bảng đánh dấu. Đếm số lượng phân tử xuất hiện. Chú ý số lượng này chỉ thay đổi khi 1 số từ trạng thái số lần xuất hiện $0 \leftrightarrow 1$
- Giả sử ta cần tính đoạn $[u, v]$. Gọi $tr[i]$ là vị trí bên trái gần i nhất có giá trị bằng a_i . Số phân tử khác nhau chính là số lượng p thỏa mãn $tr[p] < u$.

Lựa chọn kỹ thuật thứ 2 hoặc thứ 3 để đạt được độ phức tạp $O(n)$ với mỗi lần xét ước.

Tổng độ phức tạp $O(n \times k)$ với k là số ước của n .