

Lời giải

Giả sử có một cách đào các cục vàng theo một thứ tự nào đấy và thỏa mãn đề bài. Trong các cục vàng được đào, gọi sum_A là tổng thời gian đào các cục vàng, max_A là thời gian đào tối đa của một cục vàng, và any_A là thời gian đào của bất kì cục vàng nào khác. Do:

$$sum_A - max_A \leq sum_A - any_A < T$$

Vậy nên ta hoàn toàn có thể đào cục vàng có thời gian tối đa sau cùng mà không làm tệ đi tổng giá trị cách đào này.

Vì thế, thứ tự đào vàng tối ưu là ta đào các cục vàng từ nhỏ đến lớn, tăng dần theo thời gian. Vậy nên ta có thể sắp xếp trước các cục vàng tăng dần theo A_i . Khi đó, ta có thể sử dụng phương pháp quy hoạch động để giải bài toán.

Gọi $dp(i, j)$ là cách đào có giá trị lớn nhất chỉ xét i cục vàng đầu tiên, và tổng thời gian đào hiện tại là j . Chúng ta có trạng thái neo $dp(0, 0) = 0$, và ta có thể chuyển trạng thái như sau:

- Không đào cục vàng thứ i : $dp(i, j) = \max(dp(i, j), dp(i - 1, j))$
- Đào cục vàng thứ i : $dp(i, j) = \max(dp(i, j), dp(i - 1, j - A_i) + B_i)$

Chi tiết cách cài đặt các bạn có thể xem qua code ở thư mục bài giải.

Độ phức tạp: $O(N^2)$

Tag: dp, greedy
