

E: Sequence Partitioning

まず、数値 x を固定したときに、 f の最大値を x 以上にできるかどうかを考えます。

$dp[i]$ を、先頭の部分列 $[1, i]$ について同様に f を定義したとき (これを f_i とする) の f_i の最大値を x 以上にできるかどうかと定義し、 $dp[0] = \text{true}$ と初期化します。

i を小さい順に 1 から N まで見ていく動的計画法を考えます。

いま、位置 i を見ているとします。 $dp[i]$ が true になるには、ある j ($j < i$) が存在して、 $dp[j] = \text{true}$ かつ $b_{j+1} + c_i + \sum_{j < k \leq i} (a_k) \geq x$ を満たす必要があります。

これは愚直に k を全探索すると $O(N)$ かかってしまいます。そこで p_i を $0 \leq j < i$ かつ $dp[j] = \text{true}$ を満たす j についての $b_{j+1} + \sum_{j < k \leq i} (a_k)$ の最大値と定義すると、 $dp[i]$ が true となることは $p_i + c_i \geq x$ が成立することと言い換えられます。また、 p_{i+1} は p_i から更新できます。

以上より x を固定すると $O(N)$ で f の最大値が x 以上にできるかどうかを判定できることが分かったので、 x の値を二分探索することで答えとなる値を求めることができます。