

evantai – Soluție

Problema se rezolvă prin programare dinamică. Fie $B(i, j)$ numărul de subșiruri evantai care au elementele din secvența $[i, j]$ a șirului A și folosesc elementele A_i și A_j . O recurență imediată (care este practic pasul inițial din soluția finală) este următoarea:

$$B(i, j) = 1 + \text{suma } B(k, l) \text{ cu } i < k, l < j \text{ și } A_i + A_j > A_k + A_l$$

Implementarea acestei soluții obține aproximativ 30% din teste și are complexitatea $O(N^4)$. Pentru a optimiza ideea vom folosi o structură de date 2D ce ne permite aflarea sumei de pe o porțiune dreptunghiulară în timp sub-liniar. Ideea reiese din următorul pseudo-cod:

```
pentru fiecare i de la N la 1
    *pentru fiecare j de la i+1 la N
         $B(i, j) = 1 + \text{query}(j-1, A_i + A_j - 1)$ 
    *reactualizează informația în structura de date;
```

Prin operația **query** aflăm $\text{suma } B(k, l)$ cu $i < k, l < j$ și $A_i + A_j > A_k + A_l$.

Implementarea operațiilor **query** se poate realiza utilizând o structură de date denumită arbori indexați binar care suportă interogări de forma aceasta. Complexitatea totală va fi $O(N^2 * \log(N) * \log(\max))$ unde \max reprezintă valoarea maximă din șirul A .

Există soluții de complexitate $O(N^2 * \max)$ și $O(N^3 * \log(N))$ care obțin punctaje parțiale.