

3B – Mopadulo

autor zadania: Kamil Dębowski

Treść

Dany jest ciąg liczb całkowitych. Należy policzyć (modulo $10^9 + 7$) sumę jego dobrych podziałów na przedziały. Podział ciągu na przedziały jest dobry, jeśli reszta z dzielenia sumy liczb w każdym z przedziałów przez $10^9 + 7$ jest parzysta.

Rozwiązanie

Dla zadanego ciągu $a_1, a_2, \dots, a_n$ policzmy ciąg jego sum prefiksowych, a dokładniej ich reszt z dzielenia przez $10^9 + 7$. Oznaczmy go $pref_0, pref_1, \dots, pref_n$, gdzie $pref_0 = 0$ i $pref_i = (pref_{i-1} + a_i) \bmod (10^9 + 7)$. Zastanówmy się, jaka jest reszta z dzielenia przez $10^9 + 7$ sumy liczb z przedziału $[\ell, r]$. Jeśli zachodzi $pref_r \geq pref_{\ell-1}$, to suma ta to po prostu $pref_r - pref_{\ell-1}$. Jeśli zaś zachodzi $pref_r < pref_{\ell-1}$, to policzona w ten sposób suma okazałaby się ujemna, musimy ją zatem zastąpić przez $pref_r - pref_{\ell-1} + 10^9 + 7$.

Łatwo zatem wywnioskować kiedy reszta z dzielenia sumy liczb z jakiegoś przedziału jest parzysta. Dla przedziału $[\ell, r]$, jeśli zachodzi $pref_r \geq pref_{\ell-1}$, to musi zachodzić $pref_r \equiv pref_{\ell-1} \pmod{2}$. Jeśli zaś zachodzi $pref_r < pref_{\ell-1}$, to musi zachodzić $pref_r \not\equiv pref_{\ell-1} \pmod{2}$.

Użyjmy więc programowania dynamicznego. Niech $DP[i]$ oznacza liczbę podziałów na przedziały prefiksu długości i . Zaczynamy oczywiście od $DP[0] = 1$. Aby policzyć $DP[i]$ przeiterujmy się po początku ostatniego przedziału w podziale. Niech zaczyna się na pozycji j . Jest on poprawny wtedy i tylko wtedy, gdy reszta z dzielenia sumy liczb w nim jest parzysta. Musimy zatem wtedy zwiększyć $DP[i]$ o $DP[j - 1]$. Gdybyśmy zaimplementowali przejścia brutalnie, skończylibyśmy ze złożonością kwadratową, która zdecydowanie nas nie zadowala.

Z zebranymi obserwacjami łatwo jest jednak wywnioskować, co należy zrobić. Skoro by obliczyć $DP[i]$ musimy poznać sumę po $DP[j]$ dla takich j , że $pref_j \leq pref_i$ oraz $pref_i \equiv pref_j \pmod{2}$, to możemy w tym celu użyć drzewa przedziałowego. Po przeskalowaniu współrzędnych (którymi są sumy prefiksowe) pozwoli nam ono odczytywać sumę wartości z dowolnego prefiksu. Ponieważ interesuje nas jedynie suma dla wartości o zadanej parzystości, to możemy zbudować dwa drzewa przedziałowe – jedno dla wartości parzystych i jedno dla nieparzystych.

Dzięki temu będziemy mogli pytać zarówno o prefiks wartości o zadanej parzystości, jak i sufiks wartości o przeciwnej parzystości, a także wprowadzać zmiany (dodawać nowo policzone $DP[i]$ na pozycji $pref_i$). Każda z tych operacji będzie standardową operacją na drzewie przedziałowym. Zamiast drzewa przedziałowego, w celu uproszczenia kodu, możemy również użyć drzewa Fenwicka (zwanego również BIT lub drzewem potęgowym). W obu wariantach, ze względu na przeskalowanie oraz użycie odpowiedniej struktury danych, kończymy ze złożonością $\mathcal{O}(n \cdot \log(n))$.