

1B – Oranżada

autor zadania: Mateusz Radecki

Treść

W zadaniu mamy dany ciąg liczb $a_1, a_2, \dots, a_n$ oraz liczbę k . W jednym ruchu możemy zamienić dwa sąsiednie elementy ciągu miejscami. Jesteśmy pytane ile co najmniej ruchów potrzebujemy, aby pierwsze k elementów ciągu było parami różnych.

Rozwiązanie

Jeśli w ciągu jest mniej niż k różnych wartości, to nie da się osiągnąć celu, odpowiadamy zatem -1 .

W przeciwnym wypadku spójrzmy na zbiór pozycji, z których elementy przemieszczają się na jedną z k pierwszych pozycji. Oznaczmy posortowany ciąg tych pozycji przez $p_1, p_2, \dots, p_k$. Dla $i \neq j$ musi zachodzić $a_{p_i} \neq a_{p_j}$. Przemieszczenie jakiegoś elementu z pozycji i na pozycję j zajmuje nam $|i - j|$ ruchów. Okazuje się, że element z pozycji p_i opłaca się doprowadzić na pozycję i .

Dlaczego? Jeśli uprzemy się przy tej kolejności, to będziemy mogli doprowadzać elementy na miejsce jeden po drugim, idąc od lewej. Będziemy na to potrzebować $\sum_{i=1}^n |p_i - i|$ ruchów, a ponieważ musi zachodzić $p_i \geq i$, to $\sum_{i=1}^n |p_i - i| = \sum_{i=1}^n (p_i - i) = (\sum_{i=1}^n p_i) - (\sum_{i=1}^n i)$. Zauważmy, że przy każdym ruchu suma pozycji wybranych elementów może spadać co najwyżej o 1. Musimy zmniejszyć tę sumę o $(\sum_{i=1}^n p_i) - (\sum_{i=1}^n i)$, a skoro we wspomnianym sposobie zmniejszamy ją zawsze o 1, to nasze rozwiązanie jest optymalne.

Musimy zatem znaleźć takie k pozycji, których suma jest minimalna i wszystkie wartości, które się na nich znajdują, są parami różne. Łatwo zrobić to zachłannie. Możemy przejść się po ciągu od lewej i jeśli jakiejś wartości jeszcze nie widzieliśmy, to właśnie jej pozycję dodajemy do rozwiązania. Idziemy po ciągu tak długo aż spotkamy k różnych wartości.

Czemu jest to optymalne? Powiedzmy, że pierwszy raz napotykamy wartość w . Jeśli mielibyśmy wziąć do wyniku jakieś inne jej wystąpienie, to możemy poprawić wynik biorąc zamiast niego pierwsze wystąpienie w . Jeśli jednak mielibyśmy wziąć do rozwiązania całkiem inną wartość występującą później, to również możemy zamienić ją na pierwsze wystąpienie w .

Ponieważ jesteśmy w stanie zarezerwować tablicę, która pomoże nam śledzić które wartości już napotkaliśmy, to całkowita złożoność algorytmu to $\mathcal{O}(n)$.