

### **Задача 3. Сортиране - Анализ**

Автор: Владимир Владимиров

За решението на задачата ще обоснова и доразширя алгоритъма, описан доста подробно във условието.

Какво гласи той? Нека сме отделили последните няколко елемента на масива  $A[N]$  чрез функцията  $reverse(x)$  и те са на окончателните си места. Ако броят им е  $k$ , то разглеждаме редицата с начало  $0$  и край  $N-k-1$ . Сега трябва да поставим точния елемент на позиция  $N-k-1$ , а понеже редицата е сортирана възходящо следва, че елементът, който би трябвало да стои на позиция  $N-k-1$  е най-големият измежду елементите  $A[0], A[1] \dots A[N-k-1]$ . Нека това е  $A[x]$ . Тогава имаме три възможности как да го сложим на точната позиция:

1. Ако  $x=N-k-1$  тогава  $A[x]$  си е на мястото. Продължаваме да повтаряме горната стъпка, като полагаме за дясна граница  $N-k-2$ .
2. Ако  $x=0$  тогава е достатъчно да приложим  $reverse(N-k-1)$  и елементът ще отиде на крайната си позиция.
3. В противен случай прилагаме  $reverse(x)$ , за да отиде елементът на позиция  $0$  в масива и оттам извикваме  $reverse(N-k-1)$ , за да го преместим на окончателната му позиция.

Продължаваме да прилагаме тези стъпки до достигане на крайната последователност или до  $k=N-1$  (от това че последните  $X-1$  елемента са си на мястото и са по-големи от  $A[0]$  и са вдясно от него, следва че получената последователност е сортирана правилно. Естествено, трябва да се съобрази, че за решението на задачата следва първоначално да се положи  $k=0$ . Бихме могли да добавим флаг  $f=0$ , който да указва дали на някоя итерация е извикана функцията  $reverse(x)$  и ако да, да го увеличим. След края на алгоритъма ако все още  $f=0$ , това означава, че редицата е била предварително сортирана и следва да се изведе нула. Ако  $f>0$  следва, че сме извиквали  $reverse()$  и съответно сме извели резултата.

Сложността на гореописаното решение е  $O(N^2)$ . Имаме  $N$  на брой итерации на външния цикъл и намирането на максималния елемент за всяка подредица се извършва с линейна сложност, т.е. нашия алгоритъм е с обща алгоритмична сложност  $O(N^2)$ , където  $N$  е броя на елементите в редицата. Предвид ограниченията това решение е напълно достатъчно, за да решава вярно задачата.