# ЗАДАЧИ ДЛЯ ПЕРВОГО СЕМЕСТРА

Тема 2. Задачи на обработку целочисленного массива.

*В следующих задачах требуется написать функцию, получающую в качестве параметров имя массива и его длину (n). Заполнение массива числами из файла и определение его фактической длины должно выполняться отдельной подпрограммой. Требуется создать две функции чтения массива из файла: первая предполагает, что в файле задано количество чисел и сами числа, а во второй количество чисел задается, собственно, их количеством в файле (количество чисел в файле отдельным числом не задается). При решении задач следует в одних случаях использовать первую функцию, а в других вторую.*

*Функция main открывает файл, вызывает функцию, заполняющую массив числами из файла, обращается к функции печати массива на экран, вызывает функцию, преобразующую массив (или вычисляющую его характеристику) и выводит результат (или преобразованный массив) на экран.*

*В задачах 1-22 предполагается, что количество чисел в массиве (как в исходном, так и в преобразованном ) не превосходит некоторой фиксированной величины (например, 1000).*

*В остальных задачах память под хранения массива должна выделяться динамически, количество чисел в файле заранее неизвестно.*

*Файл нельзя открывать более одного раза, нельзя возвращаться к началу файла. Использовать дополнительные массивы нельзя. Ошибки (файл не удалось открыть, он пуст, наличие в файле буквы, реальное количество чисел в файле не равно заданному числу (для варианта 1 чтения массива из файла)) обработать.*

*Обозначим через $n$ количество чисел в массиве, а элемент с номером $i$ – через $x_i$, $0 \le i \le n-1$.*

**Участком** *массива назовем множество подряд идущих элементов $x_k$, $i \le k \le j$, индексы $k$ которых удовлетворяют условию $i \le k \le j$ для некоторых $1 \le i, j \le n$.* **Длиной участка** *назовем количество содержащихся в нем элементов (т.е. число $j-i+1$).*

**Постоянным участком** *массива называется такой его участок, $\{x_k\}_{k=i,\dots,j}, i < j$, что все его элементы равны между собой и для которых выполняются соотношения: 1)$i = 0$ или $x_{i-1} \ne x_i$; 2)$j = n-1$ или $x_{i+1} \ne x_j$.*

**Максимум (минимум)** *массива – это максимальный (минимальный) элемент среди всех $x_i$, $0 \le i \le n-1$ (т.е. глобальный максимум или минимум). Максимальный (минимальный) элемент – это элемент, равный максимуму (минимуму).*

**$k$-максимальным** *($k$-минимальным) значением массива назовем число $x$, определенное индуктивно по $k$: если $k = 1$, то $x$ – максимум (минимум) массива. Если $k > 1$ и $x$ равно $(k-1)$-максимальному (минимальному) значению массива, то максимум (минимум) из тех элементов массива, которые меньше (больше), чем $x$, будем называть $k$-максимальным (минимальным) значением. $k$-максимальный элемент массива – это элемент, имеющий $k$- максимальное значение.*

**Локальным максимумом** *массива называется участок $x_k$, $i \le k \le j$, все элементы которого равны между собой, для которого выполняются соотношения:*

*1) $i = 0$ или $x_{i-1} < x_i$;*

*2)$j = n-1$ или $x_{j+1} < x_j$.*

*Аналогично определяется* **локальный минимум.**

*Массив называется* **возрастающим***, если для всех $i = 0, \dots, n-2$ выполняется соотношение: $x_i \le x_{i+1}$.*

*Массив называется* **строго возрастающим***, если для всех $i = 1, \dots, n-2$ выполняется соотношение: $x_i < x_{i+1}$.*

*Аналогично определяются* **убывающий и строго убывающий** *массивы.*

**1.** Функция должна возвращать 1, если массив симметричный и 0 в противном случае.

**2.** Функция должна так преобразовать массив, чтобы его элементы шли в обратном порядке. Использовать не более чем $n/2$ обменов элементов.

**3.** Функция должна так преобразовать массив, чтобы каждый его элемент, кроме первого и последнего, заменился на полусумму соседей.

**4.** Функция должна циклически сдвинуть элементы массива на одну позицию вправо используя не более чем $n$ перемещений элементов (n - длина массива).

**5.** Функция должна циклически сдвинуть элементы массива на одну позицию влево используя не более чем $n$ перемещений элементов (n - длина массива).

**6.** Функция должна поменять местами элементы массива с соседними номерами (т.е. с номерами $2i$ и $2i+1$ для всех возможных $i$), используя не более чем $n/2$ обменов элементов ($n$ - длина массива).

**7.** Функция должна так преобразовать массив, чтобы его положительные элементы шли в обратном порядке. Использовать не более чем $n/2$ обменов элементов.

**8.** Функция должна так преобразовать массив, чтобы его отрицательные элементы шли в обратном порядке. Использовать не более чем $n/2$ обменов элементов.

**9.** Функция должна продублировать (рядом) каждый отрицательный элемент массива. Использовать не более чем $2n$ перемещений элементов.

**10.** Нормировать элементы массива на интервал $[min\{a_0, ..., a_{n-1}\}, max\{a_0, ..., a_{n-1}\}]$, т.е. в случае, когда все элементы массива совпадают, изменять массив не следует, иначе заменить каждый элемент массива на значение $min\{a_0, ..., a_{n-1}\} + (a_i - min\{a_0, ..., a_{n-1}\})/(max\{a_0, ..., a_{n-1}\} - min\{a_0, ..., a_{n-1}\})$

**11.** Заменить, если это возможно, все минимальные элементы массива на значение элемента массива, большего минимального, но не большего всех остальных элементов массива.

**12.** Функция должна циклически сдвинуть элементы массива на $k$ позиций вправо, используя не более чем $n$ обменов элементов ($n$ – длина массива).

**13.** Циклически сдвинуть на одну позицию влево постоянные участки. Так, массив (1 1 1 4 1 0 0 0 0 6 2 2 2 0 0) должен быть преобразован в (0 0 0 0 4 1 2 2 2 6 0 0 1 1 1). Количество операций присваивания элементов массива не более $n$.

**14.** Циклически сдвинуть на одну позицию вправо постоянные участки. Так, массив (1 1 1 4 1 0 0 0 0 6 2 2 2 0 0) должен быть преобразован в (0 0 4 1 1 1 1 6 0 0 0 0 2 2 2). Количество операций присваивания элементов массива не более $n$.

**15.** Функция должна поменять местами постоянные участки массива с соседними номерами (т.е.с номерами $2i$ и $2i+1$ для всех возможных $i$), используя не более чем $n/2$ обменов элементов (n - длина массива). Так, массив (1 1 1 4 1 0 0 0 0 6 2 2 2 0 0) должен быть преобразован в (0 0 0 0 4 1 1 1 1 6 0 0 2 2 2). Количество операций присваивания элементов массива не более $n$.

**16** Для заданного числа $k < n$ поменять местами начальный участок массива (элементы с индексами, меньшими $k$) с его конечным участком (остальные). Взаимный порядок элементов внутри участков измениться не должен. Так, массив (1 1 4 1 0 0 6 1 1 2 0 0) для $k = 4$ должен быть преобразован в ( 0 0 6 1 2 0 0 0 1 1 4 1).
Использовать не более чем $n$ обменов элементов.

**17.** Сравнить два неупорядоченных целочисленных массива $A$ и $B$ как числовые множества: $A = B$ и $A \subset B$. Повторы элементов в массиве не учитывать (сравнение множеств). Так, если массив A= (1 0 2 2 0 0 3 3 3 0 0 0) а B= (3 3 3 0 1 0 0 2 4 0 0 0), то $A \neq B$ и $A \subset B$.

**18.** Назовем $x$-отрезком группу подряд идущих элементов массива, каждый из которых равен $x$. Для заданного числа $x$ заменить элементы каждого $x$-отрезка на полусумму элементов, прилегающих к этому отрезку справа и слева. Если $x$-отрезок расположен в начале или конце массива, считать недостающий крайний элемент равным нулю. Так, массив (1 1 4 1 0 0 6 1 1 0 0 0) для $x = 1$ должен быть преобразован в (2 2 4 2 0 0 6 3 3 0 0 0).

**19.** Заменить все локальные минимумы в массиве одним элементом, значение которого равно 0. Функция должна возвращать количество элементов в преобразованном массиве. Так, массив (2 1 2 2 0 0 3 3 3 0 0 0) должен быть преобразован в (2 0 2 2 0 3 3 3 0). Использовать не более чем $n$ перемещений элементов массива.

**20.** Заменить каждый элемент массива колличеством элементов массива с меньшими индексами, имеющими значение, меньше данного элемента, т.е. каждый элемент массива $a_i$ заменить количеством элементов $a_j$, таких что $j < i$ и $a_j < a_i$. Так, массив (1 10 2 2 0 0 3 3 3 0 1 2) должен быть преобразован в (0 1 1 1 0 0 5 5 5 0 3 5).

**21.** Сравнить два неупорядоченных целочисленных массива $A$ и $B$ как числовые мультимножества: $A = B$ и $A \subset B$. Учитывать повторы элементов в массиве. Так, если массив A= (1 0 2 2 0 0 3 3 3 0 0 0) а B= (3 3 3 0 1 0 0 2 4 0 0 0), то $A \neq B$ и $A \notin B$.

**22.** Найти $k$-максимальное значение массива, $k$ задано. Нельзя изменять элементы массива.

**23.** В каждом участке строгого возрастания в массиве заменить все элементы на среднее значение участка (рассматриваются участки строгого возрастания, которые нельзя расширить, т.е. максимальные по включению). Так, массив (1 0 2 4 0 0 3 3 3 0 10 20) должен быть преобразован в (1 2 2 2 0 1 1 3 3 10 10 10) (среднее в смысле целого).

**24.** Поменять местами локальные максимумы с соседними номерами (т.е. с номерами $2i$ м $2i+1$ для всех возможных $i$), используя не более чем $n/2$ обменов элементов ($n$ - длина массива). Так, массив (1 0 2 2 0 0 3 3 3 0 0 0) должен быть преобразован в (2 2 0 1 0 0 3 3 3 0 0 0). Количество операций присваивания элементов массива не более $n$.

**25.** Сдвинуть циклически локальные минимумы на одну позицию влево. Так, массив (1 10 2 2 10 10 3 3 3 10 10 10) должен быть преобразован в (2 2 10 3 3 3 10 10 1 10 10 10). Количество операций присваивания элементов массива не более $n$.

**26.** Сдвинуть циклически локальные максимумы на одну позицию вправо. Так, массив (1 0 2 2 0 0 3 3 3 0 0 0) должен быть преобразован в (3 3 3 0 1 0 0 2 2 0 0 0). Количество операций присваивания элементов массива порядка $O(n)$.

**27.** Пусть элементы массива не убывают. Двоичным поиском определить, между какими индексами можно вставить заданное число $x$, не нарушив порядок. Проверять, что массив неубывающий, при считывании. В случае, если массив неубывающим не является, вернуть ошибку.

**28.** Пусть в массиве последовательно записаны цифры некоторого длинного неотрицательного десятичного числа. Реализовать функции "прибавление единицы" и "вычитание единицы" из такого числа. Вычитание свести к сложению (с числом (0-1)). Размер массива может измениться. Так, массив (9 9 9 9) при сложении с 1 должен быть преобразован в (1 0 0 0 0), а массив (0 0 0 ) при вычитании 1 станет равным (9 9 9).

**30.** Функция должна отрицательные элементы массива переместить в начало с сохранением их порядка, а положительные – в конец. Количество обменов элементов массива не должно быть более чем $n$, где $n$ – размер массива.

**31.** Функция должна отрицательные элементы массива переместить в начало, а положительные – в конец с сохранением их (положительных элементов) порядка. Количество обменов элементов массива не должно быть более чем $n$, где $n$ – размер массива.

**32.** Функция должна положительные элементы массива переместить в начало, а отрицательные – в конец. Количество перемещений элементов не должно более чем $n$, где $n$ —- размер массива.

**33.** Даны 2 массива. Функция должна возвращать количество вхождений одного массива в другой (как участок последовательности, в том числе пересекающихся вхождений).

**34.** Найти минимальное число $k$ такое, что модуль суммы всех $l$-максимальных значений по всем $l \leq k$ равен модулю суммы всех $l$-минимальных (по всем $l \leq k$) Нельзя изменять элементы массива.

**35.** Найти такое $k$, что сумма $l$-максимальных значений по всем $l \geq k$ будет максимально возможной. Нельзя изменять элементы массива.

**36.** Найти такое $k$, что сумма $l$-минимальныхх значений по всем $l \geq k$ будет минимально возможной. Нельзя изменять элементы массива.

**37.** Определить, сколько элементов массива имеют $k$-максимальное значение ($k$ задано). Нельзя изменять элементы массива.

**38.** Найти, если существует, такое $k$, что модуль суммы всех $l$-максимальных элементов массива по всем $l \geq k$ равен модулю суммы всех $l$-максимальных элементов по $l \leq k$.

**39.** В массиве A хранится перестановка $\sigma(i) = A[i]$ чисел $0, \ldots, n-1$. Определить ее четность. Каждый элемент просматривается не более O(1) раз. Если массив не является перестановкой, вывести ответ как ошибку. Так, массив A= (1 0 2 3 4 6 5) определяет четную перестановку.

В задачах 40-43 требуется вычислить значение функции $f_n(a_0, ..., a_{n-1})$, заданной индуктивно. На последовательности, состоящей из одного элемента, она принимает значение, равное этому элементу, $f_1(a) = a$. Формула для вычисления $f_2(a, b)$ известна (выделить ее отдельной функцией). Решение не должно использовать рекурсивные вызовы.

**40.** Для $n > 2$ $f_n(a_0, \ldots, a_{n-1}) = f_2(a_0, f_{n-1}(a_1, \ldots, a_{n-1}))$.

**41.** Для $n > 2$ $f_n(a_0, \ldots, a_{n-1}) = f_{n-1}(f_2(a_0, a_1), a_2 \ldots, a_{n-1})$.

**42.** Для $n > 2$ $f_n(a_0, \ldots, a_{n-1}) = f_{n-1}(f_2(a_0, a_{n-1}), a_1, \ldots, a_{n-2})$.

**43.** Для $n > 2$ $f_n(a_0, \ldots, a_{n-1}) = f_{n-1}(a_1, \ldots, a_{n-2}, f_2(a_0, a_{n-1}))$.

**44.** В массиве хранится перестановка $\sigma(i) = A[i]$ чисел $1, \ldots, n$. Не используя дополнительных массивов, найти обратную перестановку. Так, обратной к массиву A= (1 0 2 3 4 6 5) будет перестановка (1 0 2 3 4 6 5). Если массив не является перестановкой, вывести ответ как ошибку.

**45.** Дано число $m <= n$. Для каждого участка массива из $m$ элементов найти сумму $m$ стоящих рядом элементов. Количество сложений переменных типа int не более $n$. Результат сохранить в исходном массиве. Так, массив (1 0 2 3 1 -1 -1 2 3 0 0 0 1 2 3 4) для $m = 3$ должен быть преобразован в (3 5 6 3 -1 0 4 5 3 0 1 3 6 9).

**46.** В 2-х массивах хранятся коэффициенты многочленов (от одной переменной, степени могут быть различны). Поместить в третий массив коэффициенты произведения многочленов.

**47.** Даны два неубывающих массива. Соединить их в третий, тоже неубывающий массив. Число действий – порядка суммы размеров исходных массивов. Проверять, что массив неубывающий, при считывании. В случае, если один из массивов неубывающим не является, вернуть ошибку.

**48.** Даны два неубывающих массива. Пересечь их в третий тоже неубывающий массив. Число действий – порядка суммы размеров исходных массивов. Проверять, что массив неубывающий, при считывании. В случае, если один из массивов неубывающим не является, вернуть ошибку.