

Задача 7. Одномерные массивы (только чтение)

Для вычислений использовать отдельную функцию `solution` (необходимые параметры и возвращаемое значение определяются условием конкретного варианта задачи). В некоторых задачах потребуются также дополнительные функции.

Заранее придумать не менее 10 различных тестов, охватывающих как типичные, так и все возможные граничные (наиболее невероятные и показательные) ситуации. Программа при запуске должна прогонять эти тесты (для каждого теста распечатывать входной массив и результат), а также предлагать пользователю задать произвольный массив в качестве входных данных.

Для ввода и, если необходимо, вывода одномерных массивов разработать отдельные функции.

Варианты:

1. Вводится массив целых чисел. Посчитать, какое количество элементов в правой половине массива, которые больше или равны 1) какого-либо любого элемента в левой половине массива, 2) всех элементов в левой половине массива. В случае нечетного кол-ва элементов в массиве средний элемент относится к левой половине массива.
2. Вводится массив целых чисел. Найти максимальное кол-во ненулевых элементов, расположенных между нулевыми элементами массива.
3. Вводится массив целых чисел. Найти позицию последнего наименьшего элемента массива.
4. Вводится массив целых чисел. Посчитать сумму элементов, расположенных между последним минимальным элементом и первым максимальным.
5. Вводится массив целых чисел. Посчитать сумму элементов между последним четным и последним нечетным элементом.
6. Вводится массив целых чисел. Определить позицию первого отрицательного элемента в четной позиции или, что такого элемента не существует (для этого реализовать функцию, которая будет возвращать позицию найденного элемента или -1 в случае его отсутствия).
7. Вводится массив целых чисел. Найти, какое максимальное количество одинаковых элементов в массиве. Например, для массива {1, 2, 7, 12, 2, 7, 2, 3, 4, 5, 12, 7, 8} ответ 3 (3 элемента со значением 2 и 3 элемента со значением 7).
8. Вводится массив целых чисел. Найти максимальное количество подряд идущих одинаковых элементов.
9. Вводится массив целых чисел. Найти максимальное кол-во подряд идущих элементов, которые монотонно возрастают или монотонно убывают.
10. Вводится массив целых чисел. Найти максимальную сумму подряд идущих элементов. Примечание: существует алгоритм, как можно получить такую сумму в один проход — постарайтесь догадаться.

11. Вводится массив целых чисел. Найти максимальную сумму подряд идущих элементов, образующих арифметическую прогрессию. Будем считать, что два любых элемента (или один для массива из одного элемента) всегда образуют такую прогрессию. Учесть, что прогрессия может быть как положительной, так и отрицательной. Также учесть, что элементы в массиве могут быть как положительные, так и отрицательные.
12. Вводится массив целых чисел. Найти наиболее длинную подпоследовательность подряд идущих элементов, которые монотонно возрастают (реализовать функцию, которая будет возвращать позицию первого элемента такой подпоследовательности и кол-во элементов). Если таких подпоследовательностей одинаковой длины несколько – найти последнюю.
13. Вводится массив целых чисел. Найти в данном массиве кол-во интервалов убывания (т.е. подряд идущих элементов, которые монотонно убывают). Например, для массива {1, 3, -2, 11, 7, 3, 3, 2, 4, 2, 7, 12} ответ равен 4 ({3, -2}, {11, 7, 3}, {3, 2} и {4, 2}).
14. Вводится массив целых чисел. Найти позицию последнего элемента массива, второго по величине (второй по величине — минимальный элемент, больший самого минимального) или определить, что такого элемента не существует (для массива, состоящего из всех одинаковых элементов).
15. Вводится массив целых чисел. Посчитать количество элементов массива, больших или равных среднему арифметическому элементов массива, отличных от минимального (если все элементы массива одинаковые, то ответ – 0).
16. Вводится массив целых чисел. Найти позицию последнего элемента массива, который граничит с максимальным или минимальным элементом массива, однако сам не является минимальным или максимальным элементом.
17. Вводится массив целых чисел. Найти количество элементов массива, которые максимально близки к среднему арифметическому для всех элементов массива. Например, для массива {1, 4, 2, 5, 8, 4, 1, 3} ответ равен 3 (среднее арифметическое – 3.5, максимально близкие элементы со значением 3 и 4 – оба отличаются на 0.5).
18. Вводится массив целых чисел. Посчитать для массива количество локальных максимумов и минимумов. Локальный минимум – слева и справа (или только слева или только справа для первого и последнего элемента) расположены элементы с большим значением. Локальный максимум – аналогично.
19. (*) Вводится массив целых чисел. Найти самую длинную подпоследовательность подряд идущих элементов массива, такую что в данной подпоследовательности четные и нечетные (по модулю) элементы чередуются (реализовать функцию, которая будет возвращать позицию первого элемента такой подпоследовательности и кол-во элементов). Будем считать, что один любой элемент массива является последовательностью чередующихся четных и нечетных элементов. В случае нескольких таких подпоследовательностей найти последнюю. Например, для массиве {1, 2, -2, 1, 10, 3, 105, 4, -5, 4, 10, 1, 1, 3, 4, 7, 12, 10} правильным ответом будет {3, 4, 7, 12} (также существуют еще две таких последовательности длиной 4 – {-2, 1, 10, 3} и {105, 4, -5, 4}).
20. (*) Вводится массив целых чисел. Найти наиболее длинную подпоследовательность подряд идущих элементов, которые чередуются (очевидно, что любые два подряд

идущих элемента чередуются; реализовать функцию, которая будет возвращать позицию первого элемента такой подпоследовательности и кол-во элементов). В случае нескольких таких подпоследовательностей вывести вторую по счету справа. Для массива { 4, 6, 1, 2, 1, 2, 3, 2, 3, 5, 4, 7, 4, 1, 5, 1, 5, 6} правильным ответом будет {2, 3, 2, 3}.

21. (*) Вводится массив целых чисел. Найти самую длинную подпоследовательностей подряд идущих элементов массива, которые в сумме дают 0 (реализовать функцию, которая будет возвращать позицию первого элемента такой подпоследовательности и кол-во элементов). В случае нескольких таких подпоследовательностей найти вторую слева (от начала массива). В случае, если таких подпоследовательностей не существует, функция должна возвращать в качестве первого элемента подпоследовательности -1.
22. Вводится массив целых чисел. Найти позицию второго с конца элемента, который встречается в массиве наиболее часто. Если несколько значений встречаются в массиве одинаковое кол-во раз выбрать наибольший по абсолютному значению, затем просто по значению. Если такого элемента нет (все элементы массива разные), то вернуть -1.
23. Вводится массив целых чисел. Найти количество элементов в массиве, которые можно получить сложением любых 2-х других элементов массива. Для массива { 1, 0, 2, 4, 3, 10, 8, 2, 3 } правильным ответом будет 6 ($2_{[2]} = 0_{[1]} + 2_{[7]}$, $4_{[3]} = 2_{[2]} + 2_{[7]}$, $3_{[4]} = 1_{[0]} + 2_{[2]}$, $10_{[5]} = 2_{[2]} + 8_{[6]}$, $2_{[7]} = 0_{[1]} + 2_{[2]}$, $3_{[8]} = 1_{[0]} + 2_{[2]}$).
24. Вводится массив целых чисел. Посчитать сумму элементов массива, которые являются факториалом какого-либо числа. Для проверки, является ли число факториалом другого числа, реализовать функцию. Учесть, что при вычислении факториала числа n для больших n возможно переполнение при вычислениях в целых числах.
25. Вводится массив целых чисел. Найти сумму четных элементов массива, расположенных между первым простым (по модулю) элементом массива и последним простым (по модулю) элементом. Для проверки простоты числа реализовать отдельную функцию. Ноль считать простым числом. Подумать, как реализовать данный алгоритм в один проход (т.е. один цикл; цикл, который, вероятно, будет использован в функции проверки простоты числа не считаем).
26. Вводится массив целых чисел. Найти последнюю максимальную по сумме подпоследовательность подряд идущих элементов длины n (n – задается). Реализовать в виде функции, которая будет возвращать позицию первого элемента найденной подпоследовательности. В случае, если n больше длины массива, функция должна вернуть -1. Задача должна быть решена в один проход (один цикл).
27. Вводится массив целых чисел. Найти минимальный элемент, расположенный между последним максимальным элементом массива и n-ым четным элементом (n – задается, счет для n начинается с 1). Реализовать в виде функции, которая будет возвращать индекс найденного элемента. Если такой элемент найти нельзя (нет n-го четного элемента, n-ый четный элемент совпадает или расположен рядом с последним максимальным элементом), то функция должна вернуть -1.
28. (*) Вводится массив целых чисел. Найти наиболее длинную подпоследовательность подряд идущих элементов такую, что любые два элемента данной подпоследовательности отличаются друг от друга не более, чем на величину a (a – задается; реализовать функцию, которая будет возвращать позицию первого элемента

такой подпоследовательности и кол-во элементов). В существования нескольких таких последовательностей вывести вторую по счету слева.

29. (*) Вводится массив целых чисел. Найти последнюю самую длинную подпоследовательность полностью различных (ни один элемент не равен никакому другому элементу) подряд идущих элементов массива (реализовать функцию, которая будет возвращать позицию первого элемента такой подпоследовательности и кол-во элементов).
30. (*) Вводится массив целых чисел. Найти наиболее длинную «ямую» в последовательности. Под «ямой» понимается подпоследовательность подряд идущих элементов, которые сначала убывают (или равны), затем возрастают (или равны). Для левой границы массива под «ямой» также понимаются подряд идущие элементы, которые возрастают, а для правой, которые убывают. Для массива {3, 3, 4, 8, 8, 10, 7, 4, 10, 7, 7, 4, 3, 3, 8, 9, 1} «ямами» будут подпоследовательности {3, 3, 4, 8, 8, 10}, {10, 7, 4, 10}, {10, 7, 7, 4, 3, 3, 8, 9}, {9, 1} (самая длинная – {10, 7, 7, 4, 3, 3, 8, 9}). А например, массив {10, 5, 5, 3} будет одной большой «ямой» (для правой границы).
31. Вводится массив целых чисел. Найти наиболее длинную симметричную подпоследовательность подряд идущих элементов (реализовать функцию, которая будет возвращать позицию первого элемента такой подпоследовательности и кол-во элементов). В случае нескольких таких подпоследовательностей найти самую первую. Например, для массива {5, 1, 3, 1, 3, 5, 1, 5, 3, 7, 8, 8, 9, 8, 8, 5, 1} правильным ответом будет подпоследовательность {3, 5, 1, 5, 3}.
32. (*) Вводится массив целых чисел. Найти наиболее длинную подпоследовательность подряд идущих элементов такую, что все элементы данной подпоследовательности равны за исключением 2-х произвольных элементов (реализовать функцию, которая будет возвращать позицию первого элемента такой подпоследовательности и кол-во элементов). В случае нескольких таких подпоследовательностей найти самую первую. Например, для массива {4, 5, 3, 3, 7, 3, 3, 7, 6, 4, 6, 7, 7, 7, 7, 1} правильным ответом будет {5, 3, 3, 7, 3, 3} (также существует еще три таких подпоследовательности длиной 6 – {3, 3, 7, 3, 3, 7}, {4, 6, 7, 7, 7, 7} и {6, 7, 7, 7, 7, 1}).
33. (*) Вводится массив целых чисел. Найти такую подпоследовательность длины n (n – задается) подряд идущих элементов массива, которая максимально часто повторяется в исходном массиве (реализовать функцию, которая будет возвращать позицию первого элемента такой подпоследовательности). Если таких подпоследовательностей несколько, вернуть самую первую. При подсчете повторений, учесть, что эти самые повторения не должны накладываться друг на друга. Например, для массива {-3, 7, 3, 4, 8, 4, 8, 4, 7, 3, 4, 8, 0, 6, -1, 4, 7, 3, 3} и $n = 3$ две подпоследовательности длины n повторяются по 2 раза – {7, 3, 4} и {3, 4, 8} (стоит обратить внимание, что с учетом непересечений {4, 8, 4} повторяется только 1 раз). Таким образом правильным ответом будет {7, 3, 4} (функция вернет 1 – индекс первого элемента первого повторения данной подпоследовательности).
- 34.