

Практическое занятие №15

Сортировка массивов

Цель: Изучить алгоритмы сортировки массивов и научиться использовать их при обработке данных.

Порядок выполнения работы

Теоретический материал

В качестве практической проблемы, требующей решения, рассматривается известная задача сортировки (упорядочивания) массива в порядке возрастания его элементов. При решении этой задачи требуется исходный массив, содержащий произвольные целые числа, преобразовать к виду, когда каждый элемент массива находится перед другим элементом этого массива, если его значение меньше (больше), чем значение сравниваемого элемента.

В данной лабораторной работе необходимо изучить ряд известных алгоритмов сортировки и создать комплекс программ, реализующий:

- метод сортировки выбором;
- метод сортировки пузырьком;
- метод сортировки включением.

Метод сортировки выбором

Исходный массив длиной N разбивается на две части: итог и остаток. Участок массива, называемый **итогом**, располагается с начала массива и должен быть упорядоченным, а участок массива, называемый **остатком**, располагается вплотную за итогом и содержит исходные числа не отсортированной части исходного массива.

Текстуальный алгоритм сортировки выбором:

Шаг 1. Полагается $i=0$, т.е. считается, что итоговый участок - пуст.

Шаг 2. В остатке массива ищется минимальный элемент и он меняется местом с первым элементом остатка (i -ым элементом массива). После чего значение i увеличивается на единицу, тем самым расширяя итоговый участок массива (отсортированную часть исходного массива).

Шаг 3. Если $i < N$, то повторяется Шаг 2. В противном случае - конец алгоритма, т.к. итог становится равным всему массиву.

Конец алгоритма.

Схема алгоритма методом сортировки выбора представлена на рис. 1.

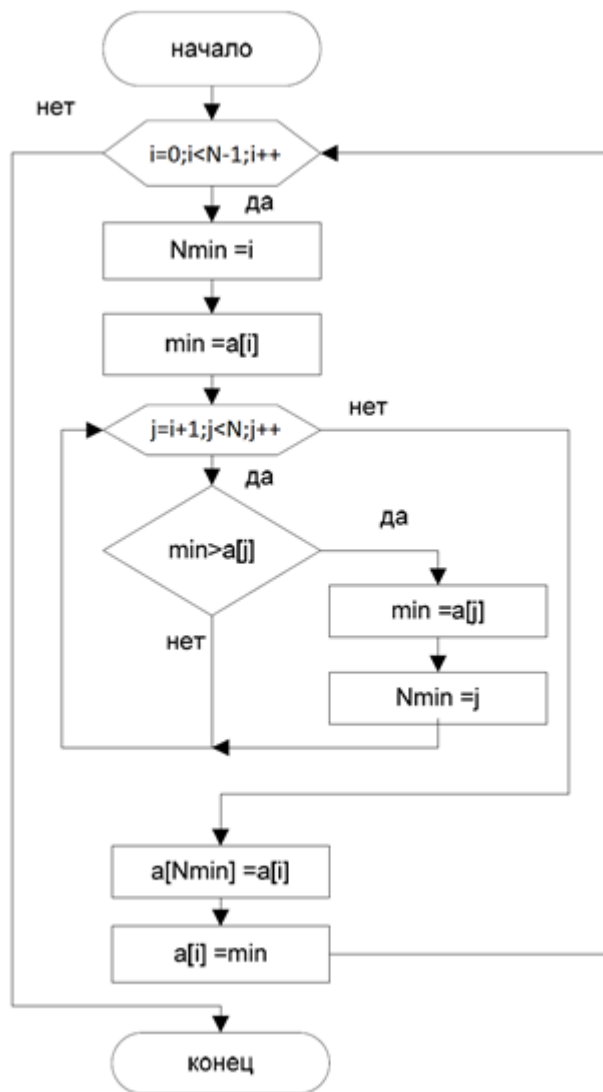


Рисунок 1. Схема алгоритма сортировки методом выбора

Метод сортировки пузырька

Аналогично, как и в методе выбора, исходный массив длиной N разбивается на две части: отсортированную (итог) и не отсортированную (остаток). Первый элемент остатка является i -ым элементом массива.

Текстуальный алгоритм сортировки пузырьком:

Шаг 1. Пусть $k=N-1$, т.е. итоговый участок состоит из одного элемента.

Шаг 2. Берется первый элемент остатка и перемещается на место в итоговый участок массива так, чтобы итог остался упорядоченным. Первый элемент остатка назовем перемещаемым. Перемещение выполняется путем сравнения перемещаемого элемента с последующим элементом. Если последующий элемент больше сравниваемого элемента, то процесс перемещения этого элемента закончен.

Шаг 3. После того, как первый элемент остатка переместился в итоговый участок, уменьшается на единицу значение переменной k , тем самым увеличивая отсортированную часть массива. Если $k>0$, то управление передается на Шаг 2, в противном случае - работа алгоритма завершена.

Конец алгоритма.

Схема алгоритма метода сортировки пузырька представлена на рис. 2.

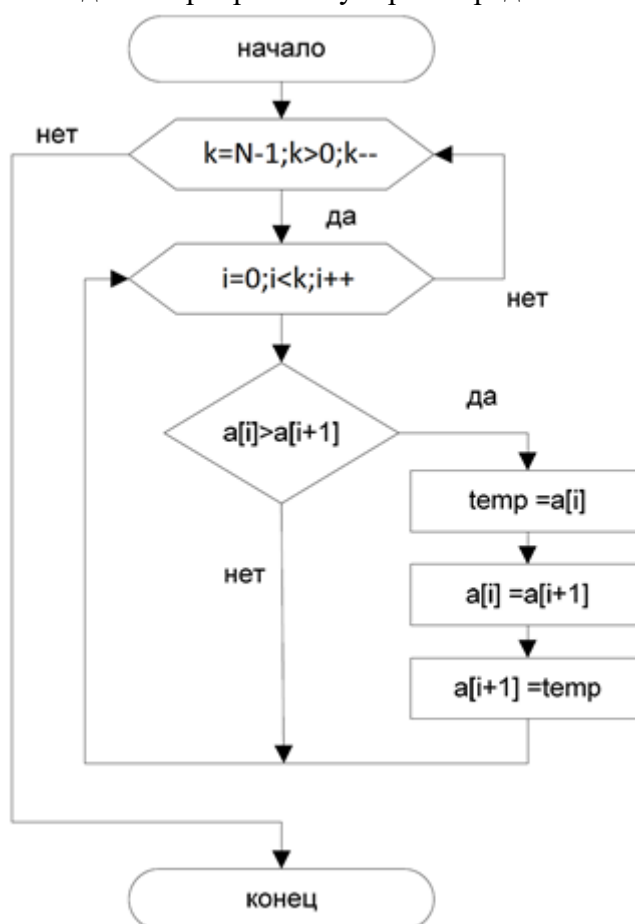


Рисунок 2. Блок-схема алгоритма сортировки методом пузырька

Метод сортировки включением

Этот метод похож на метод пузырька. Происходит такое же разбиение массива на отсортированную и не отсортированную части, но перемещение первого элемента остатка на принадлежащее ему место в итоге делается не сравнением двух соседних элементов, а с помощью метода двоичного поиска, который удобно оформить в виде отдельной процедуры.

Текстуальный алгоритм методом включением:

Шаг 1. Пусть $i=1$, т.е. итоговый участок состоит из одного элемента.

Шаг 2. Берется первый элемент остатка и перемещается в отсортированную часть массива так, чтобы итоговый участок остался упорядоченным.

Шаг 3. После того, как первый элемент остатка переместился в итоговый участок, увеличивается на единицу значение переменной i , тем самым увеличивая отсортированную часть массива. Если $i < N$, то управление передается на Шаг 2, в противном случае - работа алгоритма завершена.

Конец алгоритма.

Схема алгоритма методом сортировки включением представлена на рис. 3.

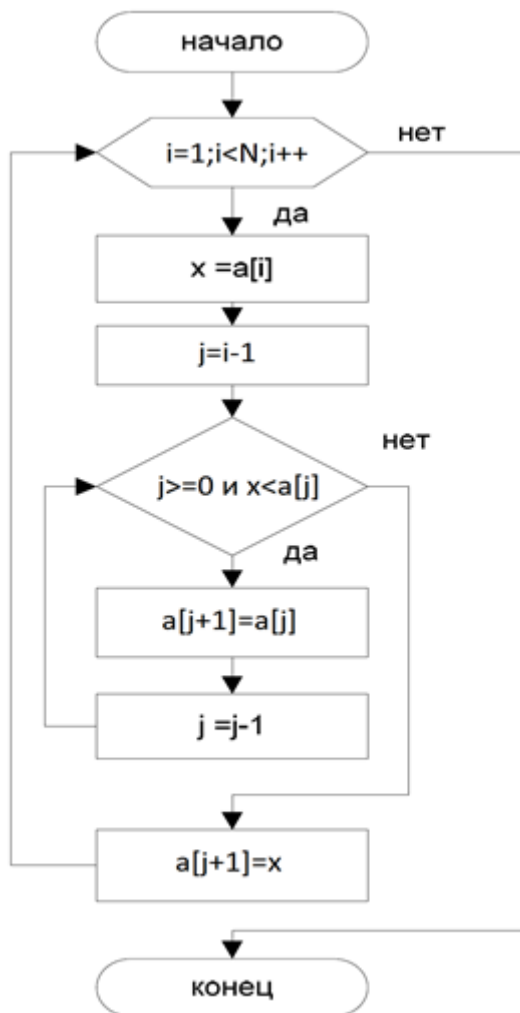


Рисунок 3. Блок-схема алгоритма прямым включением

Задание:

Используя один из изученных алгоритмов сортировки, составить программы на C#:

1. Выполнить сортировку только четных элементов массива (нечетные элементы остаются на своих местах).
2. Выполнить сортировку элементов, записанных на нечетных местах.

Контрольные вопросы:

1. Что контролируют обработчики исключений `FormatException` и `OutOfMemoryException`?
2. Как выделяется место в памяти под элементы массива?
3. Как расположены элементы массива в памяти?
4. Как осуществить заполнение массива случайным образом?
5. Перечислите основные методы, используемые при обработке массивов данных.