



Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное  
учреждение высшего образования  
«Московский государственный технический университет имени  
Н. Э. Баумана  
(национальный исследовательский университет)»  
(МГТУ им. Н. Э. Баумана)

---

ФАКУЛЬТЕТ «Информатика и системы управления»

---

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

---

## ОТЧЕТ

по Лабораторной работе №2

по курсу «Анализ Алгоритмов»

на тему: «Алгоритмы умножения матриц»

Студент группы ИУ7-56Б

\_\_\_\_\_  
(Подпись, дата)

Мамврийский И. С.  
\_\_\_\_\_  
(Фамилия И.О.)

Преподаватель

\_\_\_\_\_  
(Подпись, дата)

Волкова Л. Л.  
\_\_\_\_\_  
(Фамилия И.О.)

Преподаватель

\_\_\_\_\_  
(Подпись, дата)

Строганов Ю. В..  
\_\_\_\_\_  
(Фамилия И.О.)

Москва — 2023 г.

# Содержание

<b>Введение</b>	<b>3</b>
<b>1 Аналитическая часть</b>	<b>5</b>
1.1 Пирамидальная сортировка . . . . .	5
1.2 Сортировка перемешиванием . . . . .	6
1.3 Блинная сортировка . . . . .	7
<b>2 Конструкторская часть</b>	<b>8</b>
2.1 Разработка алгоритмов . . . . .	8

# Введение

Одной из важнейших процедур обработки структурированной информации является сортировка.

Сортировка - процесс перегруппировки заданной последовательности (кортежа) объектов в некотором определенном порядке. Такой определенный порядок позволяет, в некоторых случаях, эффективнее и удобнее работать с заданной последовательностью. В частности, одной из целей сортировки является облегчение задачи поиска элемента в отсортированном множестве.

Алгоритмы сортировки используются практически в любой программной системе. Целью алгоритмов сортировки является упорядочение последовательности элементов данных. Поиск элемента в последовательности отсортированных данных занимает время, пропорциональное логарифму количеству элементов в последовательности, а поиск элемента в последовательности не отсортированных данных занимает время, пропорциональное количеству элементов в последовательности, то есть намного больше. Существует множество различных методов сортировки данных. Однако любой алгоритм сортировки можно разбить на три основные части:

- сравнение, определяющее упорядочность пары элементов;
- перестановка, меняющая местами пару элементов;
- собственно сортирующий алгоритм, который осуществляет сравнение и перестановку элементов данных до тех пор, пока все эти элементы не будут упорядочены.

Одной из важнейшей характеристик любого алгоритма сортировки является скорость его работы, которая определяется функциональной зависимостью среднего времени сортировки последовательностей элементов данных, определенной длины, от этой длины.

Задачи данной лабораторной:

- изучить и реализовать три алгоритма сортировки: блонная, пирамидальная, перемешиванием;

- Выбрать инструменты для замера процессорного времени выполнения реализаций алгоритмов;
- Создать ПО, реализующее алгоритмы сортировки, указанные в варианте;
- Провести анализ затрат работы программы по времени и по памяти, выяснить влияющие на них характеристики;

# 1 Аналитическая часть

В этом разделе будут представлены описания алгоритмов сортировки пирамидальная, блинная, перемешиванием.

## 1.1 Пирамидальная сортировка

Пирамида (англ. binary heap) определяется как структура данных, представляющая собой объект-массив, который можно рассматривать как почти полное бинарное дерево. Каждый узел этого дерева соответствует определенному элементу массива. На всех уровнях, кроме, может быть, последнего, дерево полностью заполнено (заполненным считается уровень, который содержит максимально возможное количество узлов). Последний уровень заполняется слева направо до тех пор, пока в массиве не закончатся элементы.

В пирамиде, представленной на рисунке 1.1, число в окружности, представляющей каждый узел дерева, является значением, сортируемым в данном узле. Число над узлом — это соответствующий индекс массива. Линии, попарно соединяющие элементы массива, обозначают взаимосвязь вида “родитель-потомок”. Родительские элементы всегда расположены слева от дочерних. Данное дерево имеет высоту, равную 3; узел с индексом 4 (и значением 8) расположен на первом уровне.

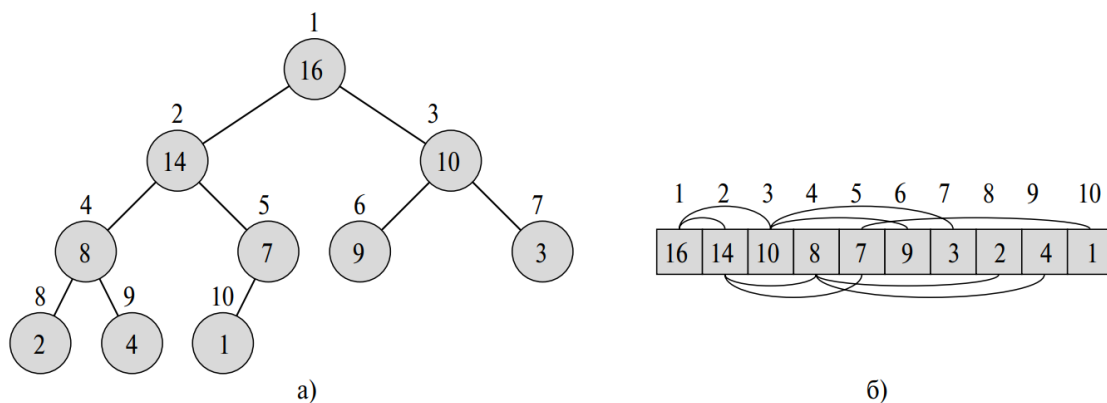


Рисунок 1.1 – Пирамида, представленная в виде а) бинарного дерева и б) массива

Обозначим, что:

- 1) Родительский элемент в массиве будет находится под индексом -  $\frac{i}{2}$ ;
- 2) Правый потомок в массиве будет находится под индексом -  $2 \cdot i$ ;
- 3) Левый потомок в массиве будет находится под индексом -  $2 \cdot i + 1$ .

Различают два вида бинарных пирамид: неубывающие и невозрастающие. В пирамидах обоих видов значения, расположенные в узлах, удовлетворяют свойству пирамиды (heap property), являющемуся отличительной чертой пирамиды того или иного вида. Свойство невозрастающих пирамид (max-heap property) заключается в том, что для каждого отличного от корневого узла с индексом  $i$  выполняется следующее неравенство

$$A[i/2] \geq A[i] \quad (1.1)$$

Другими словами, значение узла не превышает значение родительского по отношению к нему узла. Таким образом, в невозрастающей пирамиде самый большой элемент находится в корне дерева, а значения узлов поддерева, берущего начало в каком-то элементе, не превышают значения самого этого элемента. Принцип организации неубывающей пирамиды (min-heap) прямо противоположный. Свойство неубывающих пирамид (min-heap property) заключается в том, что для всех отличных от корневого узлов с индексом  $i$  выполняется такое неравенство:

$$A[i/2] \leq A[i] \quad (1.2)$$

Таким образом, наименьший элемент такой пирамиды находится в ее корне.

## 1.2 Сортировка перемешиванием

**Алгоритм сортировки перемешиванием** является модификацией алгоритма сортировки пузырьком. В отличие от сортировки пузырьком, где происходит обход последовательности только в одном направлении, в

алгоритме сортировки перемешиванием после достижения одной из границ рабочей части последовательности (то есть той части, которая еще не отсортирована и в которой происходит смена элементов) меняет направление движения. При этом при движении в одном направлении алгоритм перемещает к границе рабочей области максимальный элемент, а в другом направлении – минимальный элемент. Границы рабочей части последовательности устанавливаются в месте последнего обмена.

## 1.3 Блинная сортировка

**Блинная сортировка (pancake sort)** – алгоритм сортировки массива, в котором сортировка осуществляется переворотом части массива.

В этом алгоритме, к массиву, позволено применять только одну операцию – переворот части массива. И в отличие от других методов сортировки, где пытаются уменьшить количество сравнений, в этом нужно минимизировать количество переворотов.

Идея алгоритма заключается в том, чтобы за каждый проход, переместить максимальный элемент в конец массива.

## Вывод

В данной работе необходимо реализовать алгоритмы сортировки, описанные в данном разделе, а также провести их теоритическую оценку и проверить ее экспериментально.

## 2 Конструкторская часть

В данном разделе будут рассмотрены схемы алгоритмов сортировок (блинная, перемешиванием и пирамидальная), а также найдена их трудоемкость.

### 2.1 Разработка алгоритмов

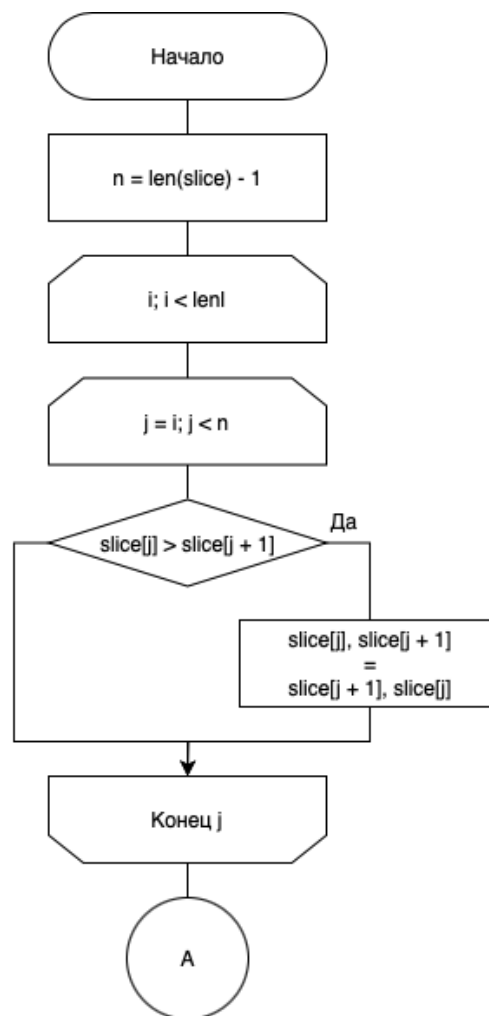


Рисунок 2.1 – Схема алгоритма сортировки Перемешиванием



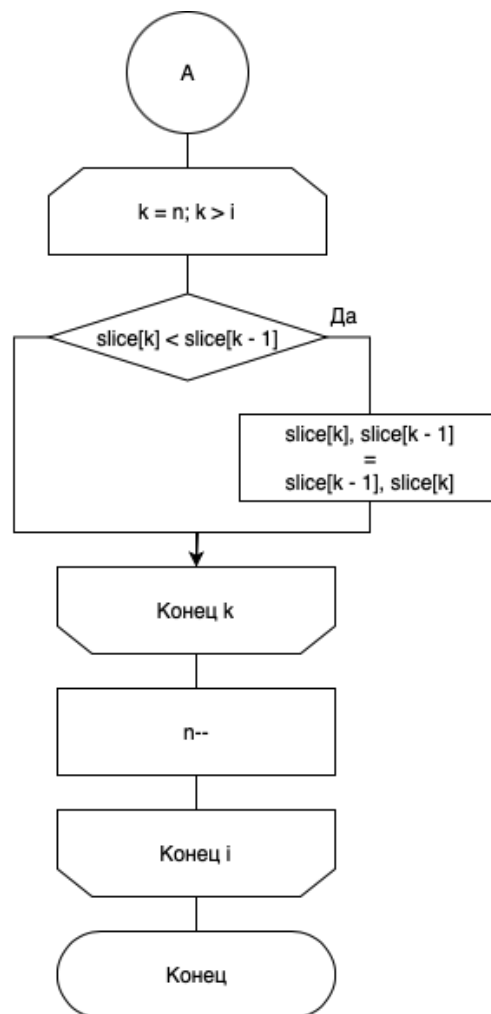


Рисунок 2.2 – Схема алгоритма сортировки Перемешиванием

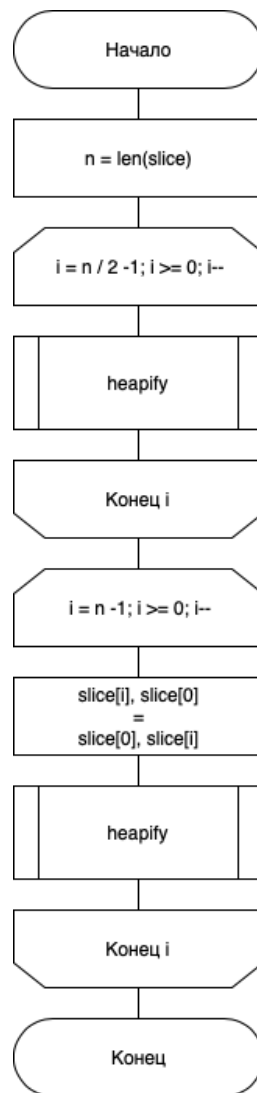


Рисунок 2.3 – Схема алгоритма Пирамидальной сортировки. Основная функция.

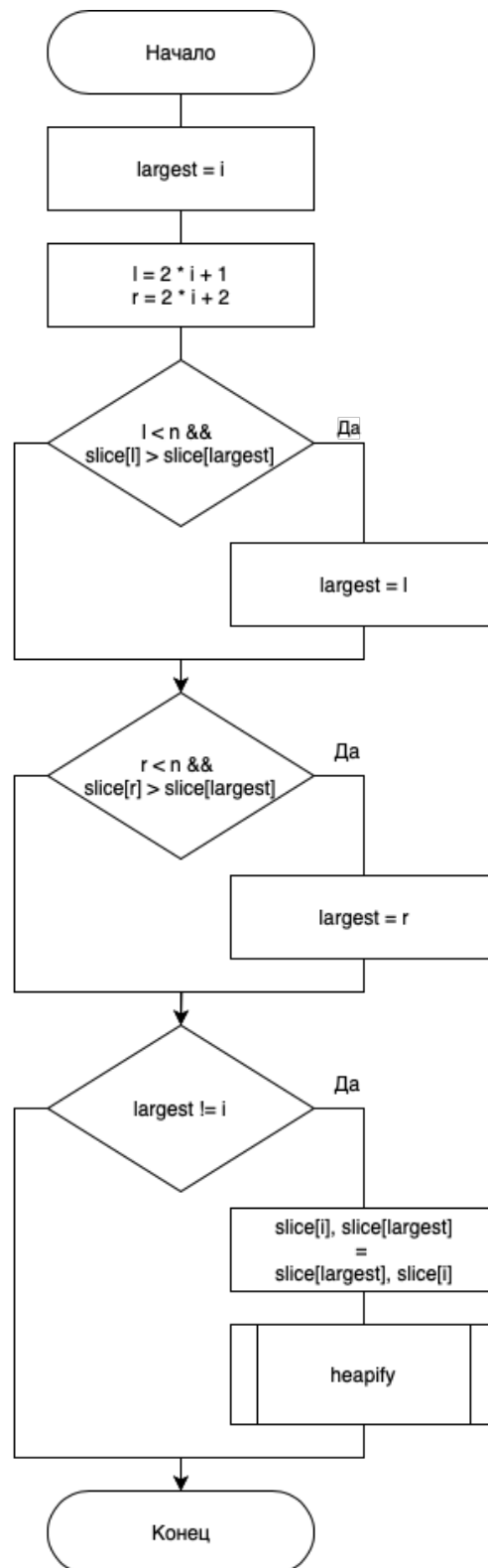


Рисунок 2.4 – Схема алгоритма Пирамидальной сортировки. Функция поиска максимального.