

ПРАКТИЧЕСКАЯ РАБОТА №1

ЭМПИРИЧЕСКИЙ АНАЛИЗ СЛОЖНОСТИ АЛГОРИТМОВ

Часть 1.2: ТЕОРЕТИЧЕСКИЙ МЕТОД ОЦЕНКИ СЛОЖНОСТИ
ИТЕРАТИВНОГО АЛГОРИТМА.

КОРРЕКТНОСТЬ ИТЕРАТИВНОГО АЛГОРИТМА

Цель: актуализация знаний и приобретение практических умений и навыков по определению вычислительной сложности алгоритмов (эмпирический подход).

Задание 1. Определить эффективный алгоритм из двух предложенных в задании 1 практической работы №1.1, используя оценку теоретической сложности каждого из алгоритмов.

Напомним задачу: дан массив из n элементов, удалить из массива все значения, равные заданному пользователем (ключевому).

Два алгоритма решения этой задачи:

x-массив, n – количество элементов в массиве, key – удаляемое значение	
<p>Алгоритм 1. delFirstMetod(x,n,key) { $i \leftarrow 1$ while ($i \leq n$) do if $x[i] = \text{key}$ then //удаление for $j \leftarrow i$ to $n-1$ do $x[j] \leftarrow x[j+1]$ od $n \leftarrow n-1$ else $i \leftarrow i+1$ endif od }</p>	<p>Алгоритм 2 delOtherMetod(x,n,key) { $j \leftarrow 1$ for $i \leftarrow 1$ to n do $x[j] = x[i]$; if $x[i] \neq \text{key}$ then $j++$ endif od if $x[1] = \text{key}$ then $n \leftarrow 0$ else $n \leftarrow j$ endif }</p>

Требования к выполнению задания 1:

1. Для каждого алгоритма привести в отчёте следующие результаты:

1.1. Для каждого из алгоритмов определить:

- На основе анализа псевдокода – формулу *функции роста* времени работы этих алгоритмов $T(n)$ в их лучшем и худшем случаях.
- По полученным функциям роста – количество T_T критических операций при $n = 10, 100, 1000, 10000$.
- Порядок роста* (скорость роста) функции роста.

- 1.2. Представить результаты исследования в таблицах для двух алгоритмов (по примеру табл. 1), указав для каждого n : T_T – количество критических операций согласно теоретическим расчетам и T_n – практически полученное количество операций при выполнении алгоритма (по счетчикам, по данным предыдущей практической работы №1.1).

Таблица 1. Сводная таблица результатов

n	Все элементы ключевые		Все элементы неключевые	
	T_T	T_n	T_T	T_n
10				
100				
1000				
10000				

Примечание: в указанных случаях T_T должно быть равно T_n .

- Сравните результаты подсчета команд, вычисленные здесь по формулам, с результатами по счетчикам в практической работе №1.1. убедитесь в том, что они равны.
- Определите ёмкостную сложность алгоритмов как функции объёма дополнительной памяти от объёма входа.
- Сделайте в отчёте вывод об эффективности алгоритмов на основе теоретического подхода к оценке.

Задание 2. Оценить теоретически вычислительную сложность алгоритма простой сортировки массива.

- Вывести на основе псевдокода или кода C++ для любого из двух алгоритмов простой сортировки массива (согласно варианту индивидуального задания практической работы №1.1) формулу функции роста времени работы алгоритма от объёма входа для лучшего и худшего случаев, а также оценить скорость роста функций роста.
- На основе функций роста рассчитать количество критических операций для объёма входа $n = 100, 1000, 10000, 100000$ и 1000000 элементов. Полученные результаты свести в сводную таблицу по образцу табл. 1 (для ситуаций а) уже упорядоченного массива и б) обратной упорядоченности).
- Сравните результаты подсчета команд, вычисленные здесь по формулам, с результатами по счетчикам в практической работе №1.1. убедитесь в том, что они равны.
- Определите ёмкостную сложность алгоритма сортировки как функцию объёма дополнительной памяти от объёма входа.
- Сделайте вывод об эмпирической вычислительной сложности алгоритма.

Задание 3. Докажите корректность алгоритма 2 из задания 1 практической работы №1.1 с помощью инварианта цикла:

- 1) Сформулируйте инвариант цикла, отражающий желаемый результат.
- 2) Докажите истинность инварианта на этапах инициализации и сохранения. Что приведёт к выводу об истинности инварианта после последней итерации (этап завершения).
- 3) Докажите конечность цикла с помощью области неопределённости.

По результатам выполнения всех заданий сформируйте отчёт, сохраните его в формате pdf и загрузите в систему СДО.

ЛИТЕРАТУРА:

1. Бхаргава А. Грожаем алгоритмы, 2-е изд. – СПб: Питер, 2024. – 352 с.
2. Вирт Н. Алгоритмы + структуры данных = программы. – М.: Мир, 1985. – 406 с.
3. Кнут Д.Э. Искусство программирования, том 3. Сортировка и поиск, 2-е изд. – М.: ООО «И.Д. Вильямс», 2018. – 832 с.
4. Седжвик Р. Фундаментальные алгоритмы на C++. Анализ/Структуры данных/Сортировка/Поиск. – К.: Издательство «Диасофт», 2001. – 688 с.
5. Алгоритмы – всё об алгоритмах / Хабр [Электронный ресурс]. URL: <https://habr.com/ru/hub/algorithms/> (дата обращения 05.02.2025).