Лабораторная работа № 1 "Измерение времени выполнения алгоритма"

Цели: Выработать навыки оценки вычислительной сложности алгоритма. **Задачи:** В лабораторной работе необходимо реализовать заданный алгоритм (см. таблицу ниже) и исследовать его вычислительную сложность. Алгоритм выбирается в соответствии с вариантом задания, полученным от преподавателя.

Вариант	Алгоритм/задача		
1	Случайные блуждания без самопересечений		
2	Поиск в массиве размера N смежного подмассива размера M с		
	наибольшим средним значением элементов, посредством		
	перебора всех подмассивов.		
3	Сортировка выбором массива размера N		
4	Исключение одинаковых элементов массива размера N		
5	Определение количества "особых" элементов в матрице размера N × N, таких что в строке слева от "особого" элемента находятся элементы меньшие его, а справа — большие.		
6	В непустой последовательности слов, слова разделяются запятой, за последним словом стоит точка. Среди всех пар a_i и b_i , где a_i — первая, b_i — последняя буквы i — го слова последовательности, определить наиболее встречающуюся.		
7	Сортировка перемешиванием массива размера N		
8	Поиск в массиве размера N количества "троек" сумма которых равна 0.		
9	Генерация всех возможных N-разрядных двоичных чисел		
10	Сортировка расчёской массива размера N		
11	Сортировка пузырьком массива размера N		
12	Определение количества седловых точек в матрице размера N×N. Элемент матрицы называется седловым, если он является наименьшим в своей строке и одновременно наибольшим в своем столбце.		
13	Определить элементы в матрице $ a_{ij} $ размера N×N, как <i>true</i> , если значения і и ј являются взаимно простыми и <i>false</i> – в противном случае.		
14	Найти строку матрицы с максимальной суммой элементов		
15	Поиск в целочисленном массиве размера N пары с наиболее близкими значениями.		

Порядок работы:

- 1. Согласно варианту на ЯП (Java/Python/C#/...) в виде консольного приложения реализовать заданный алгоритм, убедиться в его корректности;
- 2. Доработать приложение так, чтобы на вход алгоритма подавались входные данные разного размера *N*, и для каждого размера измерялось время выполнения алгоритма. Полученная таблица с результатами (размер/время) должна сохраняться в файл в формате CSV.
- 3. открыть полученный CSV-файл в электронной таблице (MS Office Excel, OpenOffice Calc или подобной) и построить график зависимости времени выполнения алгоритма от размера входных данных;
- 4. выполнить теоретическую оценку эффективности алгоритма;
- 5. сделать выводы насчет соответствия полученных результатов теоретической оценке.

Методические указания

Изучение зависимости времени работы алгоритма от размера входных данных следует проводить следующим образом. Выбираются: начальное значение N_{start} , конечное значение N_{end} , приращение ΔN , количество повторений q. Далее перебираются все значения N, начиная с N_{start} до N_{end} с шагом ΔN , и для каждого N алгоритм выполняется q раз. Полученные q20% результатов сортируются ПО возрастанию, отбрасываются максимальных значений. a ДЛЯ остальных 80% ищется среднее Полученное арифметическое. значение считается временем работы алгоритма на входных данных размера N.

Для измерения времени работы фрагмента кода следует использовать встроенные функции ЯП.

Результат измерений сохраняется в CSV-файле (см. Википедия энциклопедия. — http://ru.wikipedia.org/wiki/CSV). Этот текстовый формат предназначен для представления табличных данных. Каждая строка файла соответствует строке таблицы. Значения отдельных ячеек таблицы разделяются символом «;» (точка с запятой).

На рис. 1а приведен пример таблицы, а на рис. 16 — ее представление в формате CSV.

			Size;Time
	Size	Time	10;0,1
	10	0,1	20;0,4
	20	0,4	30;0,9
	30	0,9	
a)			б)

Рисунок 1 Пример таблицы (a) и ее представление в формате CSV (б)

Вопросы для самоконтроля

- 1. Назовите ресурсы, потребление которых обычно представляет интерес при исследовании алгоритмов.
- 2. Какие факторы влияют на время работы алгоритма?
- 3. Поясните, почему при анализе алгоритмов обычно пользуются асимптотическими оценками, а не точными величинами.
- 4. Сформулируйте определения асимптотических оценок О, Θ и Ω . Дайте асимптотическую оценку функции $T(n) = n\log_3 n + 7n^2 + 15n + 63$.
- 5. Пусть некоторую задачу алгоритм «**A**» решает за время $O(n^3)$, алгоритм «**B**» за время $O(1,02^n)$, алгоритм «**C**» за время $O(n^{2,58})$, а алгоритм «**D**» за время O(n!). Расположите алгоритмы в порядке от самого быстрого к самому медленному.
- 6. Напишите псевдокод алгоритма поиска максимального и минимального элементов на побочной диагонали матрицы размера $N \times N$. Оцените время его работы.

Содержание отчета

Отчет по лабораторной работе должен содержать:

- 1. титульный лист установленного образца с указанными ФИО студента;
- 2. постановку задачи;
- 3. псевдокод исследуемого алгоритма;
- 4. теоретическую оценку времени его работы с необходимыми пояснениями;
- 5. описание процесса исследования, полученные графики, выводы.