



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 № 6 по курсу «Анализ алгоритмов»

Тема Поиск по словарю

Студент Косарев А.А.

Группа ИУ7-51Б

Оценка (баллы)

Преподаватель Волкова Л. Л., Строганов Ю.В.

Содержание

Введение	3
1 Аналитическая часть	4
1.1 Цель и задачи	4
1.2 Структура данных «словарь»	4
1.3 Алгоритм полного перебора	5
1.4 Формализация объектов и его признаков	6
1.5 Анкетирование респондентов	6
1.6 Функция принадлежности термам числовых значений признака	7
2 Конструкторская часть	9
2.1 Описание используемых типов данных	9
2.2 Исходные файлы программы	9
3 Технологическая часть	10
3.1 Выбор языка программирования и среды разработки	10
3.2 Тестирование	10
4 Исследовательская часть	12
4.1 Технические характеристики устройства	12
4.2 Примеры работы программы	12
Заключение	14
Список использованных источников	15

Введение

Словарь — абстрактный тип данных, позволяющий хранить набор значений, обращение к которым происходит по ключам. Ключи должны допускать сравнение друг с другом. Примеры словарей достаточно разнообразны. Например, обычный толковый словарь хранит определения слов (являющиеся значениями), сопоставленные с самими словами (являющимися ключами), а банковская база данных может хранить данные клиентов, сопоставленные с номерами счетов.

Одной из основных операций в словаре является поиск значения по ключу.

1 Аналитическая часть

1.1 Цель и задачи

Цель — получение навыка поиска по словарю при ограничении на значение признака, заданном при помощи лингвистической переменной.

Для достижения поставленной цели необходимо выполнить следующие задачи:

- формализовать объект и его признак;
- провести анкетирование респондентов;
- построить функцию принадлежности термам числовых значений признака, описываемого лингвистической переменной, на основе статистической обработки мнений респондентов, выступающих в роли экспертов;
- описать алгоритм поиска в словаре объектов;
- описать структуру данных словаря;
- реализовать описанный алгоритм поиска в словаре.

1.2 Структура данных «словарь»

Словарь [2] — абстрактный тип данных, позволяющий хранить пары вида (ключ, значение) и поддерживающий операции добавления пары, а также поиска и удаления пары по ключу:

- 1) $insert(k, v)$;
- 2) $find(k)$;
- 3) $remove(k)$.

В паре (k, v) : v называется значением, ассоциированным с ключом k . Где k — это ключ, а v — значение. Семантика и названия вышеупомянутых операций в разных реализациях ассоциативного массива могут отличаться.

Операция поиска $find(k)$ возвращает значение, ассоциированное с заданным ключом, или некоторый специальный объект, означающий, что значения, ассоциированного с заданным ключом, нет. Две другие операции ничего не возвращают.

Словарь с точки зрения интерфейса удобно рассматривать как обычный массив, в котором в качестве индексов можно использовать не только целые числа, но и значения других типов — например, строки (именно по этой причине словарь также иногда называют «ассоциативным массивом»).

1.3 Алгоритм полного перебора

Алгоритмом полного перебора называют метод решения задачи, при котором по очереди рассматриваются все возможные варианты. В случае реализации алгоритма в рамках данной работы будут последовательно перебираться ключи словаря до тех пор, пока не будет найден нужный.

Трудоёмкость алгоритма зависит от того, присутствует ли искомым ключ в словаре, и, если присутствует — насколько он далеко от начала массива ключей. Пусть на старте алгоритм затрагивает k_0 операций, а при сравнении k_1 операций.

Пусть алгоритм нашёл элемент на первом сравнении (лучший случай), тогда будет затрачено $k_0 + k_1$ операций, на втором — $k_0 + 2 \cdot k_1$, на последнем (худший случай) — $k_0 + N \cdot k_1$. Если ключа нет в массиве ключей, то мы сможем понять это, только перебрав все ключи, таким образом трудоёмкость такого случая равно трудоёмкости случая с ключом на последней позиции. Трудоёмкость в среднем может быть рассчитана как математическое ожидание по формуле (1.1), где Ω — множество всех возможных случаев.

$$\sum_{i \in \Omega} p_i \cdot f_i = k_0 + k_1 \cdot \left(1 + \frac{N}{2} - \frac{1}{N+1}\right) \quad (1.1)$$

1.4 Формализация объектов и его признаков

В данной лабораторной работе словарь используется для описания «максимальной скорости автомобиля» со следующими параметрами: ключ — терм (словесное описание признака), значение — массив числовых значений признака.

Доступные термы:

- очень медленные;
- медленные;
- средние;
- быстрые;
- очень быстрые;
- не очень быстрые;
- не очень медленные.

Доступные числовые значения максимальной скорости: от 0 до 500 км/ч.

1.5 Анкетирование респондентов

На рисунке 1.1 приведены результаты анкетирования респондентов.

Студент	Терм	Значение признака (максимальная скорость авто)													
		10	20	30	40	50	100	150	200	250	300	350	400	450	500
Слава Шагалов	Очень медленные	*	*	*	*	*	*								
	Медленные							*							
	Средние								*						
	Быстрые									*	*				
	Очень быстрые										*	*	*	*	*
	Не очень быстрые						*	*	*						
Никита Котляров	Не очень медленные								*	*	*				
	Очень медленные	*	*	*	*	*									
	Медленные						*								
	Средние							*							
	Быстрые								*	*					
	Очень быстрые										*	*	*	*	*
Настя Кузнецова	Не очень быстрые					*	*	*							
	Не очень медленные							*	*	*	*				
	Очень медленные	*	*	*	*	*									
	Медленные						*								
	Средние							*	*						
	Быстрые									*	*				
Илья Бурлаков	Очень быстрые										*	*	*	*	*
	Не очень быстрые					*	*	*							
	Не очень медленные							*	*	*	*				
	Очень медленные	*													
	Медленные		*	*	*										
	Средние					*	*								
Аня Никулина	Быстрые							*	*						
	Очень быстрые									*	*	*	*	*	*
	Не очень быстрые			*	*	*	*								
	Не очень медленные				*	*	*	*							
	Очень медленные	*	*												
	Медленные			*											

Рисунок 1.1 – Результаты анкетирования

1.6 Функция принадлежности термам числовых значений признака

Построим графики функций принадлежности числовых значений переменной термам, описывающим группы значений лингвистической переменной.

Для этого для каждого значения максимальной скорости автомобиля для каждого термина из перечисленных найдём количество респондентов, согласно которым значение удовлетворяет сопоставляемому терму. Данное значение поделим на количество респондентов — это и будет значением функции μ для термина в точке.

На рисунке 1.2 показан результат подсчета значений функции принадлежности.

Терм	10	20	30	40	50	100	150	200	250	300	350	400	450	500
Очень медленные	5	4	3	3	3	1								
	1	0.8	0.6	0.6	0.6	0.2	0	0	0	0	0	0	0	0
Медленные		1	2	1		2	1							
	0	0.2	0.4	0.2	0	0.4	0.2	0	0	0	0	0	0	0
Средние				1	2	2	2	2						
	0	0	0	0.2	0.4	0.4	0.4	0.4	0	0	0	0	0	0
Быстрые							2	2	3	2				
	0	0	0	0	0	0	0.4	0.4	0.6	0.4	0	0	0	0
Очень быстрые								1	2	3	5	5	5	5
	0	0	0	0	0	0	0	0.2	0.4	0.6	1	1	1	1
Не очень быстрые		1	2	2	3	5	3	2						
	0	0.2	0.4	0.4	0.6	1	0.6	0.4	0	0	0	0	0	0
Не очень медленные				1	2	2	4	4	3	3				
	0	0	0	0.2	0.4	0.4	0.8	0.8	0.6	0.6	0	0	0	0

Рисунок 1.2 – Функция принадлежности термам числовых значений признака

На рисунке 1.3 представлены графики функций принадлежности термам числовых значений признака.

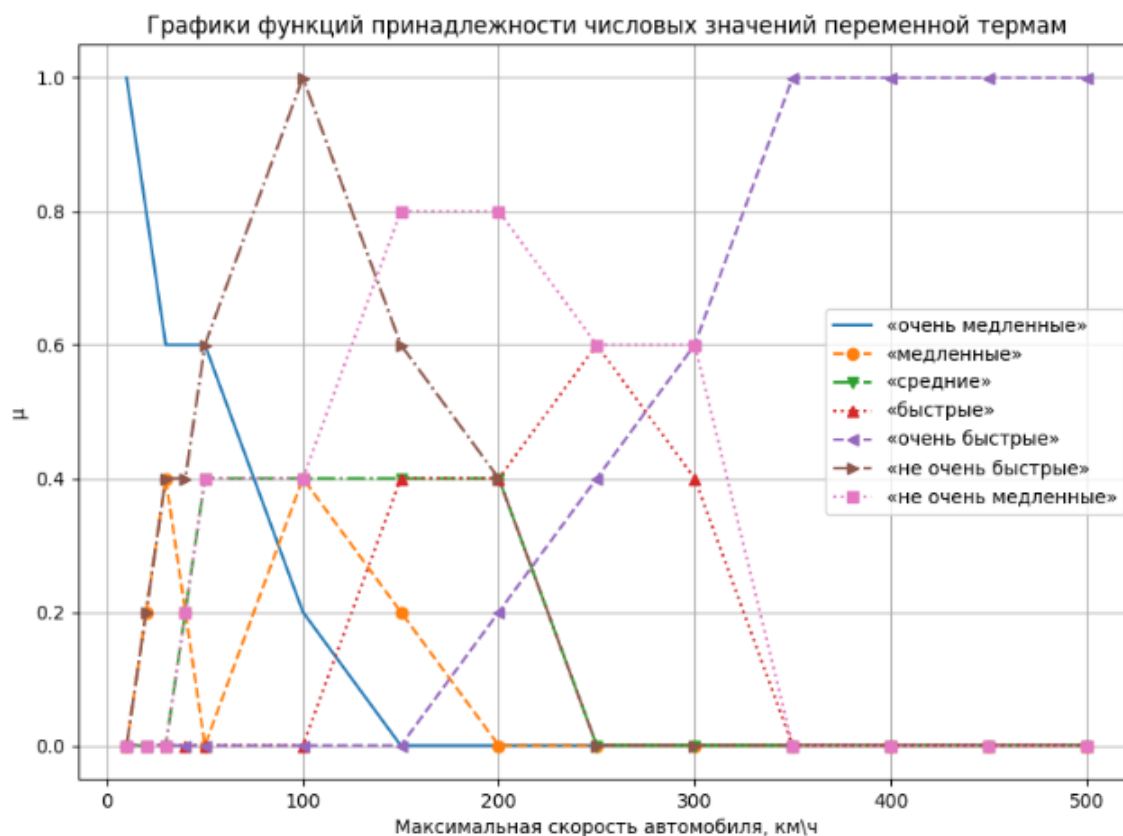


Рисунок 1.3 – Графики функций принадлежности термам числовых значений признака

2 Конструкторская часть

В этом разделе будут представлено описание используемых типов данных, а структура программы.

2.1 Описание используемых типов данных

При реализации алгоритмов будут использованы следующие типы данных:

- 1) словарь — встроенный тип *dict* [1] в *Python*;
- 2) массив ключей — встроенный тип *list* в *Python*;
- 3) длина массива/словаря — целое число *int*.

2.2 Исходные файлы программы

Программа будет состоять из двух модулей: *main.py* — файл, содержащий основную программу; *draw.py* — файл, содержащий программу отрисовки графика функций принадлежности числовых значений переменной термам.

3 Технологическая часть

В данном разделе будут рассмотрены средства реализации лабораторной работы, представлены тестовые данные, которые использовались для проверки корректности работы алгоритмов.

3.1 Выбор языка программирования и среды разработки

Для реализации алгоритмов был выбран язык *Python*, так как он предоставляет необходимую функциональность, позволяющую решить поставленные задачи на реализацию алгоритмов, а в качестве среды разработки — *PyCharm*. Для построения графиков использовалась библиотека *matplotlib* [4].

3.2 Тестирование

Классы эквивалентности:

- использование одиночных термов;
- использование составных термов с «очень»;
- использование составных термов с «не» и «очень».

В таблице 3.1 представлены тесты. Все тесты пройдены успешно.

Таблица 3.1 – Функциональные тесты

Запрос	Ожидаемы результат	Фактический результат
медленные автомобили	Niva 4x4	Niva 4x4
очень быстрые автомобили	Toyota Mark II Subaru Impreza Ford Mustang BMW X6 BMW X5 BMW M4	Toyota Mark II Subaru Impreza Ford Mustang BMW X6 BMW X5 BMW M4
не очень быстрые автомобили	Niva 4x4 Lada Granta	Niva 4x4 Lada Granta
не очень медленные автомобили	Lada Granta Mazda 6 Mazda 3 Nissan Skyline Nissan Silvia Toyota Land Cruiser	Lada Granta Mazda 6 Mazda 3 Nissan Skyline Nissan Silvia Toyota Land Cruiser

4 Исследовательская часть

В данном разделе будут представлены примеры работы программы и предоставлена информация о технических характеристиках устройства.

4.1 Технические характеристики устройства

Ниже представлены характеристики компьютера, на котором проводилось тестирование программы:

- операционная система Windows 10 Домашняя 21H2;
- оперативная память 16 Гб;
- процессор Intel(R) Core(TM) i7-10870H CPU @ 2.20 ГГц.

Во время тестирования ноутбук был подключен к сети электропитания. Из программного обеспечения были запущены только среда разработки *PyCharm* и браузер *Chrome*.

Процессор был загружен на 15%, оперативная память – на 50%.

4.2 Примеры работы программы

На рисунке 4.1–4.2 представлен результат работы программы.

```
Разбор запроса: ['не', 'очень', 'медленные', 'автомобили']  
['средние', 'быстрые']  
Выявленные термины:  
    средние  
    быстрые  
Результаты запроса:  
    Lada Granta-средние  
    Mazda 6-быстрые  
    Mazda 3-быстрые  
    Nissan Skyline-быстрые  
    Nissan Silvia-быстрые  
    Toyota Land Cruiser-быстрые
```

Рисунок 4.1 – Пример работы программы №1

```
Разбор запроса: ['очень', 'быстрые', 'автомобили']  
['очень быстрые']  
Выявленные термины:  
    очень быстрые  
Результаты запроса:  
    Toyota Mark II-очень быстрые  
    Subaru Impreza-очень быстрые  
    Ford Mustang-очень быстрые  
    Lamborghini Huracan-очень быстрые  
    BMW X6-очень быстрые  
    BMW X5-очень быстрые  
    BMW M4-очень быстрые
```

Рисунок 4.2 – Пример работы программы №2

Заключение

Цель, которая была поставлена в начале лабораторной работы, была достигнута: получен навык поиска по словарю при ограничении на значение признака, заданного при помощи лингвистической переменной.

Решены все поставленные задачи:

- формализован объект и его признак;
- проведено анкетирование респондентов;
- построена функция принадлежности термам числовых значений признака, описываемого лингвистической переменной, на основе статистической обработки мнений респондентов, выступающих в роли экспертов;
- описан алгоритм поиска в словаре объектов;
- описана структура данных словаря;
- реализован описанный алгоритм поиска в словаре.

Список использованных источников

1. dict Python [Электронный ресурс]. — URL: <https://docs.python.org/3/library/2to3.html?highlight=dictto3fixer-dict> (дата обращения: 20.12.2022).
2. С. Шапошникова. Словари — Лаборатория линуксоида [Электронный ресурс]. — URL: <https://younglinux.info/python/dictionary> (дата обращения: 20.12.2022).
3. time — Time access and conversions [Электронный ресурс]. — URL: https://docs.python.org/3/library/time.html#time.process_time (дата обращения: 02.10.2022)
4. Matplotlib documentation [Электронный ресурс]. — URL: <https://matplotlib.org/stable/index.html> (дата обращения: 10.10.2022)
5. Windows 10 Home [Электронный ресурс]. — URL: <https://www.microsoft.com/en-us/d/windows-10-home/d76qx4bznwk4?activetab=pivot:overviewtab> (дата обращения: 10.10.2022)
6. Intel(R) Core(TM) i7-10870H CPU @ 2.20GHz [Электронный ресурс]. — URL: <https://ark.intel.com/content/www/ru/ru/ark/products/208018/intel-core-i710870h-processor-16m-cache-up-to-5-00-ghz.html> (дата обращения: 10.10.2022)