



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 Гурова Н.А.

Группа ИУ7-54Б

Оценка (баллы) \_\_\_\_\_

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

# Оглавление

<b>Введение</b>	<b>3</b>
<b>1 Аналитическая часть</b>	<b>4</b>
1.1 Словарь . . . . .	4
1.2 Полный перебор . . . . .	4
<b>2 Конструкторская часть</b>	<b>6</b>
2.1 Требования к вводу . . . . .	6
2.2 Требования к программе . . . . .	6
2.3 Разработка алгоритмов . . . . .	6
<b>3 Технологическая часть</b>	<b>9</b>
3.1 Средства реализации . . . . .	9
3.2 Сведения о модулях программы . . . . .	9
3.3 Реализация алгоритмов . . . . .	9
3.4 Тестирование . . . . .	10
<b>4 Исследовательская часть</b>	<b>11</b>
4.1 Исследование . . . . .	11
<b>Заключение</b>	<b>14</b>
<b>Список использованных источников</b>	<b>15</b>

# Введение

Одной из ключевых операций при обработке данных является поиск. Время работы алгоритмов поиска зависит от размера набора данных. Для представления данных используются различные структуры, которые удобно использовать в конкретных задачах. Одной из таких структур данных является словарь [1]. Следовательно, становится актуальной задача быстрого поиска в словаре.

Цель данной работы: получить навык поиска по словарю при ограничении на значение признака, заданного при помощи лингвистической переменной.

Для достижения цели необходимо решить следующие задачи:

- формализовать объект и его признак;
- сформировать анкету для заполнения респондентом;
- провести анкетирование респондентов;
- описать структуру данных словаря;
- предложить и реализовать алгоритм поиска в словаре.

# 1 Аналитическая часть

В данном разделе были рассмотрены словарь как структура данных и алгоритм полного перебора по словарю.

## 1.1 Словарь

Обычные списки (массивы) представляют собой набор пронумерованных элементов. Номер элемента в списке однозначно идентифицирует сам элемент. Но идентифицировать данные по числовым номерам не всегда удобно. Например, маршруты поездов в России идентифицируются численно-буквенным кодом (число и одна цифра), также численно-буквенным кодом идентифицируются авиарейсы. Для хранения информации о рейсах поездов или самолетов в качестве идентификатора удобнее было бы использовать не число, а текстовую строку.

Структура данных, позволяющая идентифицировать ее элементы не по числовому индексу, а по произвольному, называется словарем или ассоциативным массивом [1]. Данная структура хранит пары вида «ключ-значение». При поиске возвращается значение, которое ассоциируется с данным ключом, или «Пара не найдена», если по данному ключу нет значений.

## 1.2 Полный перебор

Полный перебор [2] — метод решения, при котором поочередно перебираются все ключи словаря, пока не будет найден нужный.

Трудоёмкость алгоритма зависит от того, присутствует ли искомый ключ в словаре, и, если присутствует — насколько он далеко от начала массива ключей.

Так, если в начале алгоритм затрагивает  $b$  операций, а при сравнении  $k$  операций, то:

- элемент найден на первом сравнении за  $b + k$  операций (лучший случай);
- элемент найден на  $i$ -ом сравнении за  $b + i \cdot k$  операций;
- элемент найден на последнем сравнении за  $b + N \cdot k$  операций, где  $N$  — размер словаря (худший случай);

При этом средняя трудоемкость равна:

$$f = b + k \cdot \left( 1 + \frac{N}{2} - \frac{1}{N+1} \right) \quad (1.1)$$

## Вывод

Была изучена структура данных — словарь. Был рассмотрен подход к поиску в словаре — алгоритм полного перебора.

## 2 Конструкторская часть

В этом разделе были представлены требования к вводу и программе, а также схема алгоритма полного перебора.

### 2.1 Требования к вводу

На вход программе должна подаваться строка, представляющая из себя вопрос от пользователя.

### 2.2 Требования к программе

Выходные данные программы — ответ на вопрос пользователя, найденный в словаре, или сообщение о том, что найти ответ не получилось. Программа не должна аварийно завершаться при некорректном вводе пользователя.

### 2.3 Разработка алгоритмов

На рисунке 2.1 представлена схема алгоритма полного перебора.

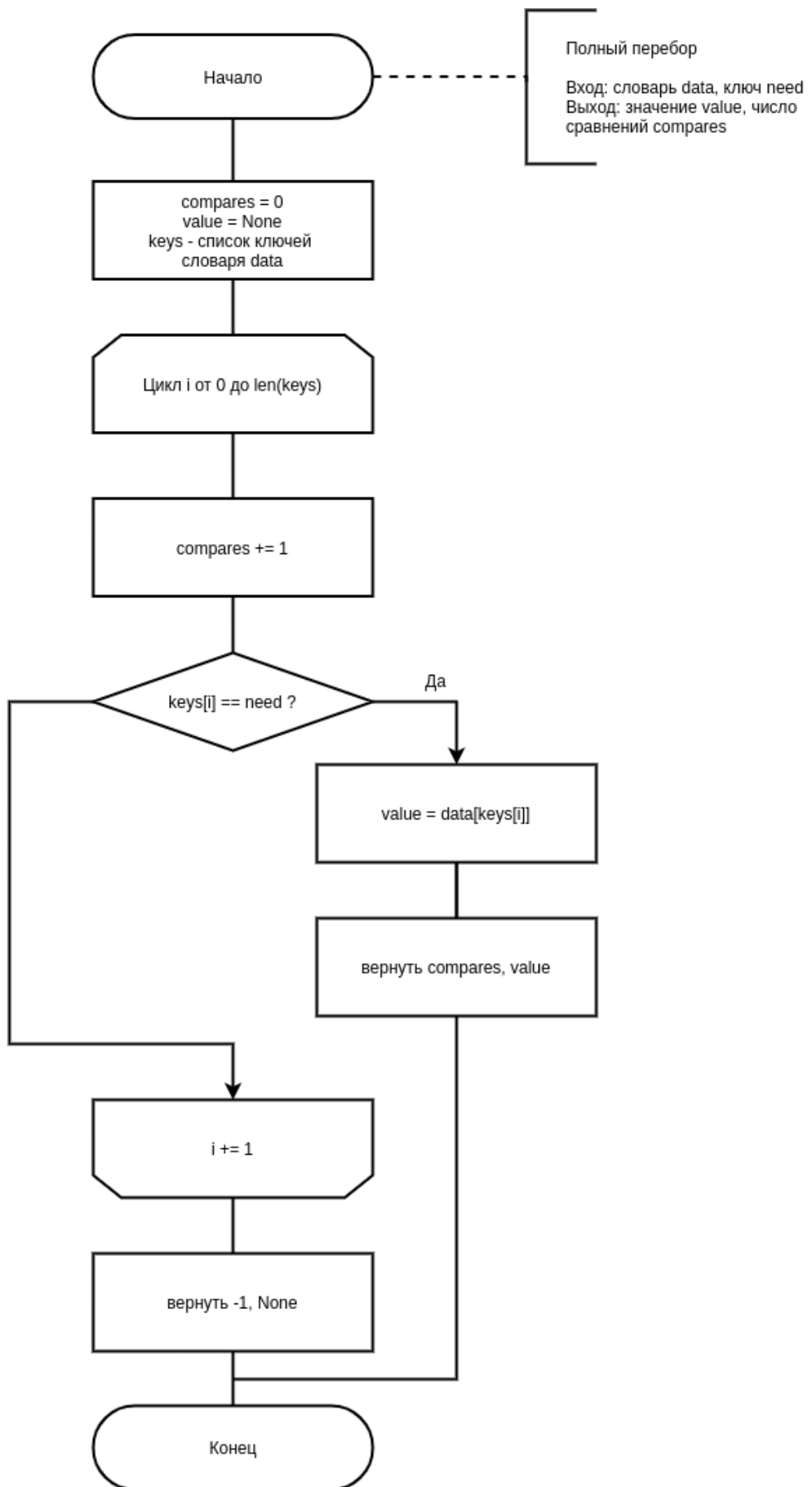


Рисунок 2.1 – Схема алгоритма полного перебора

## Вывод

В данном разделе была построена схема алгоритма полного перебора, были описаны требования ко вводу и программе.



## 3 Технологическая часть

В данном разделе были приведены средства реализации и листинги кода.

### 3.1 Средства реализации

В качестве языка программирования для реализации данной лабораторной работы был выбран язык программирования Python [3]. Данный язык имеет все инструменты для решения поставленной задачи.

### 3.2 Сведения о модулях программы

Программа состоит из трех модулей:

1. algorithms.py - хранит реализацию алгоритма поиска в словаре;
2. unit\_tests.py - хранит реализацию тестирующей системы и тесты;
3. tools.py - хранит реализацию вспомогательных функций.

### 3.3 Реализация алгоритмов

В листинге 3.1 представлен алгоритм полного перебора.

Листинг 3.1 – Реализация алгоритма поиска полным перебором

```
1      def BruteForceSearch(self , key):
2          k = 0
3          kk = list(self.data.keys())
4          for elem in self.data:
5              k += 1
6              if key == elem:
7                  // Writing to the log file
8                  self.f.write(f"{kk.index(key)},{key},{k}\n")
9                  return self.data[elem]
10         return -1
```

## 3.4 Тестирование

В данном разделе приведена таблица с тестами (таблица 3.1). Все тесты пройдены успешно.

Таблица 3.1 – Функциональные тесты

Входные данные	Пояснение	Результат
большое	Средний элемент	Ответ верный
маленькое	Первый элемент	Ответ верный
среднее	Последний элемент	Ответ верный
мяу	Несуществующий элемент	Ответ верный (-1)

## Вывод

В данном разделе была представлена реализация алгоритма полного перебора. Алгоритм был протестирован.

## 4 Исследовательская часть

В данном разделе приведена постановка эксперимента.

### 4.1 Исследование

В данной работе, в качестве респондентов, принимали участия следующие студенты.

- Малышев И.Н.
- Николаев С.С.
- Фролова В.А.
- Шабанова А.В.
- Щербина М.А.

Объектом данного исследования является продолжительность жизни человека. Исследуемый признак — количество лет, прожитых человеком.

Возможные значения признака (описывают количество):

- мало;
- средне;
- много;
- очень много;
- не очень много;
- не очень мало.

По результатам опроса была сформирована и занесена в таблицу обобщённая статистика. Упомянутая таблица представлена на рисунке 4.1.

Результаты опроса							
		Тезис					
		мало	средне	много	очень много	не очень много	не очень мало
Количество лет	< 25	4	0	0	0	0	0
	25	1	3	0	0	1	0
	50	0	1	1	0	2	3
	75	0	1	4	0	0	0
	100	0	0	2	3	0	0
	> 100	0	0	0	5	0	0

Рисунок 4.1 – Обобщённые результаты опроса.

В данной таблице приведены количество голосов, отданных респондентами в пользу истинности разных утверждений. В узлах таблицы расположено количество голосов в пользу высказывания, формируемого по принципу «количество лет это тезис» (Например, за истинность высказывания «50 лет это много» проголосовал один человек).

На рисунке 4.2 приведена таблица, содержащая нормализованные значения из таблицы 4.1.

Преобразованные результаты опроса							
		Тезис					
		мало	средне	много	очень много	не очень много	не очень мало
Количество лет	< 25	0,80	0,00	0,00	0,00	0,00	0,00
	25	0,20	0,60	0,00	0,00	0,33	0,00
	50	0,00	0,20	0,14	0,00	0,67	1,00
	75	0,00	0,20	0,57	0,00	0,00	0,00
	100	0,00	0,00	0,29	0,38	0,00	0,00
	> 100	0,00	0,00	0,00	0,63	0,00	0,00

Рисунок 4.2 – Нормализованные значения

Построенные функции принадлежности термам числовых значений признака, описываемого лингвистической переменной, на основе статистической обработки мнений респондентов, выступающих в роли экспертов, приведены на рисунке 4.3

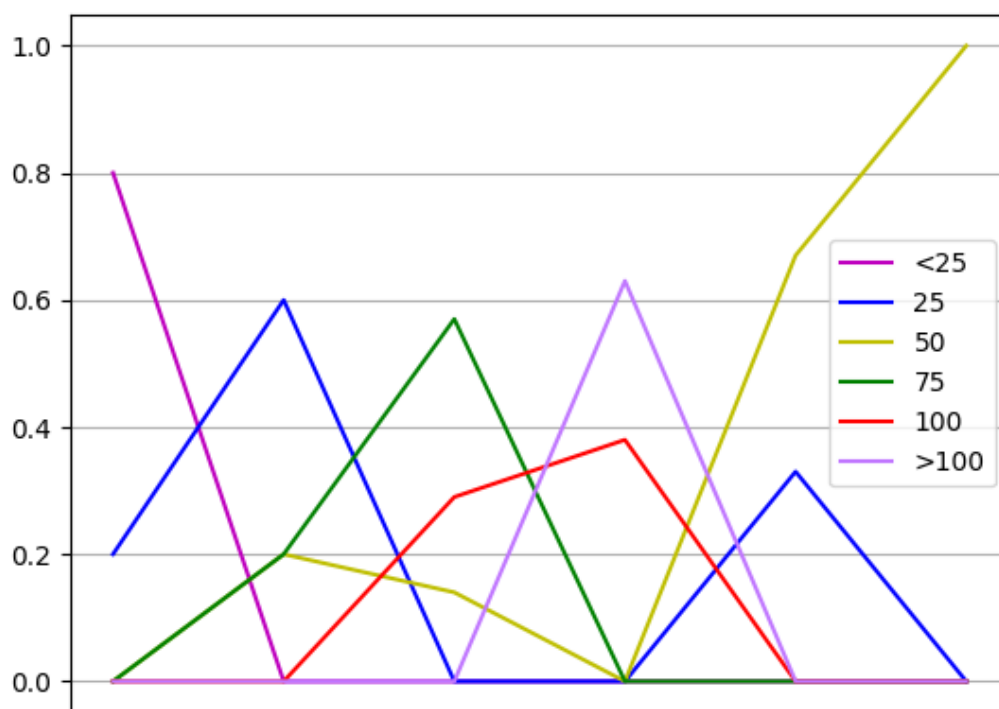


Рисунок 4.3 – Функции принадлежности

## Вывод

Таким образом, на основании экспертной оценки, были точно заданы функции принадлежности числового значения указанного признака некоторому термину. Благодаря данным функциям можно с некоторой точностью определить описанную принадлежность.

# Заключение

В ходе выполнения лабораторной работы были решены следующие задачи:

- формализован объект и его признак;
- сформирована анкета для заполнения респондентом;
- проведено анкетирование респондентов;
- описана структура данных словаря;
- реализован алгоритм поиска в словаре.

Поставленная в начале работы цель достигнута.

# СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- [1] Словари [Электронный ресурс]. Режим доступа: <https://informatics.msk.ru/mod/book/view.php?id=6694> (дата обращения: 10.12.2021).
- [2] Полный перебор [Электронный ресурс]. Режим доступа: <http://skud-perm.ru/posts/polnyj-perebor> (дата обращения: 10.12.2021).
- [3] Welcome to Python [Электронный ресурс]. Режим доступа: <https://www.python.org> (дата обращения: 04.09.2021).