МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ, СВЯЗИ И МАССОВЫХ КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ

Ордена Трудового Красного Знамени

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Московский технический университет связи и информатики»

Кафедра «Математическая кибернетика и информационные технологии»

Дисциплина «Структуры и алгоритмы обработки данных»

Курсовая работа

Выполнил:

Никитин П.М.,

группа БФИ2202

Проверил:

Симонов С. Е.

Москва

2024

Содержание

[ВВЕДЕНИЕ 3](#_Toc166984475)

[1. Постановка технического задания 5](#_Toc166984476)

[2. Техническая реализация 6](#_Toc166984477)

[ЗАКЛЮЧЕНИЕ 11](#_Toc166984478)

# ВВЕДЕНИЕ

В рамках выполнения данной курсовой работы необходимо создать программный продукт, выполняющий функции анализа данных на основе данных о продажах товаров в магазине.

Данные, на основе которых выполняется анализ, представлены в формате csv – Comma Separated Value, формата хранения табличных данных. Строки таблицы соответствуют строкам текста содержимого файла, а столбцы таблицы разделяются специальным символом разделителя. В базовой вариации символом разделителя является запятая, но также в качестве разделителя часто используется точка с запятой.

Важную роль в выполнении данного задания играют такие структуры данных как список, хеш-таблица и её реализация в рамках языка Python – словарь. Список представляет собой упорядоченный набор значений. Хеш-таблица — это структура данных, которая реализует интерфейс ассоциативного массива. Она позволяет хранить пары “ключ-значение” и выполнять операции добавления новой пары, удаления и поиска пары по ключу. В основе хеш-таблицы лежит хеш-функция, которая преобразует ключ в индекс массива, где хранится значение. Это обеспечивает быстрый доступ к данным по ключу, так как хеш-функция позволяет напрямую перейти к месту хранения данных без необходимости их последовательного поиска.

Хеш-функции, однако, сложно составить так, чтобы каждому возможному ключу соответствовал свой уникальный индекс. Ситуация, при которой различным ключам соответствует одинаковое значение хеш-функции, называется коллизией. В рамках данной курсовой работой в качестве метода разрешения коллизией используется простое рехеширование, также называемое линейным рехешированием или зондированием, заключающееся в линейном поиске первой свободной ячейки таблицы после подвергнувшейся коллизии.

Для визуализации проанализированных данных используется библиотека matplotlib, предоставляющая широкие возможности по созданию графиков. Для упорядочивания некоторых данных применяется сортировка вставками.

Цель работы: Разработка программного продукта, выполняющего функции анализа данных на основе информации о продажах товаров в магазине, с применением структур данных и алгоритмов, изученных в процессе обучения по дисциплине «Структуры и алгоритмы обработки данных».

Таким образом, в ходе выполнения курсовой работы необходимо решить следующие задачи:

1. Формулировка технического задания на основе требований к программе для курсовой работы;

2. Составление программного кода в соответствии с поставленным техническим заданием;

3. Написание отчёта по курсовой работе.

# 1. Постановка технического задания

В рамках выполнения курсовой работы следует выполнить следующие требования технического задания. Необходимо написать программу, которая будет считывать данные из CSV файла, содержащего информацию о продажах товаров в магазине. Данные в файле содержатся в следующем формате: | Номер заказа | Дата заказа | Название товара | Категория товара | Количество продаж | Цена за единицу | Общая стоимость.

Следовательно, необходимо:

1. Рассчитать общую выручку магазина.
2. Найти товар, который был продан наибольшее количество раз.
3. Найти товар, который принес наибольшую выручку.
4. Составить отчет, содержащий информацию об общей выручке магазина, количестве проданных единиц каждого товар и доле каждого товара в общей выручке.

Для решения задач необходимо использовать структуры данных, такие как массивы и хеш-таблицы, а также различные алгоритмы обработки данных, например, сортировку и поиск. Также необходимо учитывать возможные ошибки ввода-вывода и обрабатывать их в соответствии с требованиями.

Учитывая вышеуказанные требования, в разрабатываемом программном коде будет использована хеш-таблица в качестве структуры данных для хранения и анализа данных из CSV файла, а также метод простого рехеширования (линейного зондирования) будет использован для разрешения коллизий. Для корректной визуализации информации будет использован алгоритм быстрой сортировки.

# 2. Техническая реализация

Важнейшей структурой данных, использующейся для выполнения задания, является хеш-таблица. Её реализация в коде курсовой работы – это Python класс HashTable. Рассмотрим его подробнее.

Класс HashTable – это реализация хеш-таблицы с открытой адресацией. При инициализации экземпляра класса, он принимает параметр size, определяющий размер хеш-таблицы. В качестве стандартной хеш-функции используется сложения числовых кодов символов ключа с последующим взятием остатка от деления на длину хеш-таблицы. Метод put реализует внесение значений в хеш-таблицу. Ключу ставится в соответствие значение хеш-функции, указывающее на некоторую ячейку таблицы. Если она пуста – то ключ и значение вносятся в неё. В противном случае начинается линейный перебор таблицы в поисках свободной ячейки. Метод get реализует обратный функционал – взятие значений из таблицы по ключу. Ему также ставится в соответствие значение хеш-функции, после чего линейно перебирает ячейки хеш-таблицы вплоть до обнаружения соответствия ключа. В случае, если изначально указанная ячейка хеш-таблицы пуста или в таблице не обнаруживается соответствующего ключа, вместо искомого значения метод возвращает None.

Программа начинается со считывания данных из csv-файла, содержащего сведения о заказах в магазине музыкальных инструментов. Для считывания информации используются стандартные методы языка Python – с помощью функции open() происходит открытие файла, с помощью readlines() осуществляется его построчное чтение. Строковый метод split(“;”) позволяет разделить строку на несколько подстрок по разделителю, в качестве которого выступает точка с запятой. Обработка строки файла происходит только в том случае, если она содержит 7 колонок в соответствие с ожидаемым форматом данных, в противном случае выводится сообщение о некорректности данных. В случае, если дата заказа не представлена в формате день.месяц.год или в ней содержатся нечисловые символы, то вместо неё датой заказа принимается день запуска программы.

Затем следует обработка файла sells.csv – этот блок кода отвечает за чтение данных из файла sells.csv. Каждая строка файла разделяется по символу «;», после чего выполняются следующие действия: проверяется, содержит ли строка ровно 7 элементов после разделения и, если нет, выводится сообщение о некорректных данных. Далее проверяется формат даты и, если дата не соответствует формату dd.mm.yyyy или содержит нечисловые символы, используется текущая дата. После этого данные из строки преобразуются в список с элементами: идентификатор заказа, дата, название товара, категория, количество, цена и стоимость. Этот список добавляется в список orders, который содержит информацию о каждом заказе, далее с использованием полученных данных создаётся хеш-таблица для хранения информации о товарах: в этом блоке кода для каждого заказа из списка orders проверяется, есть ли уже товар с таким названием в хеш-таблице hash\_goods. Если товар найден, то его данные обновляются: увеличивается количество и стоимость, а если товар не найден – он добавляется в хеш-таблицу с соответствующими значениями.

Следующими в коде исполняются функции анализа данных: total\_income(), вычисляющая общий доход от всех заказов, max\_selled(), находящая товар с максимальным количеством продаж, и max\_income(), находящая товар с максимальной выручкой.

Далее в программном коде следует генерация отчёта с помощью matplotlib – данный блок программы реализован функцией report(), которая создаёт текстовый отчет и график распределения выручки от различных товаров с помощью библиотеки matplotlib, использующейся для визуализации выводимой информации. Так, сначала создаётся полотно для графика, на него наносится круговая диаграмма с данными о выручке различных товаров, а также подписываются её оси, заголовок и легенда.

После алгоритма генерации отчёта в коде реализован алгоритм быстрой сортировки quick\_sort. Это рекурсивная функция быстрой сортировки, которая использует случайный элемент в качестве опорного и разделяет массив на три части: элементы меньше опорного, равные опорному и больше опорного. Затем она рекурсивно сортирует подмассивы. Далее следует функция income\_daily, которая строит график ежедневной прибыли магазина с использованием алгоритма быстрой сортировки quick\_sort и библиотеки matplotlib для визуализации выводных данных.

После своего исполнения программа выводит отчёт по общей выручке магазина и доле каждого товара в ней, круговую диаграмму распределения выручки от различных товаров и график ежедневной прибыли магазина (рисунки 1-3).

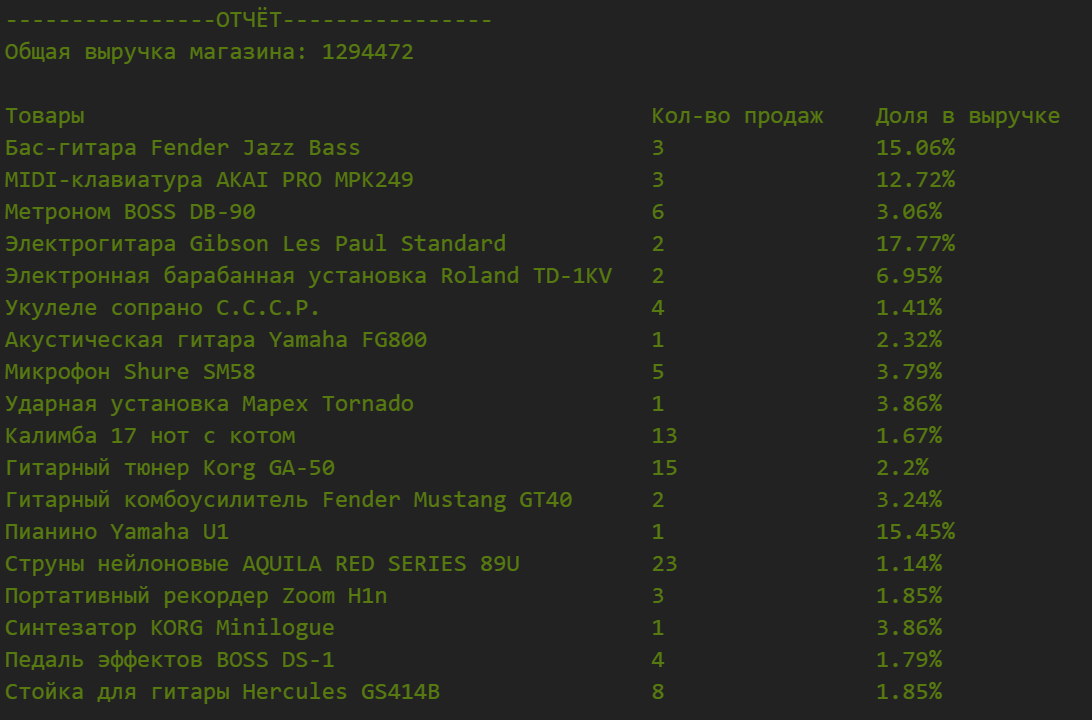


Рисунок 1 – Итоговый вывод с отчётом по общей выручке магазина и доле каждого товара в ней

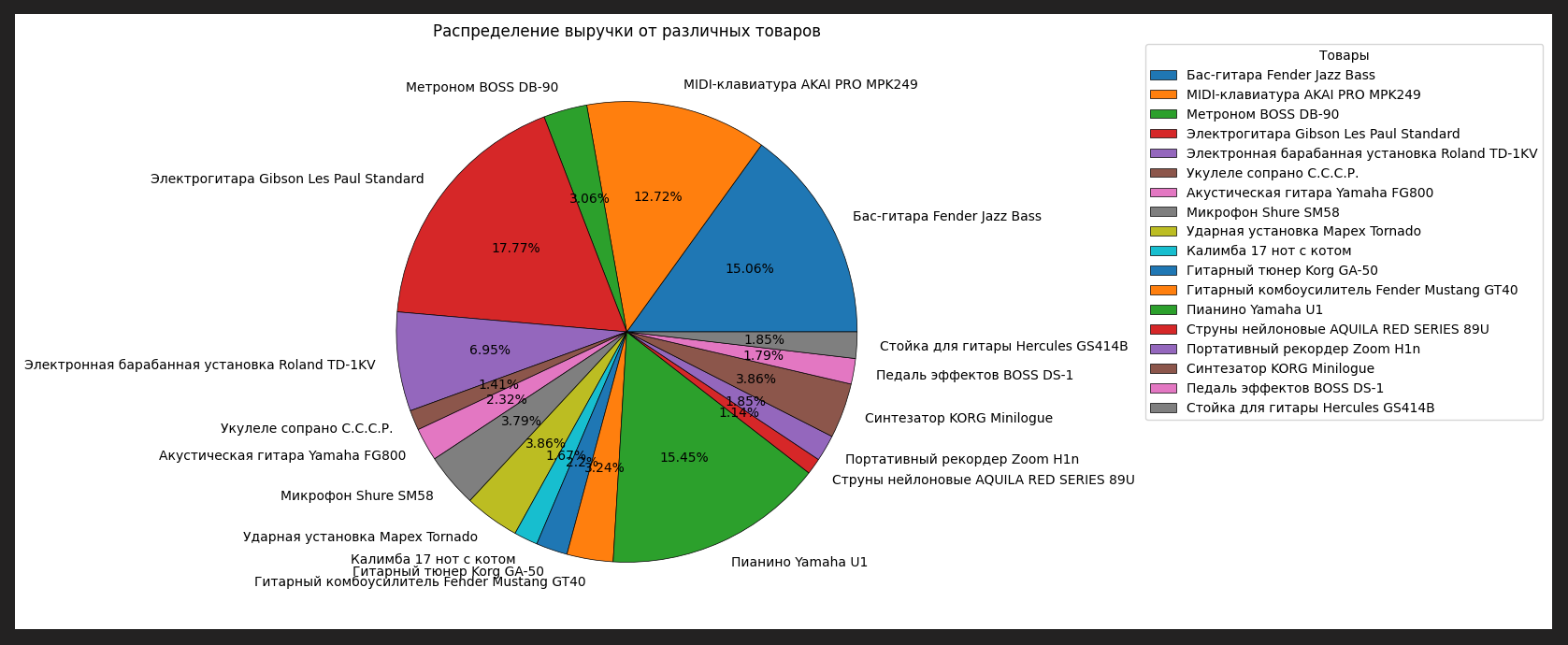


Рисунок 2 – Круговая диаграмма распределения выручки от различных товаров

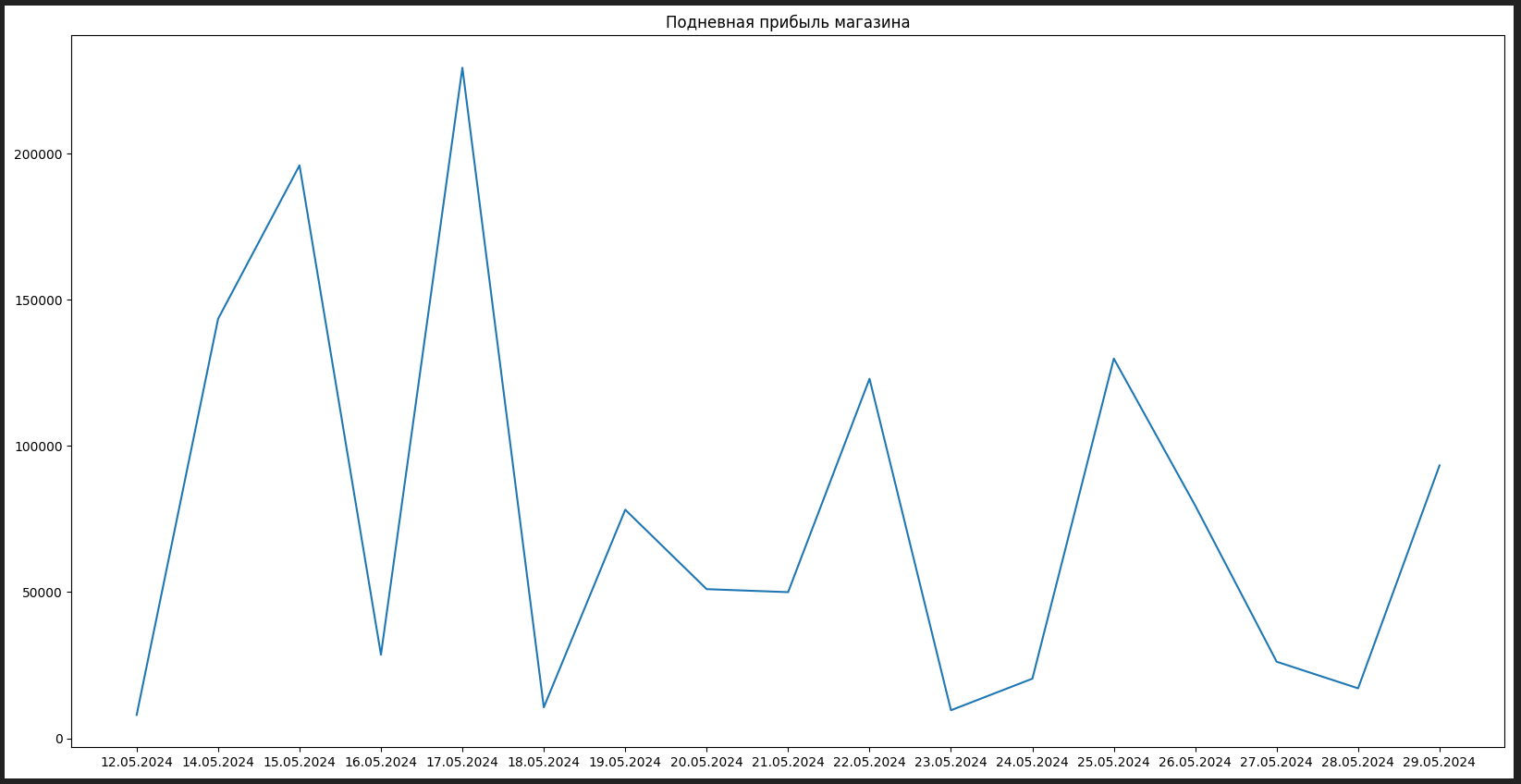


Рисунок 3 – График ежедневной прибыли магазина

# ЗАКЛЮЧЕНИЕ

В заключение, в процессе выполнения курсовой работы было составлено техническое задание на основе требований к программе для курсовой работы, создан программный код в соответствии с поставленным техническим заданием и написан отчёт по курсовой работе. В ходе создания программного кода была составлена программа, считывающая данные из CSV файла, содержащего информацию о продажах товаров в магазине, рассчитывающая общую выручку магазина и долю каждого товара в ней, и выводящая эти данные в виде отчёта. В результате выполнения программы также была создана круговая диаграмма распределения выручки от различных товаров и график ежедневной прибыли магазина.