# **Федеральное государственное бюджетное профессиональное образовательное учреждение**

**Торжокский политехнический колледж**

# **Федерального агентства по государственным резервам**

**ОТЧЁТ**

# о зимней работе

**по профессиональному модулю МДК.01.01**

«**РАЗРАБОТКА ПРОГРАМНЫХ МОДУЛЕЙ**»

наименование профессионального модуля

## Специальность ***09.02.07 «Информационные системы и программирование»***

Период с «28» декабря 2024г. по «08» февраля 2025г.

## Выполнили студенты

Тимофеев А.Д. Стариков С.К.

(ФИО, подпись)

|  |  |
| --- | --- |
| Преподаватель  Бойков А.М.  (ФИО, подпись) | оценка |

## 2024

**Содержание**

1. Практическая значимость разрабатываемого ПС ......................... 3
2. Функциональные требования к ПС .................................................. 4
3. Модульная структура проектируемого ПС ..................................... 5
4. Описание алгоритмов ............................................................................ 6
5. Разработка проекта Visual Studio C# ................................................. 8

ВВЕДЕНИЕ отсутствует

**Интервалы страдают, полуторный интервал, отступ первой строки абзаца – 1.25 см.**

**1. Практическая значимость разрабатываемого ПС**

Программное средство (ПС) предназначено для прогнозирования спроса на товары на основе анализа исторических данных о продажах, хранящихся в CSV-файлах.

**Пользователи ПС:**

* Менеджеры по продажам и закупкам
* Владельцы малого и среднего бизнеса
* Аналитики товарных запасов

**Практическая значимость:**

1. Оптимизация товарных запасов (исключение избытка/дефицита)
2. Автоматизация процесса прогнозирования
3. Улучшение точности прогнозов за счёт фильтрации аномальных значений

**Если бы данного ПС не было:**

* Анализ проводился бы вручную с высокой вероятностью ошибок
* Требовалось бы больше времени на обработку данных
* Снизилась бы точность прогнозов из-за невозможности корректно обрабатывать выбросы

Привести примеры конкретных сценариев, которые сейчас выполняют пользователи для решения поставленной задачи.

Описать критерии эффективности, по которым планируется оценивать эффект от внедрения данного программного средства

**2. Функциональные требования к ПС**

Требования должно однозначно и подробно раскрывать смысл соответствующей функции. В требованиях должна содержаться информация:

- как планируется использовать данную функцию в программе;

-какие существуют ограничения;

-какие возможны исключительные или ошибочные ситуации;

-какие требуются исходные данные;

-какие ожидаются результаты от выполнения функции

**Основные функции:**

1. Загрузка данных из CSV-файла формата:  
   Название товара; Дата продажи; Количество

Необходимо расписать требования к структуре такого файла, месту его хранения и способу указания этого файла пользователем.

Описать возможные ошибочные сценарии и реакцию программы на них.

Привести примеры таких файлов

1. Фильтрация аномальных значений (выбросов)

Описать, в чем смысл и задача фильтрации выбросов

По какому принципу должна осуществляться фильтрация

Привести конкретные примеры

1. Расчет средних дневных продаж по каждому товару

Аналогично фильтрации выбросов, описать и эту функцию

1. Прогнозирование спроса на заданное количество дней

Тоже самое, функция не описана

1. Сохранение результатов в текстовый файл

Функция не описана (каких результатов? Куда? В какой файл, какого формата? Что будет, если не сохраниться? И т.д. и т.п.

Привести примеры выходных файлов

**Пример использования:**

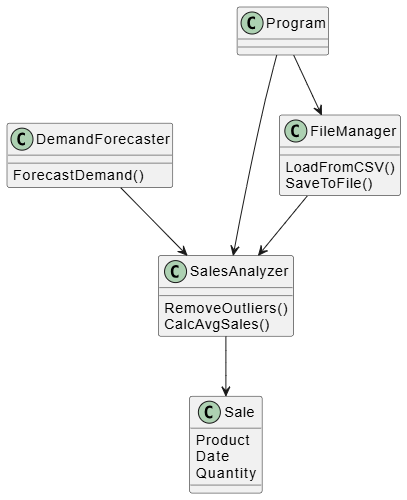
1. Пользователь запускает программу
2. Программа загружает данные из файла sales.csv
3. Автоматически фильтрует аномальные значения
4. Пользователь вводит количество дней для прогноза (например, 7)
5. Программа выводит прогноз и сохраняет его в файл forecast.txt

**3. Модульная структура проектируемого ПС**

**Диаграмма классов: (диаграмма подписывается как рисунок)**

**В диаграмме не приведены типы возвращаемых значений методов и их аргументы**

**В классах нет ни одного поля**

****

Каждый отдельный класс требует описания:

- для чего нужен этот класс (какую роль играет);

- описать каждый отдельный метод; (сигнатура, решаемая задача, какие входные/выходные данные)

- описать каждое отдельное поле класса

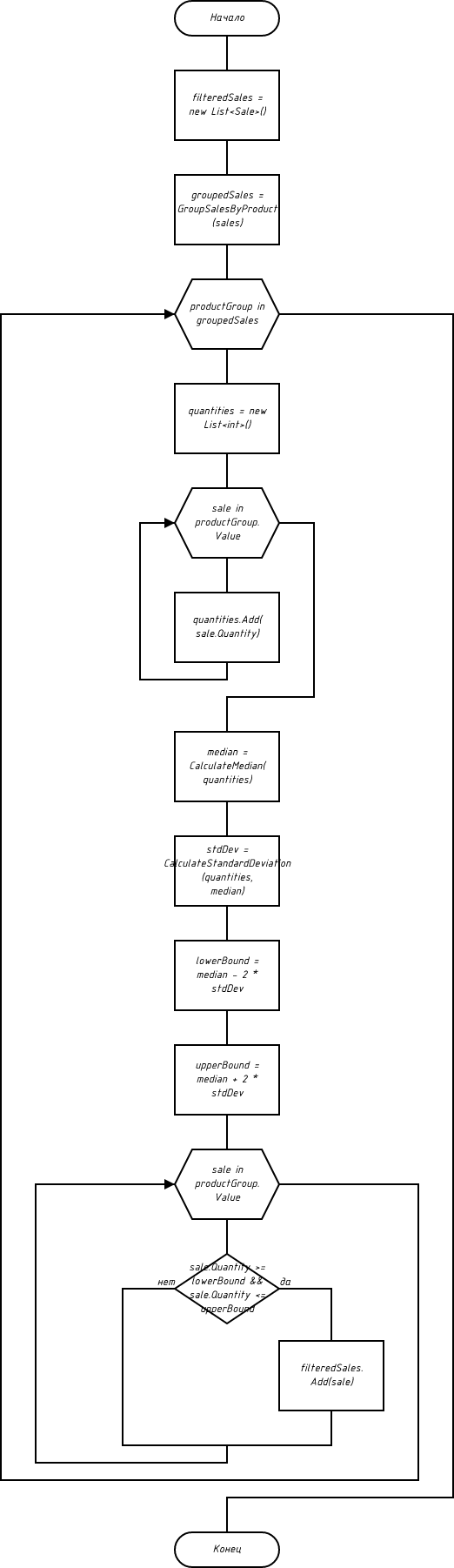
**4. Описание алгоритмов**

**Алгоритм 1: Фильтрация выбросов**

1. Группируем продажи по товарам
2. Для каждого товара:
   * Сортируем значения по возрастанию (значения чего?)
   * Вычисляем медиану (медиану из чего?)
   * Вычисляем стандартное отклонение (каким образом вычисляется?)
   * Определяем границы:  
     нижняя = медиана - 2\*σ  
     верхняя = медиана + 2\*σ
   * Удаляем значения вне границ

Что такое σ? Из описания это непонятно

**Блок-схема алгоритма фильтрации:**

****

**Блок-схему не видно. Переделать**

**Алгоритм 2: Прогнозирование спроса**

1. Для каждого товара:
   * Суммируем все продажи
   * Делим на количество дней с продажами → среднее за день
   * Умножаем на количество дней прогноза
2. Возвращаем результат (какой?)

**Блок-схемы алгоритма нет**

**Разработанное программное средство должно в точности соответствовать структуре, представленной в разделе 3, а также выполнять все функции, обозначенные в разделе 2.  
5. Разработка проекта Visual Studio C#**

**Полный код программы:**

using System;

using System.Collections.Generic;

using System.IO;

namespace DemandForecast

{

class Sale

{

public string Product { get; set; }

public DateTime Date { get; set; }

public int Quantity { get; set; }

public Sale(string product, DateTime date, int quantity)

{

Product = product;

Date = date;

Quantity = quantity;

}

}

class SalesAnalyzer

{

public List<Sale> RemoveOutliers(List<Sale> sales)

{

// Реализация алгоритма фильтрации

}

public Dictionary<string, double> CalculateAverageSales(List<Sale> sales)

{

// Реализация расчета средних значений

}

}

class DemandForecaster

{

public Dictionary<string, double> ForecastDemand(

Dictionary<string, double> avgSales, int days)

{

// Реализация прогнозирования

}

}

class FileManager

{

public List<Sale> LoadFromCSV(string path)

{

// Реализация загрузки из CSV

}

public void SaveToFile(Dictionary<string, double> forecast, string path)

{

// Реализация сохранения в файл

}

}

class Program

{

static void Main(string[] args)

{

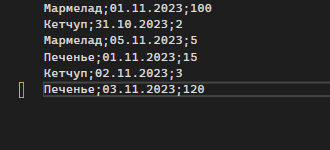
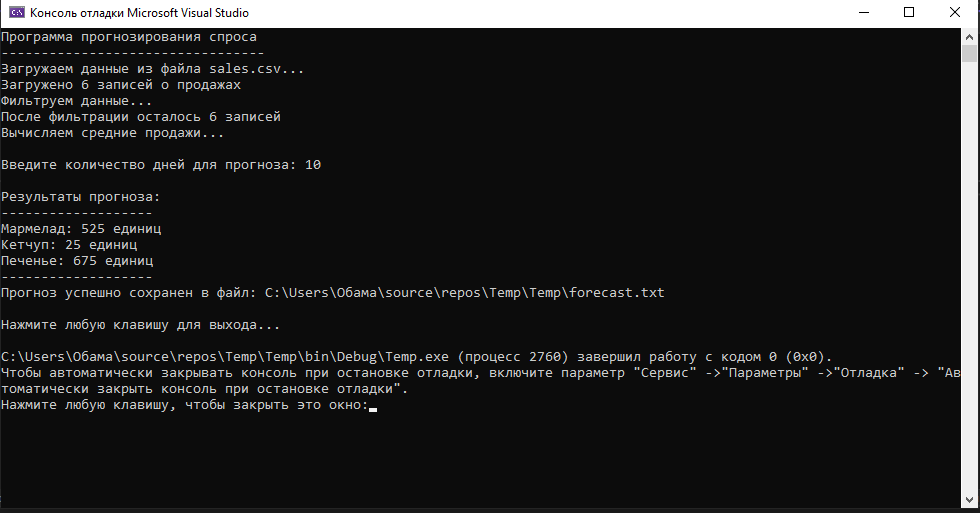
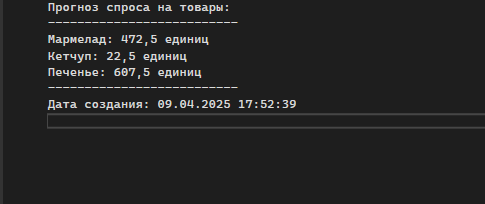
// Основная логика программы

}

}

}

**Скриншоты работы программы:**

1. Пример CSV-файла с данными:  
    
2. Пример вывода прогноза:  
    
3. Пример сохраненного файла с прогнозом:  
    

**Вывод:**  
 Разработанное программное средство успешно решает задачу прогнозирования спроса на товары. Модульная структура позволяет легко расширять функционал. Алгоритмы фильтрации и прогнозирования обеспечивают точность результатов. Программа может быть интегрирована в более крупные системы управления товарными запасами.