Министерство транспорта Российской Федерации

Федеральное государственное автономное образовательное

учреждение высшего образования

«Российский университет транспорта»

(ФГАОУ ВО РУТ(МИИТ), РУТ (МИИТ)

Институт транспортной техники и систем управления

Кафедра «Управление и защита информации»

Лабораторная работа № 12

по дисциплине: «Программирование и основы алгоритмизации»

на тему: «Файлы»

Выполнил: ст. гр. ТУУ-111

Грачева Н.С.

Вариант №5

Проверил: к.т.н., доц. Сафронов А.И.

Москва – 2024 г.

## Цель работы

В среде *Microsoft* *Visual Studio* на языке *Visual C#* в консольном режиме составить программное обеспечение для решения типовых задач программирования по тематике «Файлы» («*Files*»), заданных по варианту.

## Формулировка задачи

**Вариант №5.** В интегрированной среде разработки (*IDE*) *Microsoft Visual Studio* составить учебную консольную программу. Убедиться, что *IDE* создала решение (*solution*) с классической структурой каталогов и файлов программного продукта, разрабатываемого на языке *Visual C#* (*csproj*). Подготовить исходный текстовый файл с именем «***storage01.txt***» по пути «*../bin/Debug/*» решения (*solution*). В каждой строке текстового файла последовательно записаны: код, название, количество, стоимость товаров некоторой фирмы. Все значения для одного товара разделены между собой символом доллар «$». Каждый товар в файле занимает строку (сколько товаров – столько и строк). В результате работы учебной программы:

– в текстовый файл с именем «***storage02.txt***» записываются товары, количество которых равно нулю,

– в другой текстовый файл с именем «***storage03.txt***» записываются оставшиеся товары, отсортированные по количеству (от большего к меньшему),

– по итогам записи файлов в консоль построчно выводятся наименования товаров, которых нет в наличии.

## Блок-схема

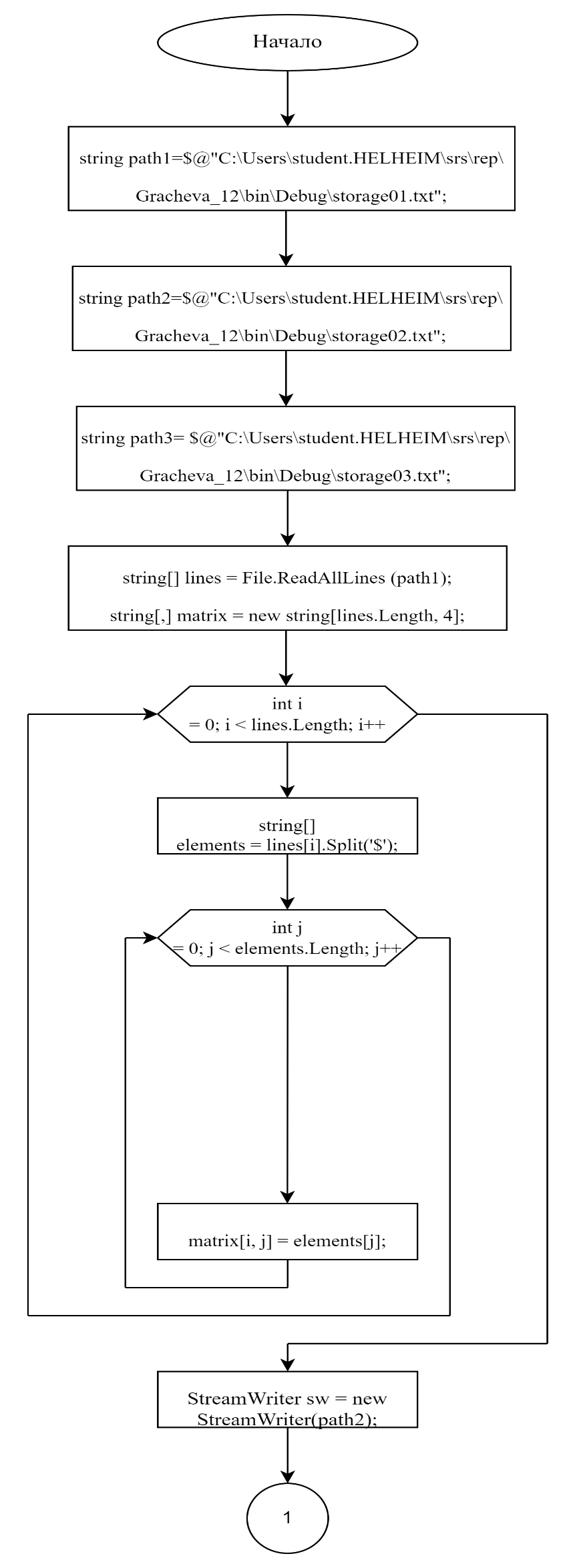


Рисунок 1­ — первая часть блок-схемы

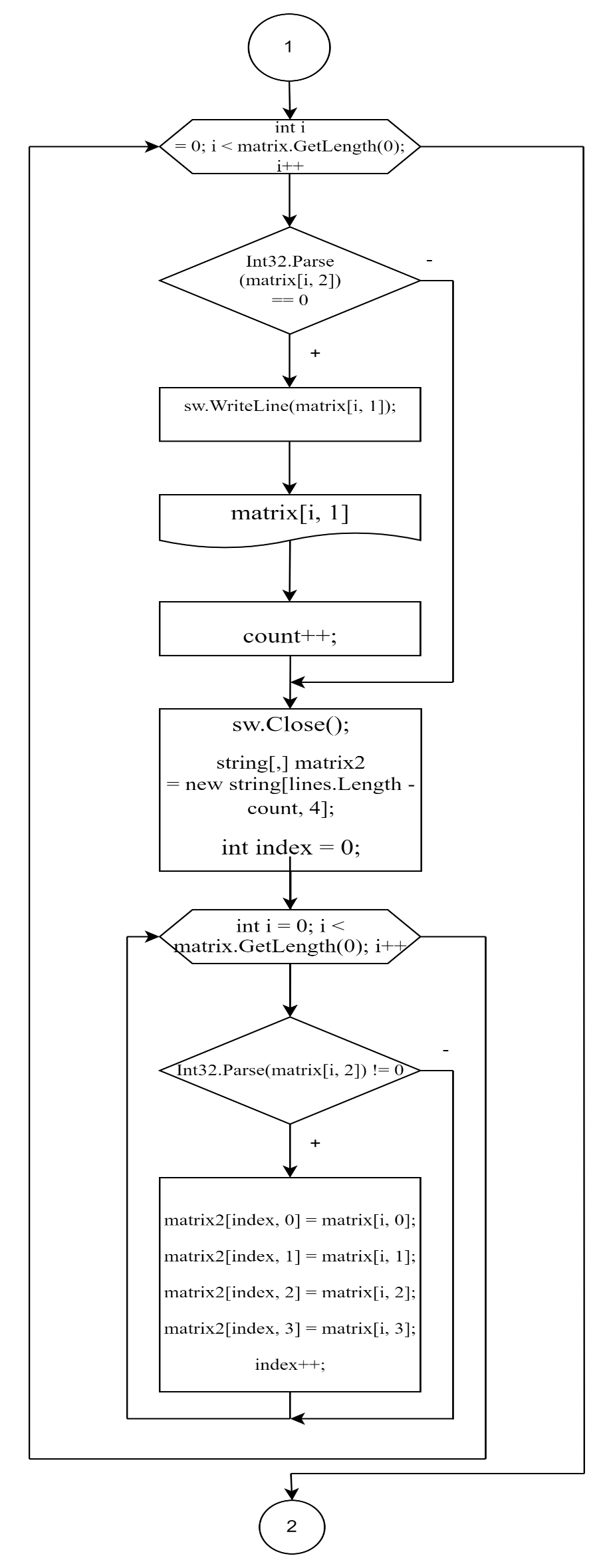


Рисунок 2 – вторая часть блок-схемы

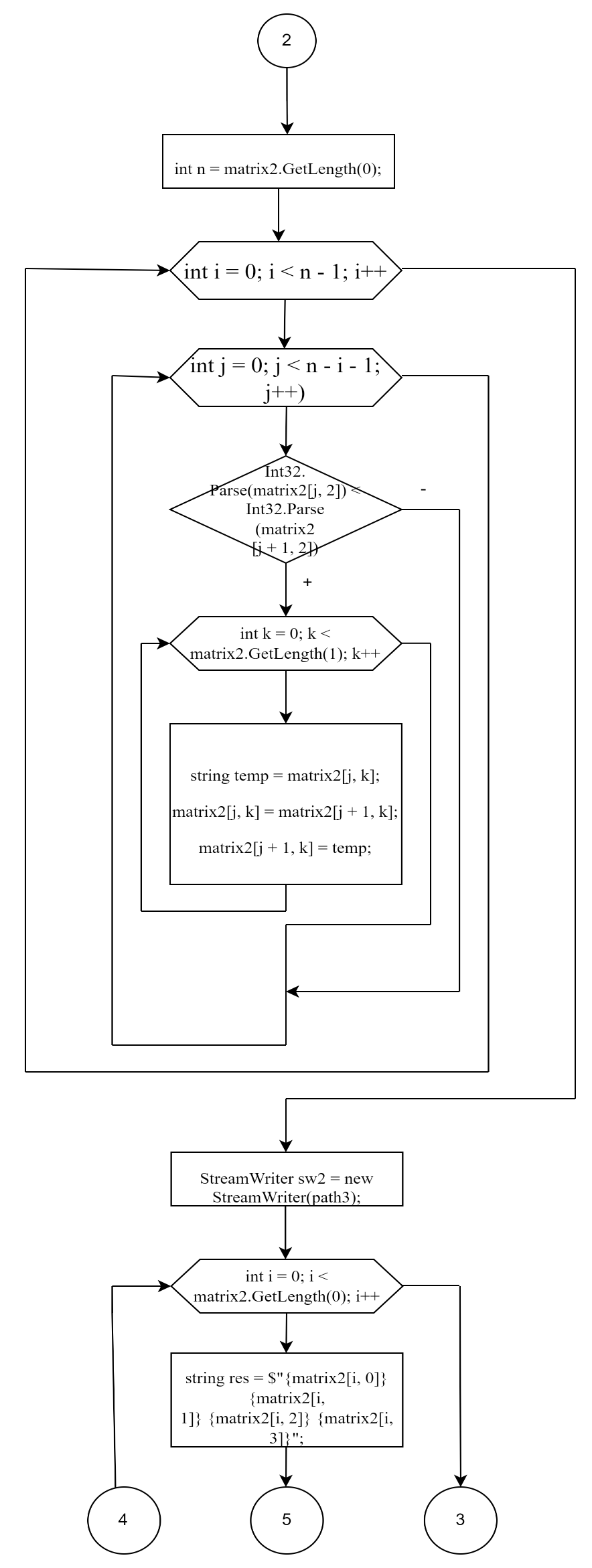


Рисунок 3 – третья часть блок-схемы

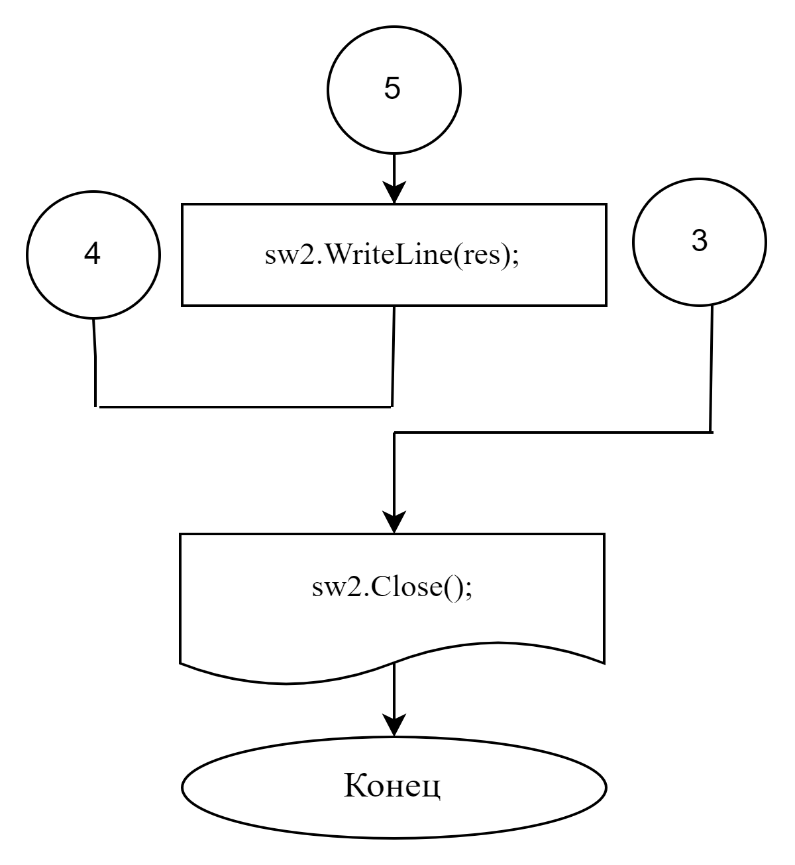


Рисунок 4 – четвёртая часть блок-схемы

## Подбор тестовых примеров

*Файл с товарами содержит следующие данные:*

001$творог$5$120

002$картофель$0$50

003$морковь$10$80

004$огурцы$2$40

005$молоко$0$70

006$помидоры$4$30

007$колбаса$3$200

008$сыр$1$90

009$хлеб$0$25

010$курица$6$500

## Листинг (код программы)

using System;  
using System.IO;  
  
  
namespace lr\_12  
{  
    internal class Program  
    {  
        static void Main(string[] args)  
        {  
            int count = 0;  
            string path1 = $@"C:\Users\student.HELHEIM\srs\rep\Gracheva\_12\bin\Debug\storage01.txt";  
            string path2 = $@"C:\Users\student.HELHEIM\srs\rep\Gracheva\_12\bin\Debug\storage02.txt";  
            string path3 = $@"C:\Users\student.HELHEIM\srs\rep\Gracheva\_12\bin\Debug\storage03.txt";  
  
  
            string[] lines = File.ReadAllLines(path1);  
            string[,] matrix = new string[lines.Length, 4];  
  
  
            for (int i = 0; i < lines.Length; i++)  
            {  
                string[] elements = lines[i].Split('$');  
                for (int j = 0; j < elements.Length; j++)  
                {  
                    matrix[i, j] = elements[j];  
                }  
            }  
  
            StreamWriter sr = new StreamWriter(path2);  
            for (int i = 0; i < matrix.GetLength(0); i++)  
            {  
                if (Int32.Parse(matrix[i, 2]) == 0)  
                {  
                    sr.WriteLine(matrix[i, 1]);  
                    Console.WriteLine(matrix[i, 1]);  
  
                    count++;  
                }  
            }  
            sr.Close();  
  
            string[,] matrix2 = new string[lines.Length - count, 4];  
            int index = 0;  
            for (int i = 0; i < matrix.GetLength(0); i++)  
            {  
                if (Int32.Parse(matrix[i, 2]) != 0)  
                {  
                    matrix2[index, 0] = matrix[i, 0];  
                    matrix2[index, 1] = matrix[i, 1];  
                    matrix2[index, 2] = matrix[i, 2];  
                    matrix2[index, 3] = matrix[i, 3];  
                    index++;  
                }  
            }  
  
            int n = matrix2.GetLength(0);  
            for (int i = 0; i < n - 1; i++)  
            {  
                for (int j = 0; j < n - i - 1; j++)  
                {  
                    if (Int32.Parse(matrix2[j, 2]) < Int32.Parse(matrix2[j + 1, 2]))  
                    {  
                        for (int k = 0; k < matrix2.GetLength(1); k++)  
                        {  
                            string temp = matrix2[j, k];  
                            matrix2[j, k] = matrix2[j + 1, k];  
                            matrix2[j + 1, k] = temp;  
                        }  
                    }  
                }  
            }  
  
            StreamWriter sr2 = new StreamWriter(path3);  
            for (int i = 0; i < matrix2.GetLength(0); i++)  
            {  
                string res = $"{matrix2[i, 0]} {matrix2[i, 1]} {matrix2[i, 2]} {matrix2[i, 3]}";  
                sr2.WriteLine(res);  
            }  
            sr2.Close();  
            Console.ReadKey(true);  
        }  
    }  
}

## Расчёт тестовых примеров на ПК

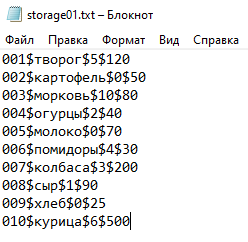


Рисунок 5- Список товаров

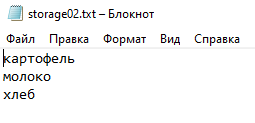


Рисунок 6 - Файл с товарами, количество которых равно нулю

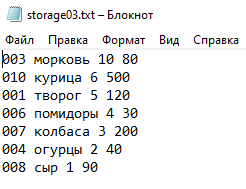


Рисунок 7 - Файл с товарами, отсортированных по количеству (от большего к меньшему)

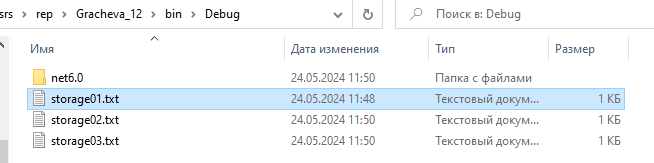


Рисунок 8 - исходный текстовый файл с именем «***storage01.txt***» по пути «*../bin/Debug/*»

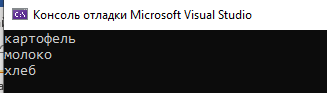


Рисунок 9 - наименования товаров, которых нет в наличии

## Вывод

В среде *Microsoft* *Visual Studio* на языке *Visual C#* в консольном режиме составила программное обеспечение для решения типовой задачи программирования по тематике «Файлы» («*Files*»), заданной по варианту. Научилась работать с данными, которые находятся в файлах, и использовать их для решения задачи.