МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ

РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное

учреждение высшего образования

«Южно-Уральский государственный университет

(национальный исследовательский университет)»

Высшая школа электроники и компьютерных наук

Кафедра системного программирования

**ОТЧЕТ**

о выполнении практического задания № 3

по дисциплине

«Структуры и алгоритмы обработки данных»

|  |  |
| --- | --- |
|  | Выполнил:  студент группы КЭ-203  Старостенок Д.В.  Проверил:  ст. преподаватель кафедры СП  Петрова Л.Н. |

Челябинск – 2022

ОГЛАВЛЕНИЕ

[1. Задача № 1 “ОЧЕРЕДИ” 2](#_Toc99446992)

[1.1. Физические величины: 2](#_Toc99446993)

[1.2. Область допустимых значений физических величин: 2](#_Toc99446994)

[1.3. Единицы измерения физических величин: 2](#_Toc99446995)

[1.4. Ограничения: 2](#_Toc99446996)

[1.5. Анализ ожидаемых результатов: 2](#_Toc99446997)

[1.6. Листинг программы: 3](#_Toc99446998)

[1.7. Контрольный тест: 4](#_Toc99446999)

[2. Задача № 2 “МНОЖЕСТВА”. 6](#_Toc99447000)

[2.1. Задача: 6](#_Toc99447001)

[2.2. Физические величины: 6](#_Toc99447002)

[2.3. Область допустимых значений физических величин: 6](#_Toc99447003)

[2.4. Единицы измерения физических величин: 6](#_Toc99447004)

[2.5. Ограничения: 6](#_Toc99447005)

[2.6. Анализ ожидаемых результатов: 6](#_Toc99447006)

[2.7. Листинг программы: 6](#_Toc99447007)

[2.8. Контрольный тест: 7](#_Toc99447008)

[Вывод 8](#_Toc99447009)

1. Задача № 1 “Очереди”

Задача:

Удалить первый и последний элементы очереди целых чисел.

* 1. Физические величины:

int, var

* 1. Область допустимых значений физических величин:

int – от -2147483648 до 2147483647

* 1. Единицы измерения физических величин:

int – целочисленный тип, 4 байта

* 1. Ограничения:

Структура очереди

* 1. Анализ ожидаемых результатов:

Очередь, в которой отсутствует первый и последний элемент

* 1. Листинг программы:

namespace PR\_3\_Queue

{

class Program

{

static void Main(string[] args)

{

Console.WriteLine("Введите количество чисел:");

int n = Convert.ToInt32(Console.ReadLine()); // Считываем количество чисел в очереди

Queue<int> numbers = new Queue<int>();

Queue<int> temp = new Queue<int>();

Random rnd = new Random();

for (int i = 0; i < n; i++) // Добавление случайных чисел в очередь

{

numbers.Enqueue(rnd.Next(-10, 10));

}

Queue<int> newNumbers = new Queue<int>(numbers); // Копируем содержимое основной очереди, чтобы показать хранящиеся элементы

var k = n;

Console.WriteLine("Содержимое очереди:");

while (k > 0)

{

var num = newNumbers.Dequeue();

Console.WriteLine(num);

k--;

}

k = n;

Console.WriteLine("Содержимое очереди без последнего элемента:");

while (k > 1)

{

var num = numbers.Dequeue();

Console.WriteLine(num);

temp.Enqueue(num); // перенос в temp всех элементов очереди кроме последнего

k--;

}

numbers.Clear(); // очищаем очередь от оставшегося числа

temp.Dequeue(); // убираем первый элемент из временного стека

k = n - 2;

while (k > 0) // перенос из temp в основную очередь

{

var num = temp.Dequeue();

numbers.Enqueue(num);

k--;

}

Console.WriteLine("Содержимое очереди без первого и последнего элемента:");

k = n - 2;

while (k > 0) // Вывод чисел из основной очереди

{

var num = numbers.Dequeue();

Console.WriteLine(num);

k--;

}

}

}

}

* 1. Контрольный тест:

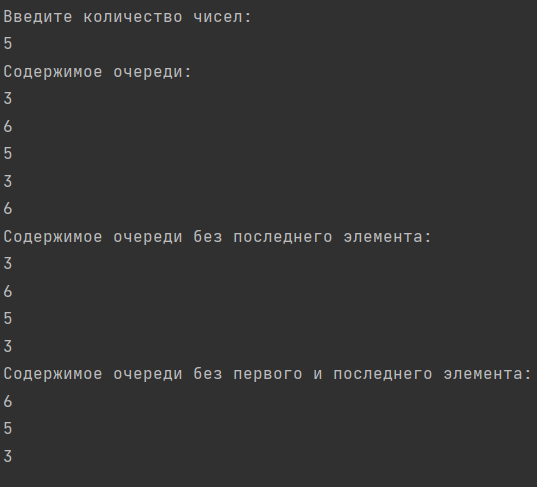


Рис. 1. Тест очереди 1

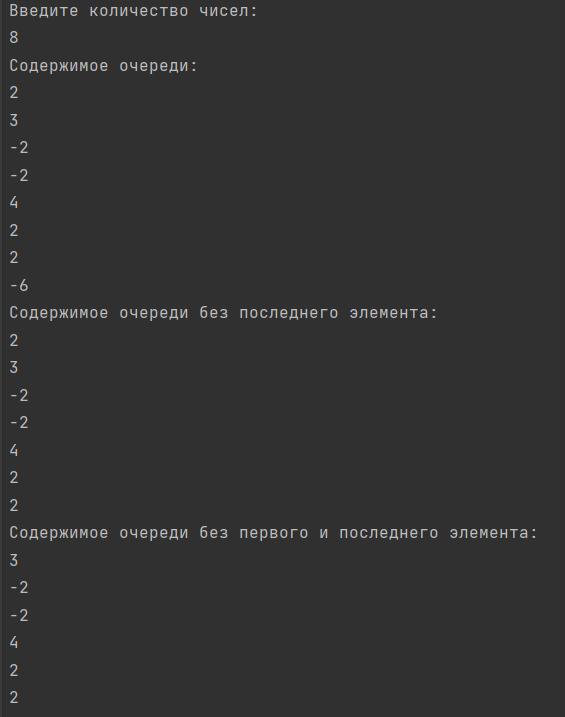


Рис. 2. Тест очереди 2

1. Задача № 2 “множества”.
   1. Задача:

Пусть дан текст, состоящий из строчных латинских букв и цифр. Определить, каких букв – гласных или согласных – больше в этом тексте.

* 1. Физические величины:

int, string

* 1. Область допустимых значений физических величин:

int – от -2147483648 до 2147483647

string – ~1 миллиард символов

* 1. Единицы измерения физических величин:

int – целочисленный тип, 4 байта

string – строки, 2 ГБ

* 1. Ограничения:

Длина введенного слова

* 1. Анализ ожидаемых результатов:

Подсчет гласных и согласных слов, определение большего количества

* 1. Листинг программы:

namespace PR\_3\_Sets

{

class Program

{

private static readonly HashSet<char> Vowels = new("aeiouy"); // Множество гласных букв

private static readonly HashSet<char> Consonants = new("bcdfghjklmnpqrstvwxz"); // Множество согласных букв

static void Main(string[] args)

{

Console.WriteLine("Введите слово");

string text = Console.ReadLine()?.ToLower(); // Считывание строки и перевод в нижний регистр

if (text != null)

{

int vowelsCount = text.Count(x => Vowels.Contains(x)); // Подсчет гласных букв

int consonantsCount = text.Count(x => Consonants.Contains(x)); // Подсчет согласных букв

if (vowelsCount > consonantsCount)

Console.WriteLine("Гласных больше");

else if (vowelsCount < consonantsCount)

Console.WriteLine("Согласных больше");

else if (vowelsCount == consonantsCount)

Console.WriteLine("Количество гласных и согласных равны");

}

}

}

}

* 1. Контрольный тест:

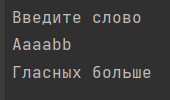


Рис. 3. Тест в котором больше гласных

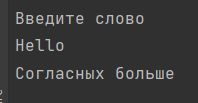


Рис. 4. Тест в котором больше согласных

Вывод

В результате проделанной работы были получены навыки реализации и работы с очередями и множествами. По заданию созданы 2 программы, которые помогли закрепить полученные навыки.