**18 Коллекции. Классы-прототипы**

Задание №1. Решить задачи с использованием класса Stack. Даны две строки s1 и s2. Из каждой можно читать по одному символу. Выяснить, является ли строка s2 обратной s1.

Листинг программы:

class Demo{

static void Main(){

string s1 = "asdfghj";

string s2 = "jhgfdsa";

Stack<char> chStack = new Stack<char>();

foreach (char ch in s1)

chStack.Push(ch);

int j = 0;

bool flag = true;

while (chStack.Count != 0)

{

if (s2[j] != chStack.Pop())

{

flag = false;

break;

}

j++;

}

Console.WriteLine("Строка s1: {0}, строка s2: {1}",s1,s2);

Console.WriteLine("{0}", (flag) ? "строки обратны друг другу" : "строки не обратны друг другу");

}}

Таблица 18.1 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| Строка s1: asdfghj, строка s2: jhgfdsa | строки обратны друг другу |

Анализ результатов:

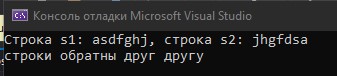


Рисунок 18.1 – Результат работы программы

Источник: собственная разработка

Задание №2. Решить следующие задачи с использованием класса Queue. Дан файл, содержащий числа. За один просмотр файла напечатать элементы файла в следующем порядке: сначала все числа, из интервала [a,b], потом все числа, меньшие a, потом все числа, большие b, сохраняя исходный порядок в каждой группе чисел.

Листинг программы:

class Program{

static void Main(string[] args){

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

Console.WriteLine("Введите значение а: ");

int a = Convert.ToInt16(Console.ReadLine());

Console.WriteLine("Введите значение b: ");

int b = Convert.ToInt16(Console.ReadLine());

FileStream file = new FileStream(@"f.txt", FileMode.Open);

StreamReader reader = new StreamReader(file);

string s = reader.ReadToEnd();

reader.Close();

string[] sp = s.Split(' ');

for (int i = 0; i < sp.Length; i++)

{

queue.Enqueue(Convert.ToInt16(sp[i]));

}

Console.WriteLine();

Console.WriteLine("1) Элементы в диапазоне от a до b :");

foreach (int n in queue)

{

if ((n > a) && (n < b)) Console.WriteLine(n);

}

Console.WriteLine();

Console.WriteLine("2) Элементы меньшие а :");

foreach (int n in queue)

{

if (n < a) Console.WriteLine(n);

}

Console.WriteLine();

Console.WriteLine("3) Элементы большие b :");

foreach (int n in queue)

{

if (n > b) Console.WriteLine(n);

}

Console.ReadLine();

}

}

Таблица 18.2 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| 1..20  5  10 | 1) Элементы в диапазоне от a до b  6 7 8 9  2) Элементы меньшие а :  1 2 3 4  3) Элементы большие b :  11 12 13 14 15 16 17 18 19 20 |

Анализ результатов:

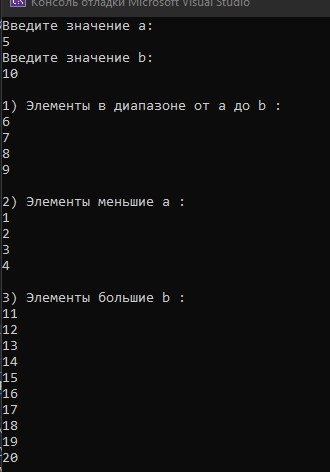


Рисунок 18.2 – Результат работы программы

Источник: собственная разработка