Министерство образования и молодежной политики Свердловской области



ГАПОУ СО «Екатеринбургский колледж транспортного строительства»

# Отчёт по программе «.«Коллекции c#. Собственные методы для работы

# с коллекциями»»

Выполнил: Мартынов Арсений Сергеевич

Группа: ПР-21

Преподаватель: Мирошниченко Г.В

2023

**Задание**

Для заданного значения n запишем в стек все числа от 1 до n, а затем

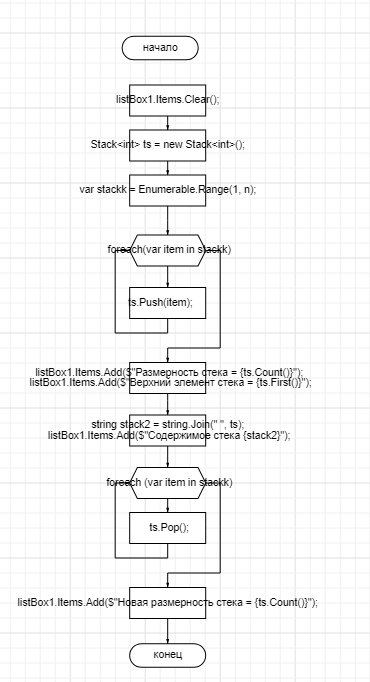
извлечем из стека.

**Входные и выходные данные**

Значение numericupdown1 – входные данные

Stack<int>ts - вывод элементов стека

**Блок-схема**

****

**Листинг программы**

listBox1.Items.Clear();

Stack<int> ts = new Stack<int>();

int n = Convert.ToInt32(numericUpDown1.Value);

var stackk = Enumerable.Range(1, n);

foreach(var item in stackk)

{

ts.Push(item);

}

listBox1.Items.Add($"Размерность стека = {ts.Count()}");

listBox1.Items.Add($"Верхний элемент стека = {ts.First()}");

string stack2 = string.Join(" ", ts);

listBox1.Items.Add($"Содержимое стека {stack2}");

foreach (var item in stackk)

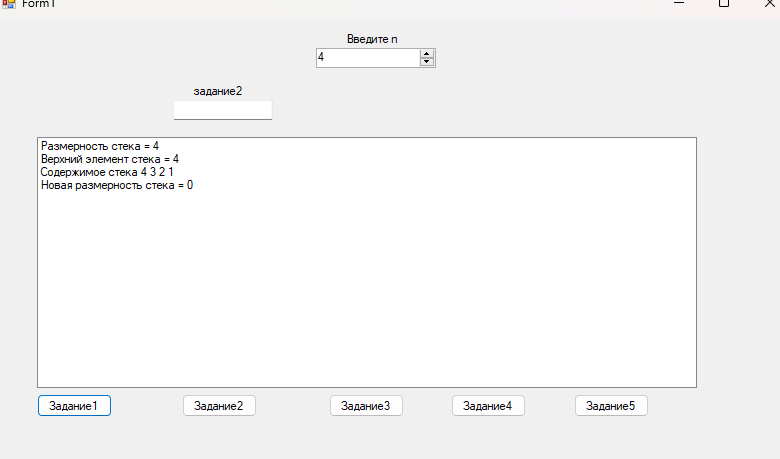
{

ts.Pop();

}

listBox1.Items.Add($"Новая размерность стека = {ts.Count()}");

**Тестовые ситуации**

****

**Задача**

Пользователь вводит математическое выражение, которое записывается в

текстовый файл. - t.txt

А) Проверить баланс круглых скобок в данном выражении. Выдать

информацию о балансе на экран.

скобки сбалансированы

Возможно лишняя ( скобка в позиции:5

Б) Если скобок не хватает, то добавить недостающие скобки, если скобки

лишние, то удалить лишние скобки. Записать новое математическое

выражение в файл t1.txt

**Входные и выходные данные**

А)

Строка texbox1

Б)

Строка texbox1

**Выходные данные**

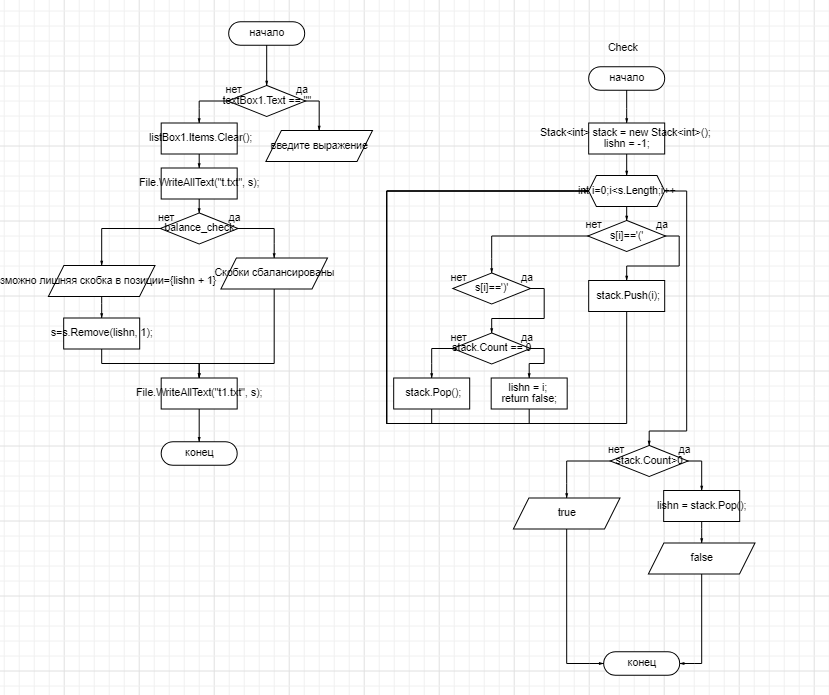
А)

Строка в listbox1, текстовый файл t.txt

Б)

Строка в listbox1, текстовый файл t1.txt

**Блок-схема**

****

**Листинг программы**

static bool Check(string s,out int lishn)

{

Stack<int> stack = new Stack<int>();

lishn = -1;

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

{

if (s[i] == '(')

{

stack.Push(i);

}

else if (s[i]==')')

{

if (stack.Count == 0)

{

lishn = i;

return false;

}

else { stack.Pop(); }

}

}

if (stack.Count>0)

{

lishn = stack.Pop();

return false;

}

return true;

}

if (textBox1.Text == "")

{

MessageBox.Show("введите выражение");

}

else

{

listBox1.Items.Clear();

string s = textBox1.Text;

File.WriteAllText("t.txt", s);

bool balance\_check = Check(s, out int lishn);

if (balance\_check)

{

listBox1.Items.Add("Скобки сбалансированы");

}

else

{

listBox1.Items.Add($"Возможно лишняя скобка в позиции={lishn + 1}");

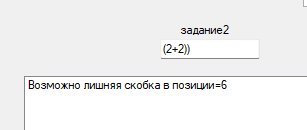
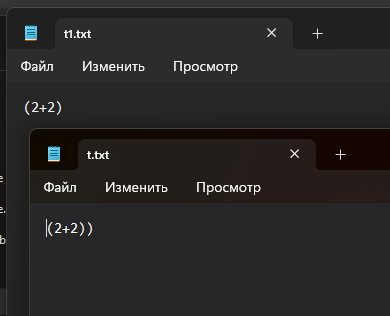
s=s.Remove(lishn, 1);

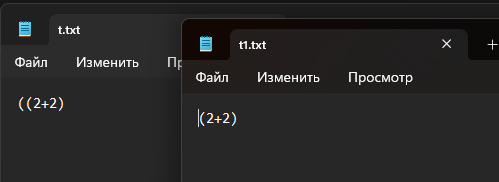
}

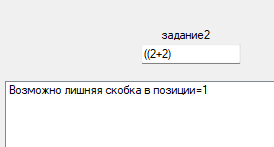
File.WriteAllText("t1.txt", s);

}

**Тестовые ситуации**

****

****

****

**Задание 4:**

В текстовом файле записана информация о людях (фамилия, имя,

отчество, возраст, вес через пробел). Вывести на экран вначале

информацию о людях младше 40 лет, а затем информацию о всех

остальных.

**Иванов Сергей Николаевич 21 64**

**Петров Игорь Юрьевич 45 88**

**Семёнов Михаил Алексеевич 20 70**

**Пиманов Александр Дмитриевич 53 101**

**Листинг**Queue<People> people = new Queue<People>();

listBox1.Items.Clear();

if (File.Exists("f2.txt"))

{

if (new FileInfo("f2.txt").Length != 0)

{

StreamReader sr = File.OpenText("f2.txt");

while (!sr.EndOfStream)

{

string s = sr.ReadLine();

People p = new People(s);

people.Enqueue(p);

}

sr.Close();

var otb = from t in people where (t.Get\_year() < 40) select t;

var otb2 = from t in people where (t.Get\_year() >= 40) select t;

foreach (People s1 in otb)

{

listBox1.Items.Add(s1.Output());

}

foreach(People s2 in otb2)

{

listBox1.Items.Add(s2.Output());

}

people.Clear();

}

else

{

MessageBox.Show("файл пуст");

}

}

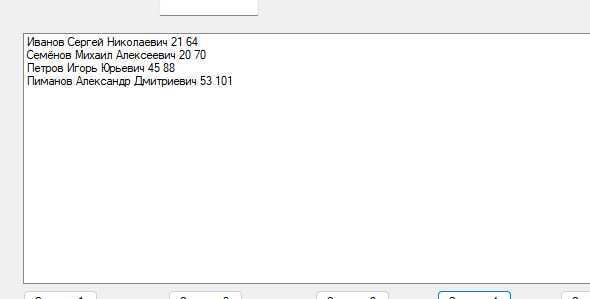
else

{

MessageBox.Show("такого файла нет");

}

**Тестовые ситуации**

****

**Задание 5:**

5.В 2-х текстовых файлах записана информация о людях (фамилия, имя,

отчество через пробел) и(возраст, вес через пробел). Вывести на экран

информацию о людях, отсортированную по возрасту, с группировкой по

первой букве фамилии.

**Листинг**

Queue<string> name = new Queue<string>(File.ReadAllLines("name.txt"));

Queue<string> age = new Queue<string>(File.ReadAllLines("age.txt"));

var people = name.Zip(age, (name1, info1) => new { Name = name1, Info = info1 }).Select(x => new { Fullname = x.Name, Age1 = Convert.ToInt32(x.Info.Split(' ')[0]), Weight = Convert.ToInt32(x.Info.Split(' ')[1]) }).OrderBy(x => x.Age1).GroupBy(x => x.Fullname.Split(' ')[0][0]);

foreach (var group in people)

{

listBox1.Items.Add($"Люди с фамилией на букву: {group.Key}");

foreach (var person in group)

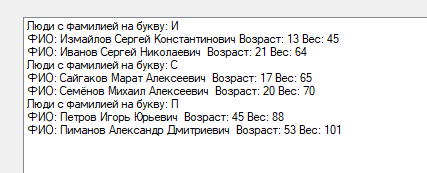
{

listBox1.Items.Add($"ФИО: {person.Fullname} Возраст: {person.Age1} Вес: {person.Weight}");

}

}

**Тестовые ситуации**

****