## Министерство образования и молодежной политики Свердловской области



ГАПОУ СО «Екатеринбургский колледж транспортного строительства»

Отчёт по программе «**Коллекции и WindowsForms**»

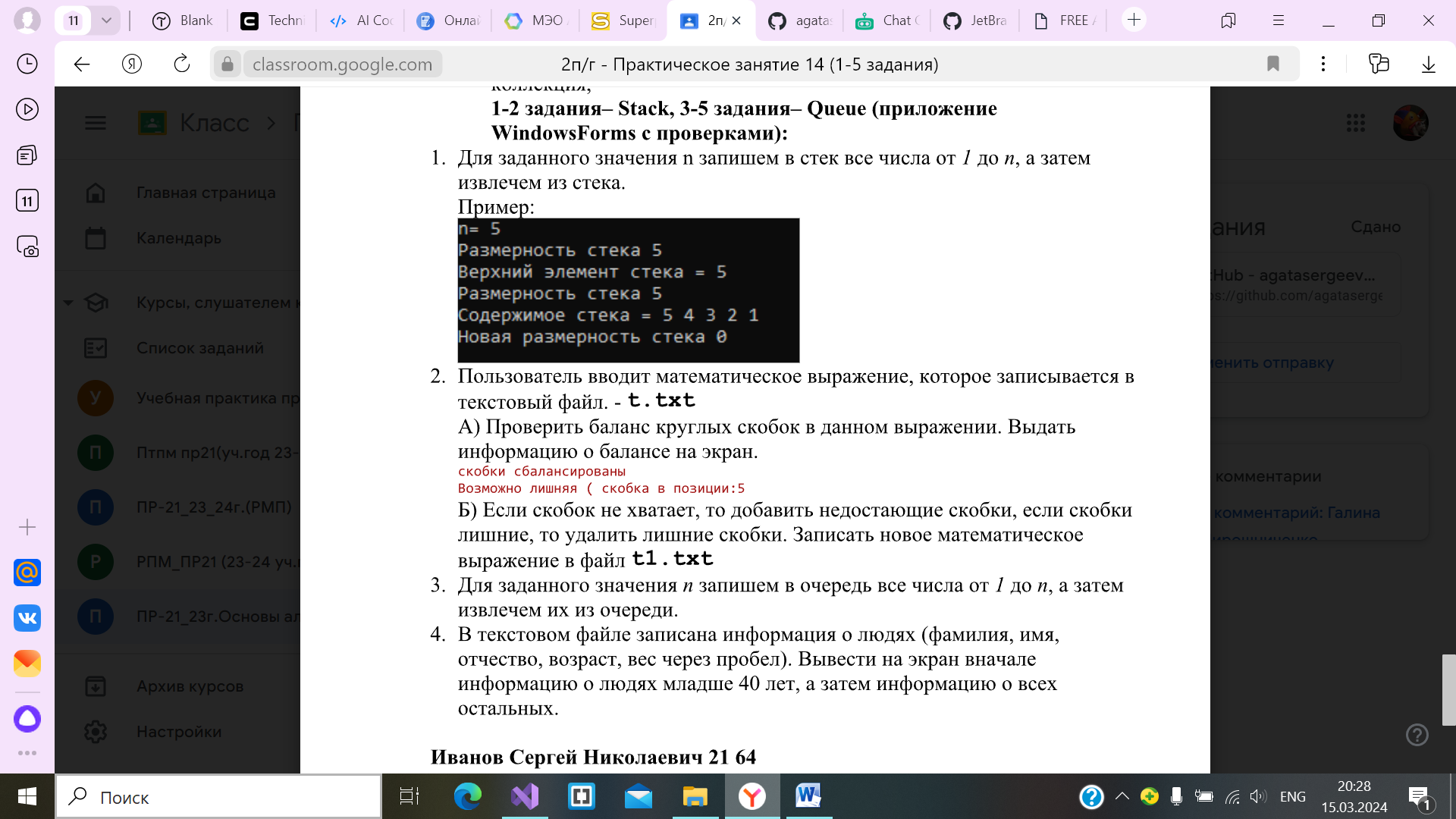
Выполнил: Сергеева Агата Андреевна

Группа: ПР-21

Преподаватель: Мирошниченко Г.В

2024 г.

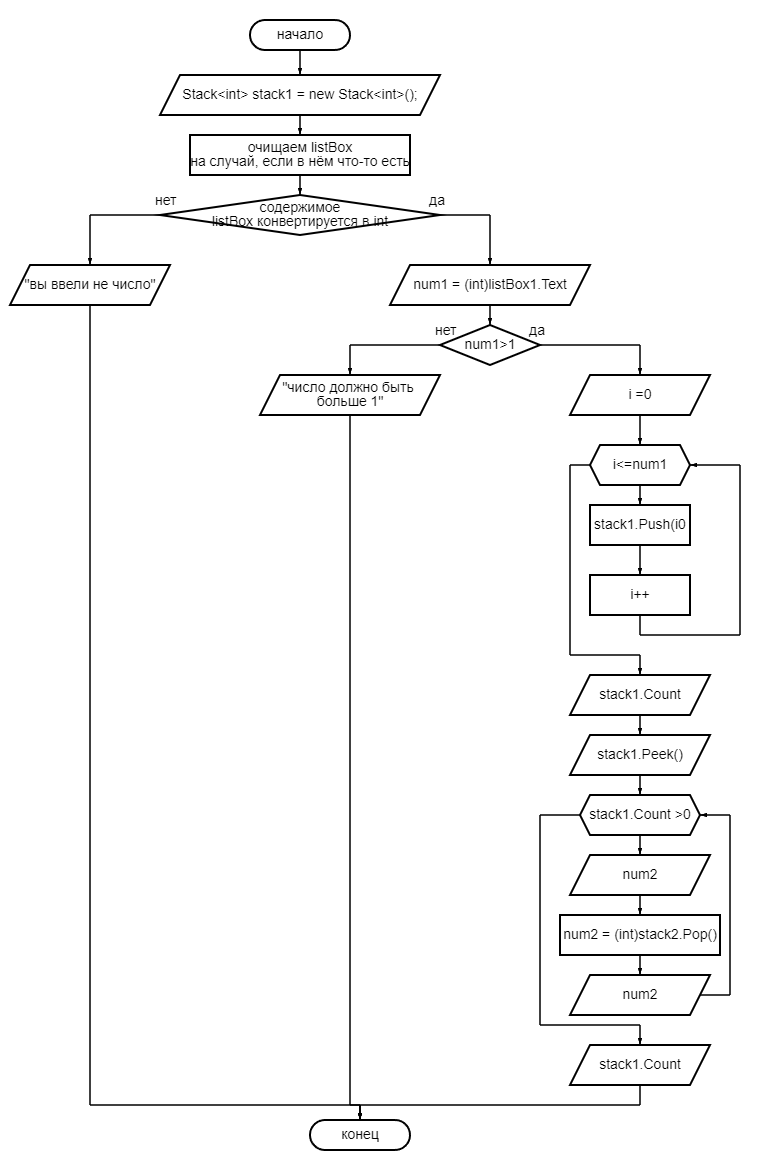
**Задание 1:**



**Входные данные:** n - число, int.

**Выходные данные:** верхний элемент стека, int; размерности в до и после извлечения, int; вывод содержимого, string.

**Блок-схема:**



**Листинг программы:**

Stack<int> stack1 = new Stack<int>();

listBox1.Items.Clear();

if (int.TryParse((textBox1.Text), out int a))

{

int num1 = Convert.ToInt32(textBox1.Text);

if (num1 > 1)

{

for (int i = 1; i <= num1; i++)

stack1.Push(i);

listBox1.Items.Add($"Размерность стека: {stack1.Count}");

listBox1.Items.Add($"Верхний элемент стека: {stack1.Peek()}");

while (stack1.Count > 0)

{

int num2 = (int)stack1.Pop();

listBox1.Items.Add(num2);

}

listBox1.Items.Add($"Новая размерность стека: {stack1.Count}");

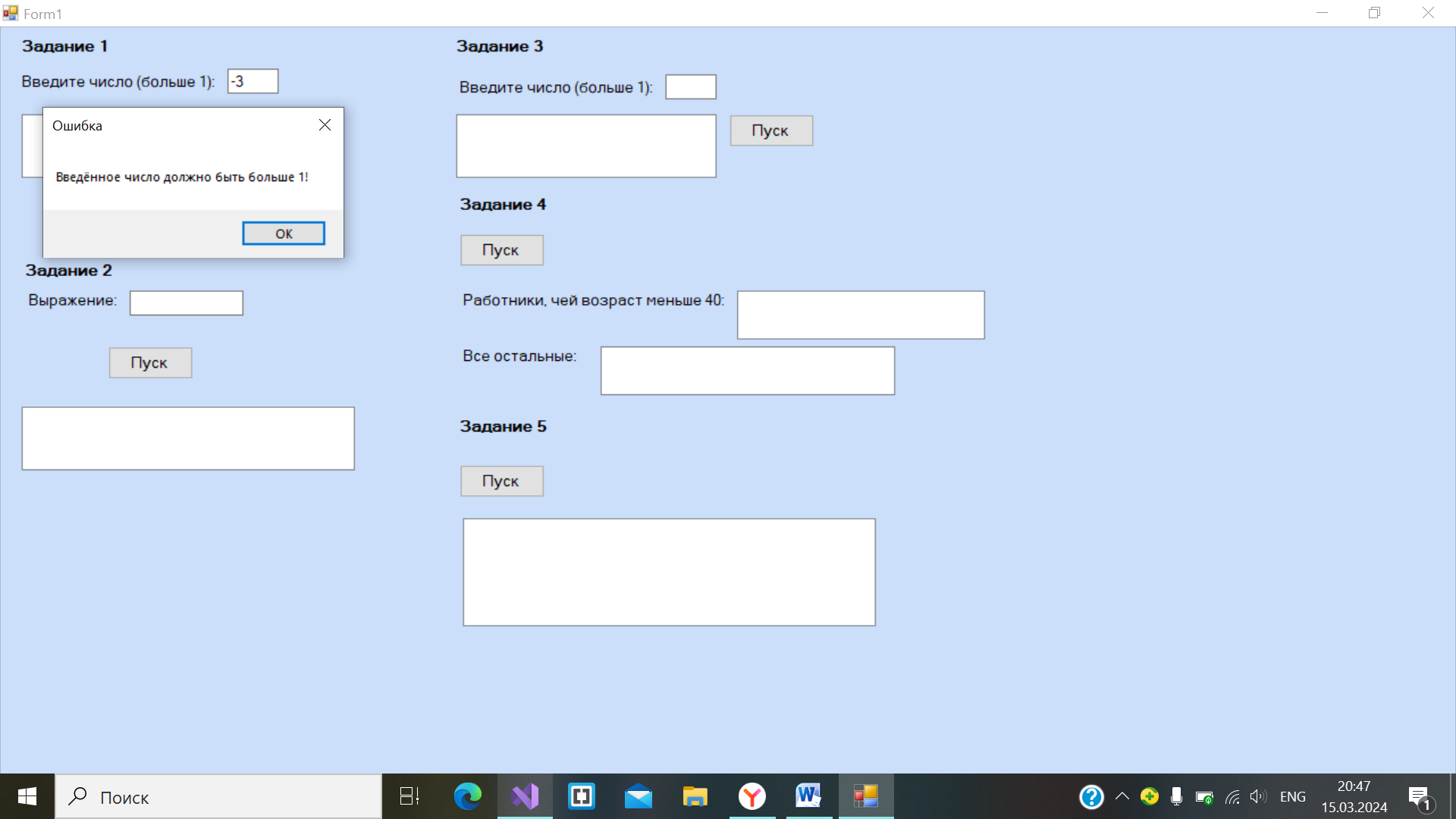
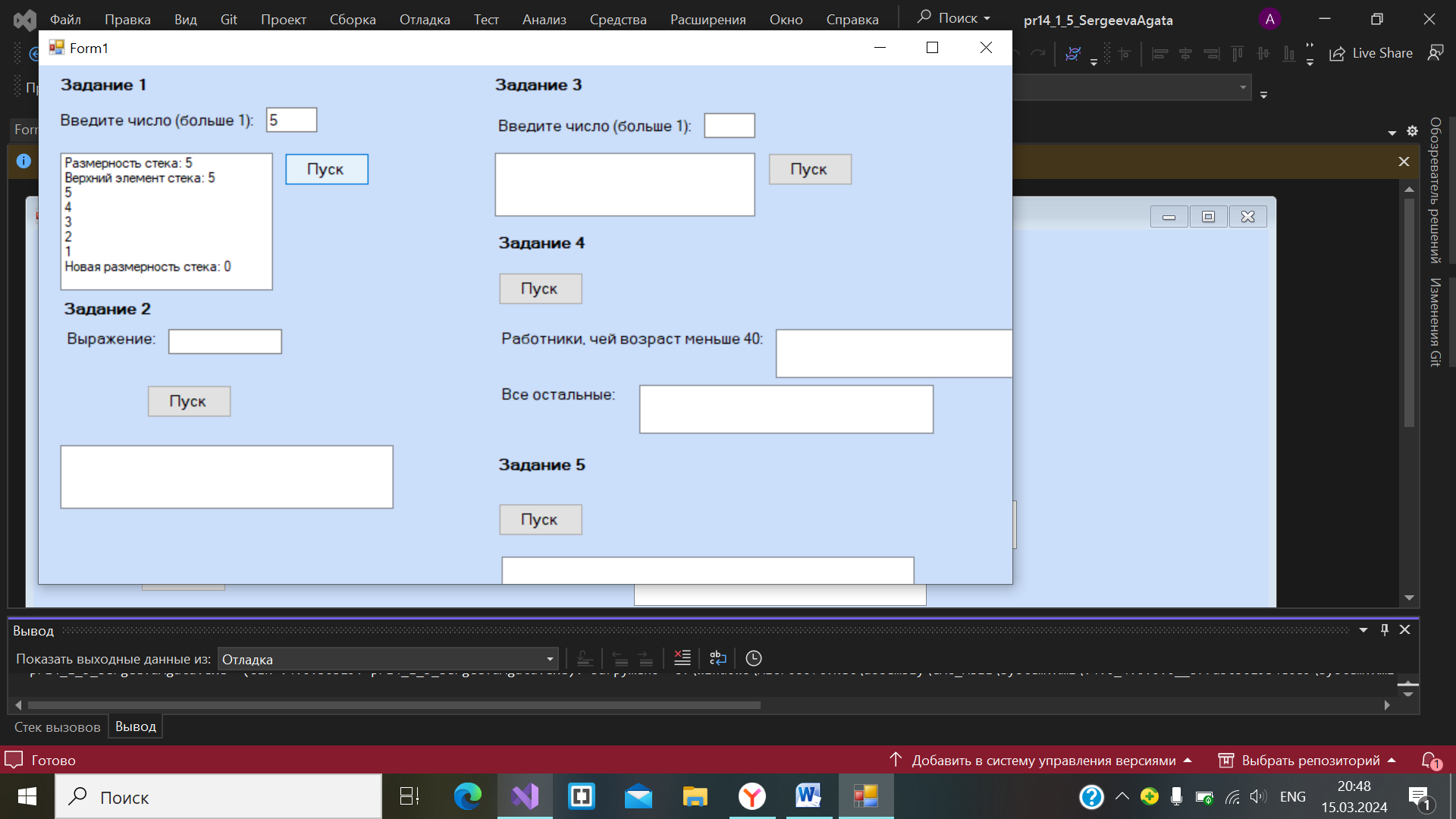
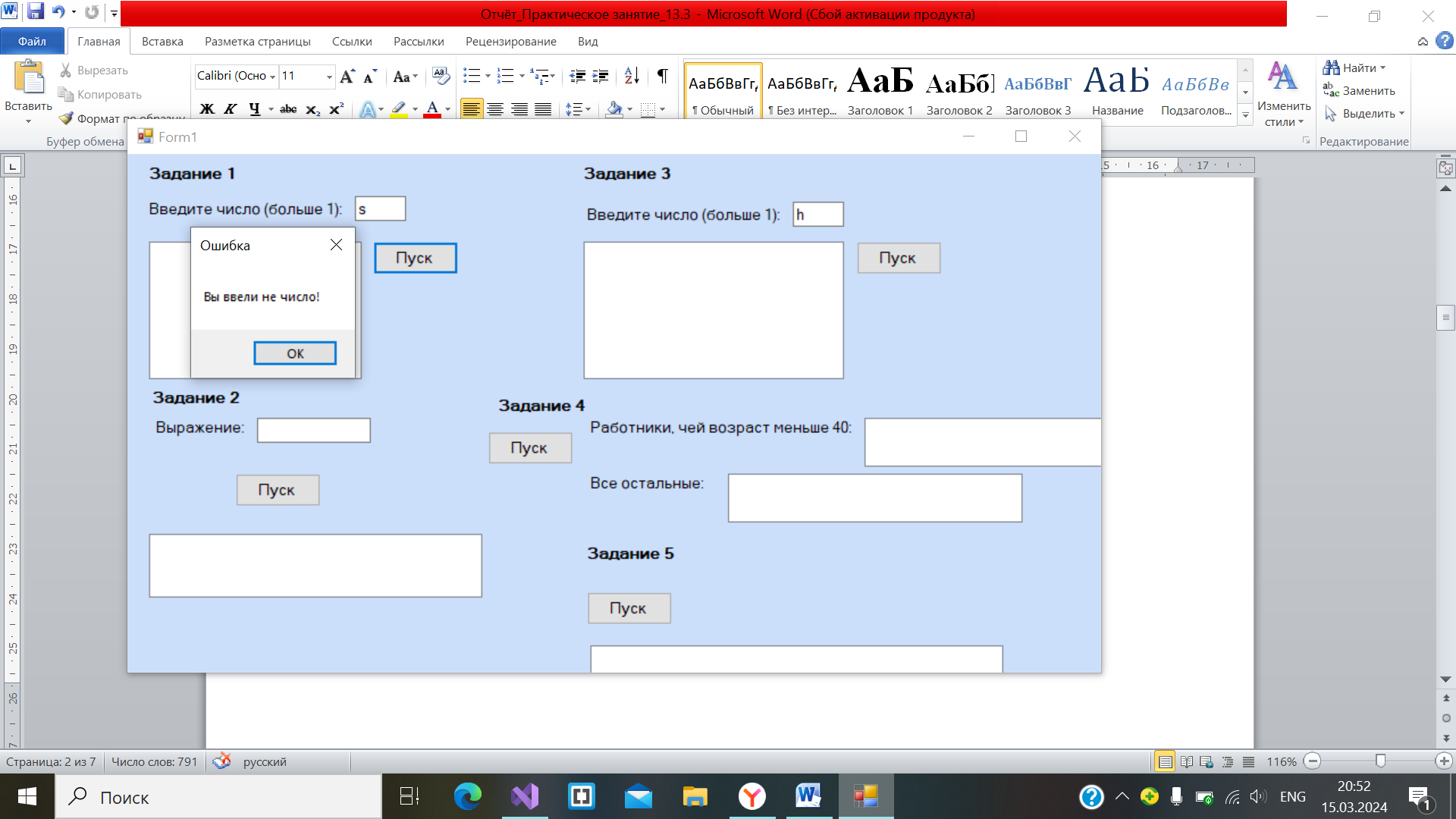
}

else MessageBox.Show("Введённое число должно быть больше 1!","Ошибка",MessageBoxButtons.OK);

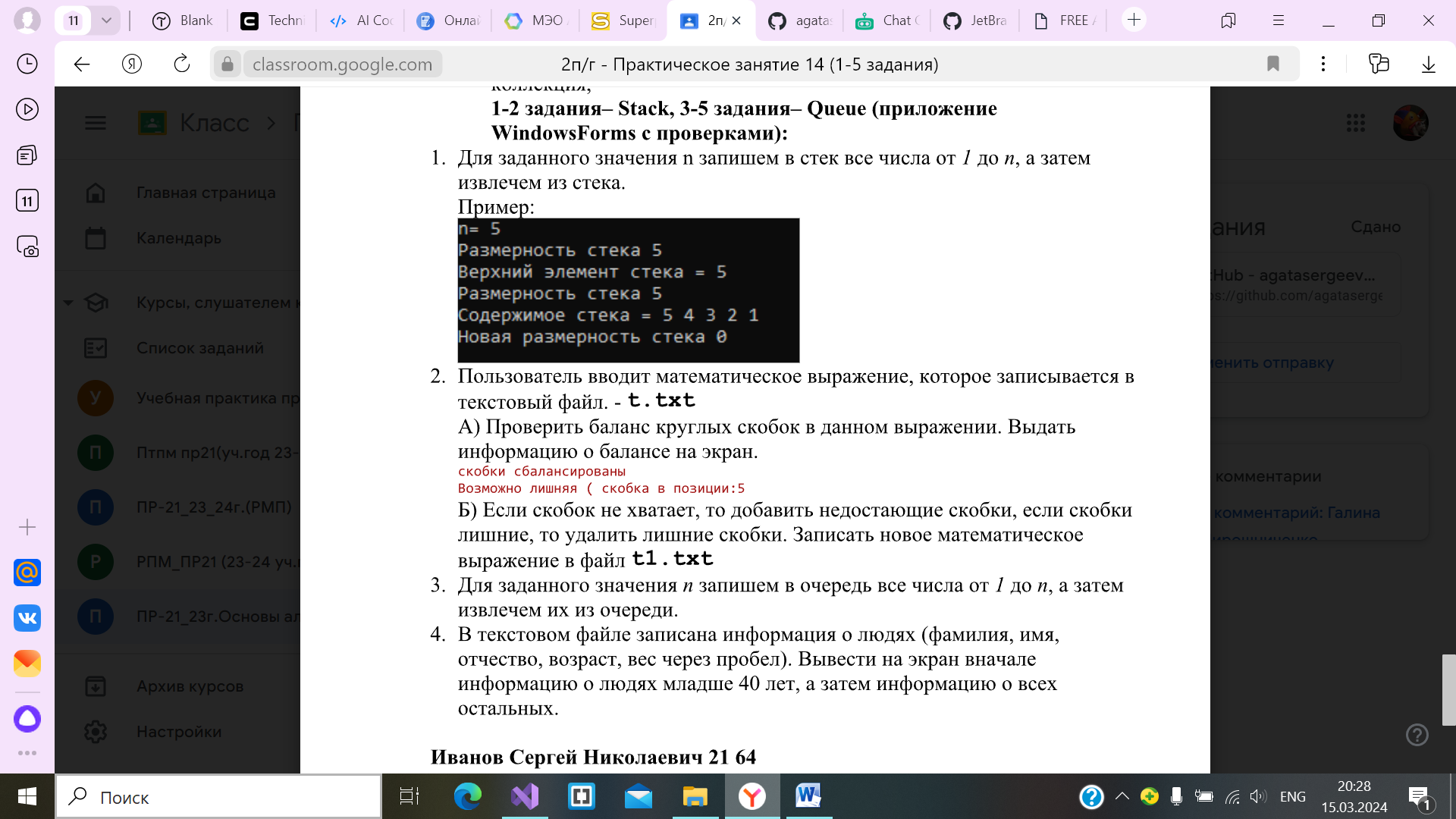
}

else MessageBox.Show("Вы ввели не число!", "Ошибка", MessageBoxButtons.OK);

**Тестовые ситуации:**

**Задание 2:**



**Входные данные:** математическое выражение, string.

**Выходные данные:** математическое выражение (возможно исправленное), string;

**Блок-схема:**

**Листинг программы:**

int count\_1 = 0, count\_2 = 0;

if (textBox2.Text != "")

{

StreamWriter p = File.CreateText("t.txt");

p.WriteLine(textBox2.Text);

p.Close();

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

{

if (File.Exists("t1.txt"))

{

StreamReader skob = new StreamReader("t.txt");

string prim = skob.ReadLine();

if (prim.IndexOf(")") - prim.IndexOf("(") > 0)

{

foreach (char z in prim)

{

if (z == ')') count\_1++;

else if (z == '(') count\_2++;

}

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

{

if (prim[i] == '(' && prim[i + 1] == ')') textBox2.Text = textBox2.Text.Remove(i, 2);

}

if (count\_1 == count\_2) listBox2.Items.Add($"В выражениии {textBox2.Text} скобки сбалансированы");

else if (count\_1 > count\_2)

{

listBox2.Items.Add($"В выражениии {textBox2.Text} скобки не сбалансированы.");

listBox2.Items.Add($"Лишняя скобка ) на позиции {prim.LastIndexOf(")") + 1}");

DialogResult d = MessageBox.Show("Вы хотите добавить скобку?", "Вопрос", MessageBoxButtons.YesNo);

StreamWriter new\_ur = new StreamWriter("t1.txt");

if (d == DialogResult.Yes)

{

new\_ur.WriteLine("(" + textBox2.Text);

}

else

{

new\_ur.WriteLine(textBox2.Text.Remove(prim.LastIndexOf(")"), 1));

}

new\_ur.Close();

}

else

{

listBox2.Items.Add($"В выражениии {textBox2.Text} скобки не сбалансированы.");

listBox2.Items.Add($"Лишняя скобка ( на позиции {prim.IndexOf("(") + 1}");

DialogResult d = MessageBox.Show("Вы хотите добавить скобку?", "Вопрос", MessageBoxButtons.YesNo);

StreamWriter new\_ur = new StreamWriter("t1.txt");

if (d == DialogResult.Yes)

{

new\_ur.WriteLine(textBox2.Text + ")");

}

else

{

new\_ur.WriteLine(textBox2.Text.Remove(prim.IndexOf("("), 1));

}

new\_ur.Close();

}

}

else MessageBox.Show("Скобки расставлены неправильно", "Ошибка", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

else MessageBox.Show("Файл 't.txt'не найден", "Ошибка", MessageBoxButtons.OK, MessageBoxIcon.Error);

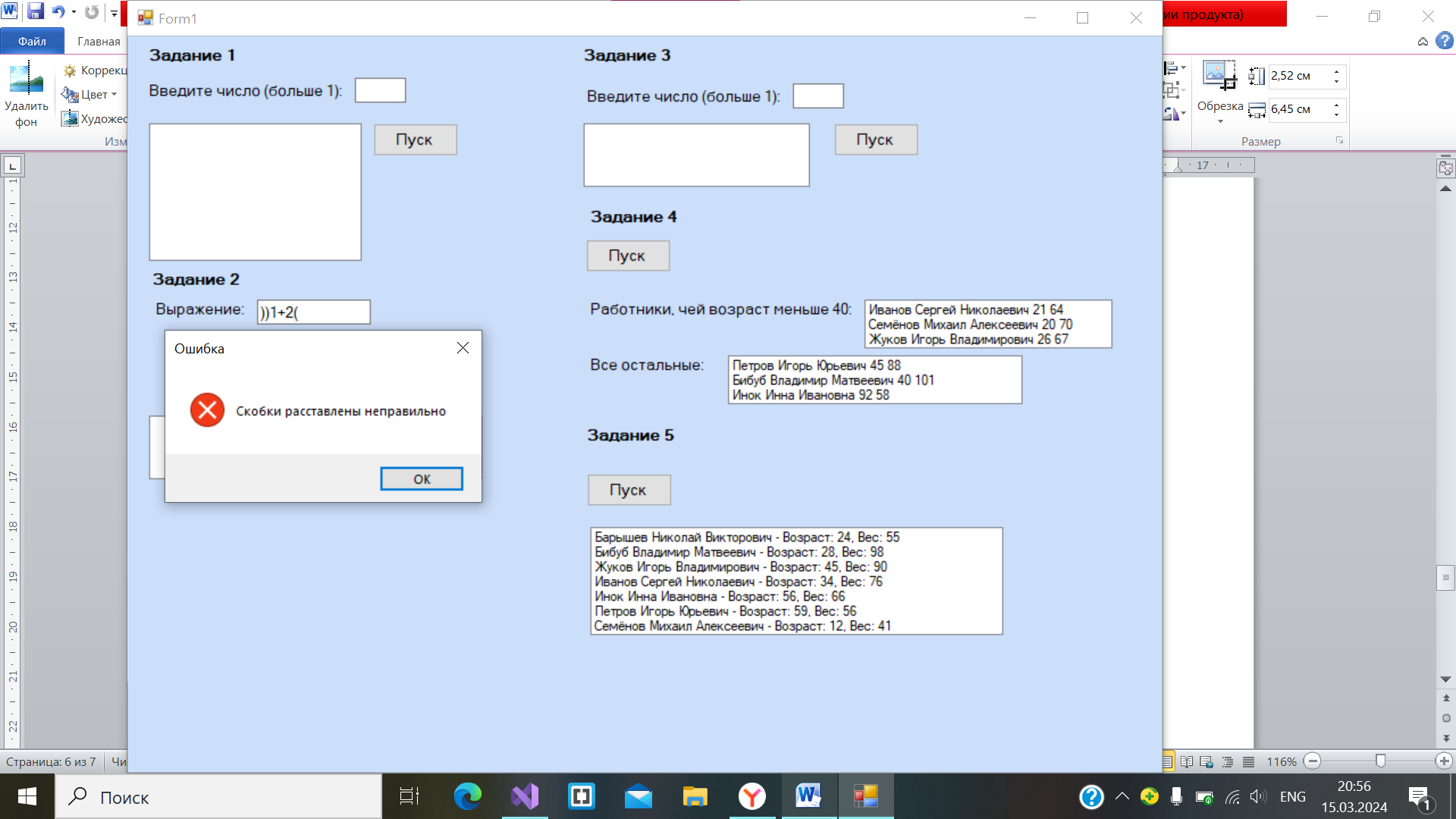
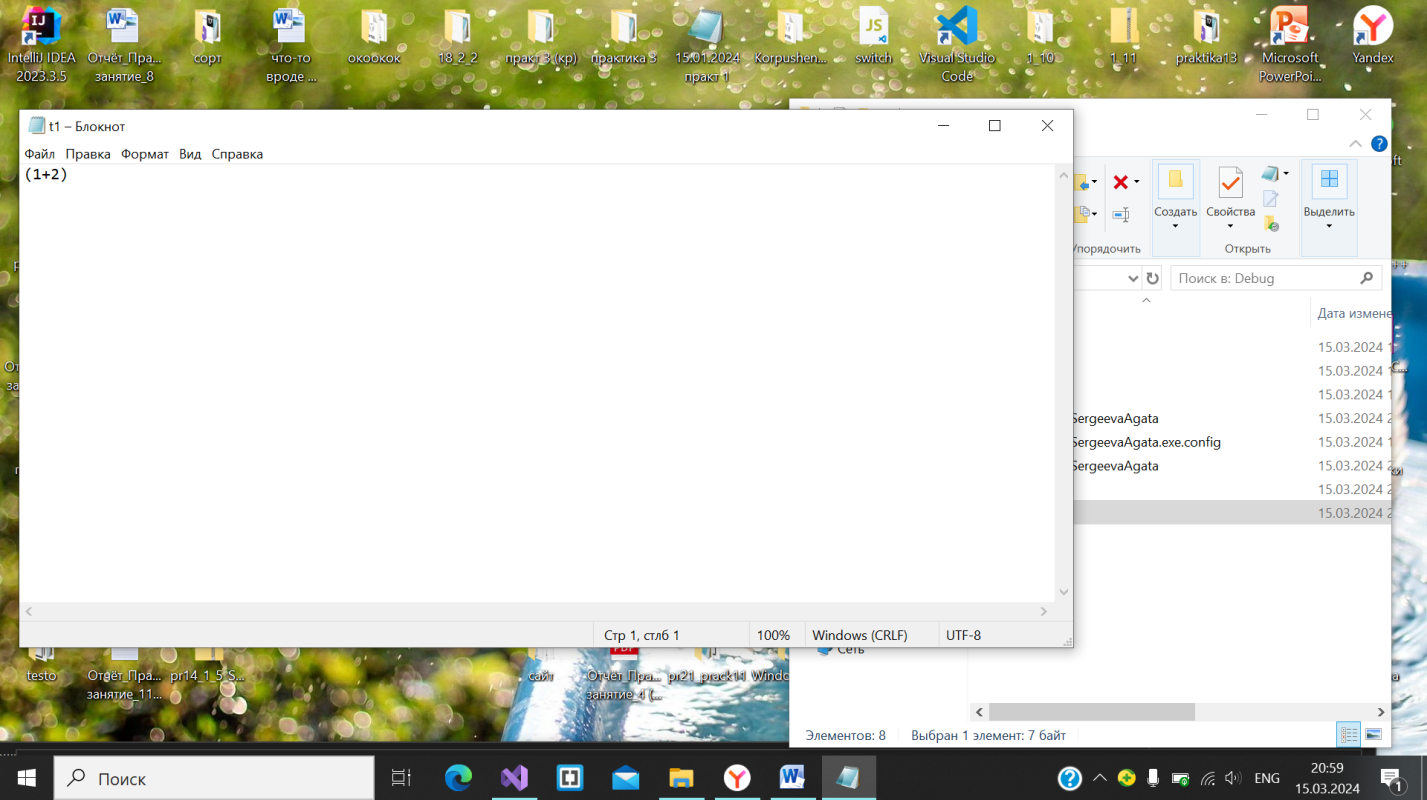
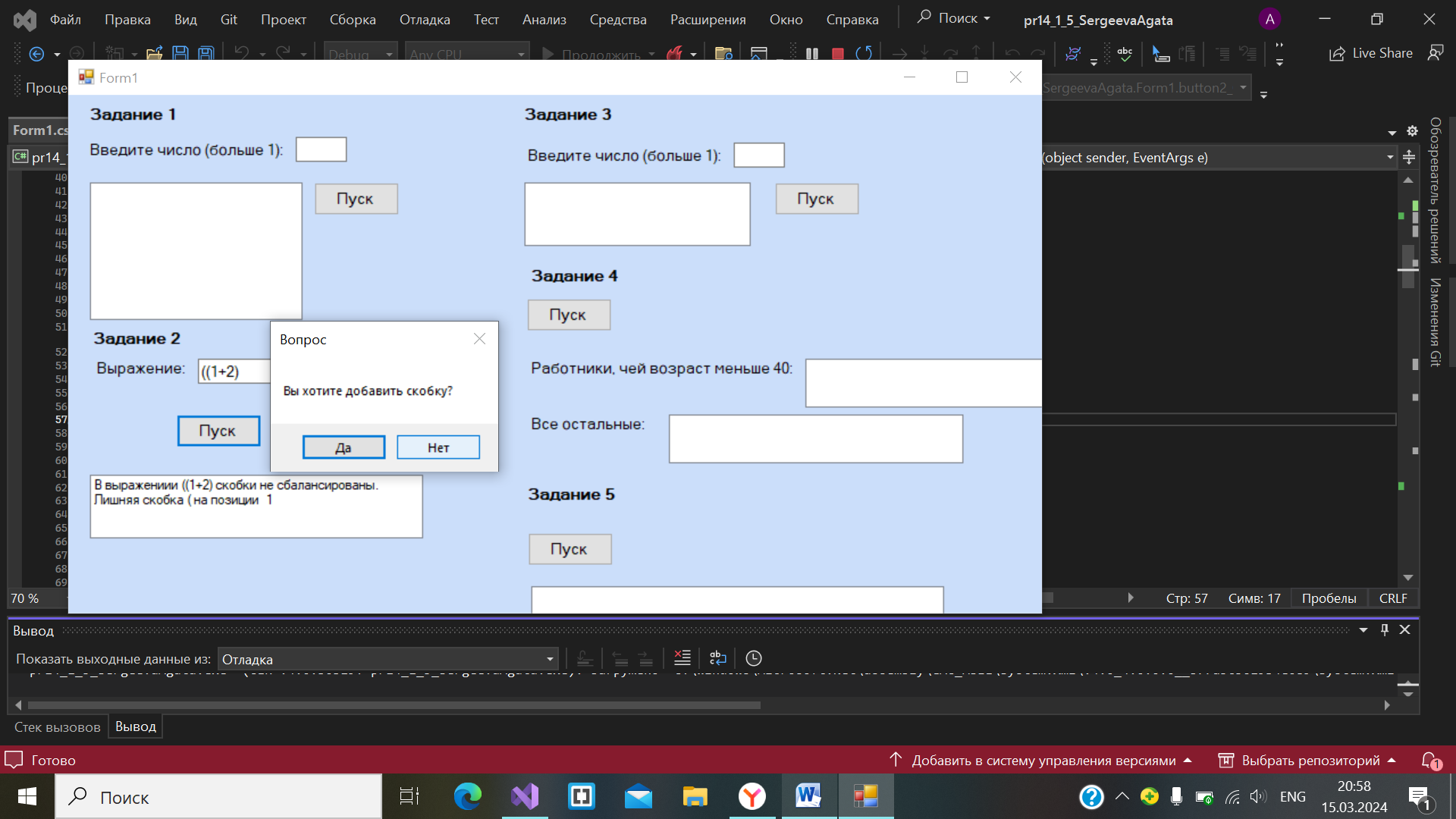
}

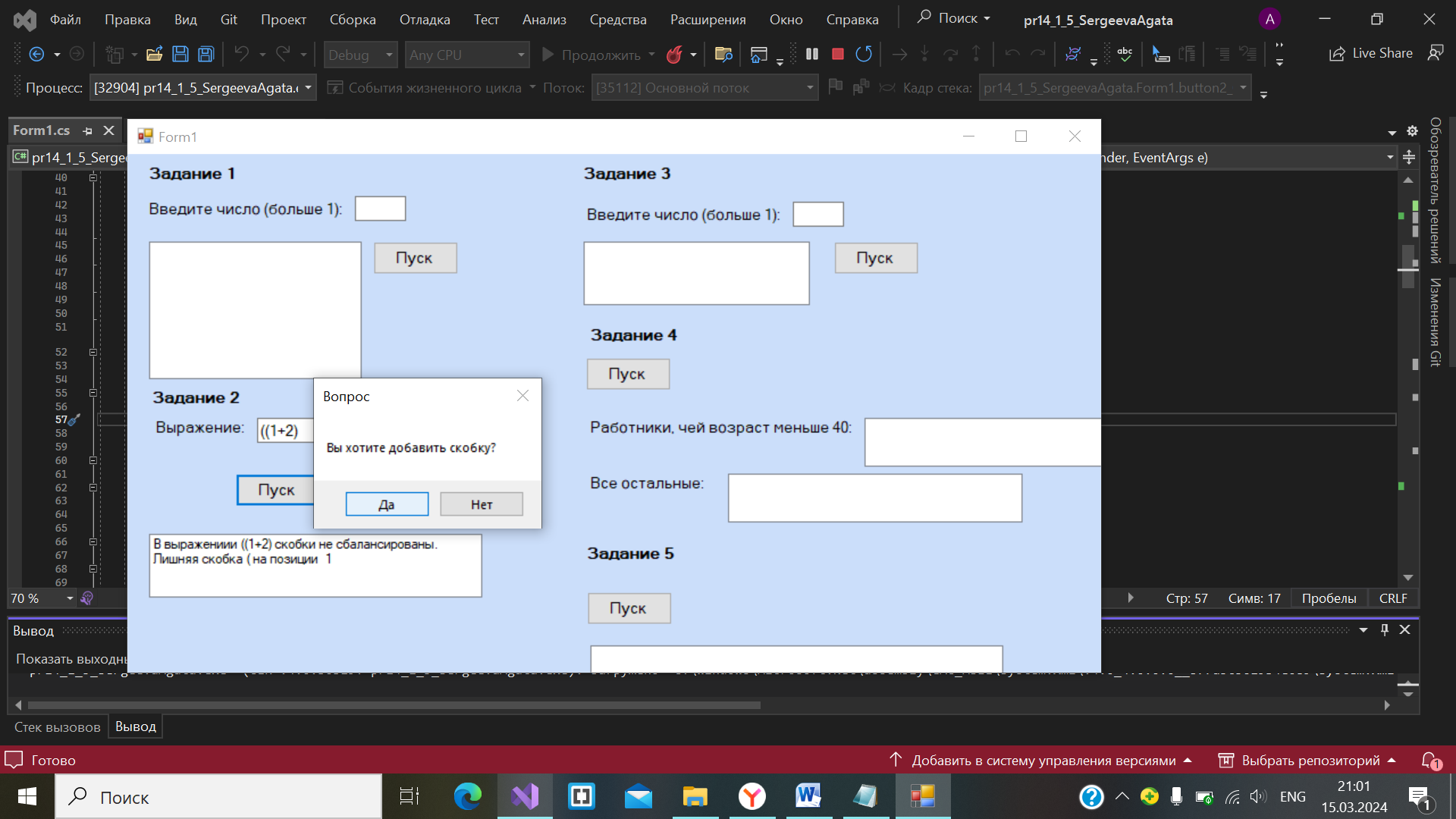
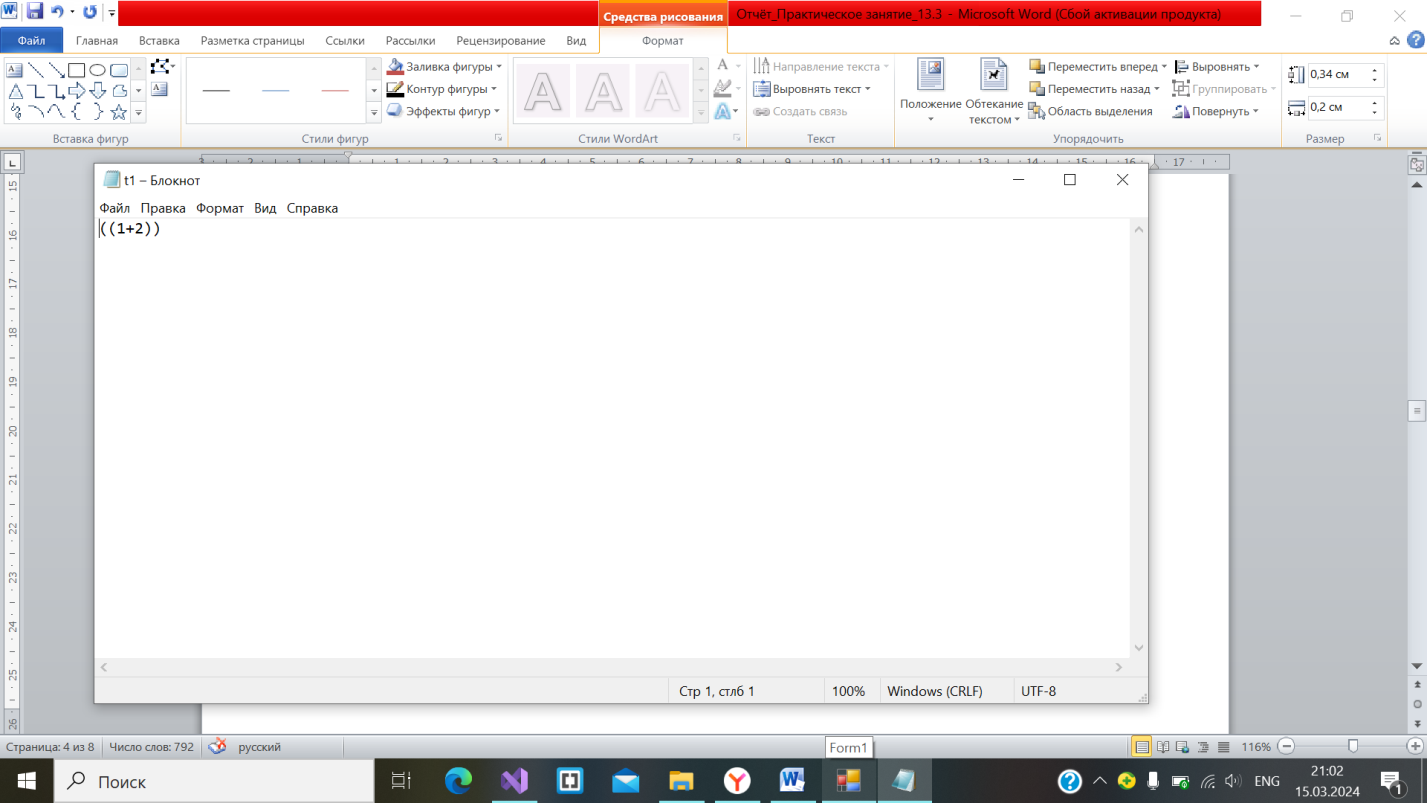
else MessageBox.Show("Файл 't.txt'не найден", "Ошибка", MessageBoxButtons.OK, MessageBoxIcon.Error);

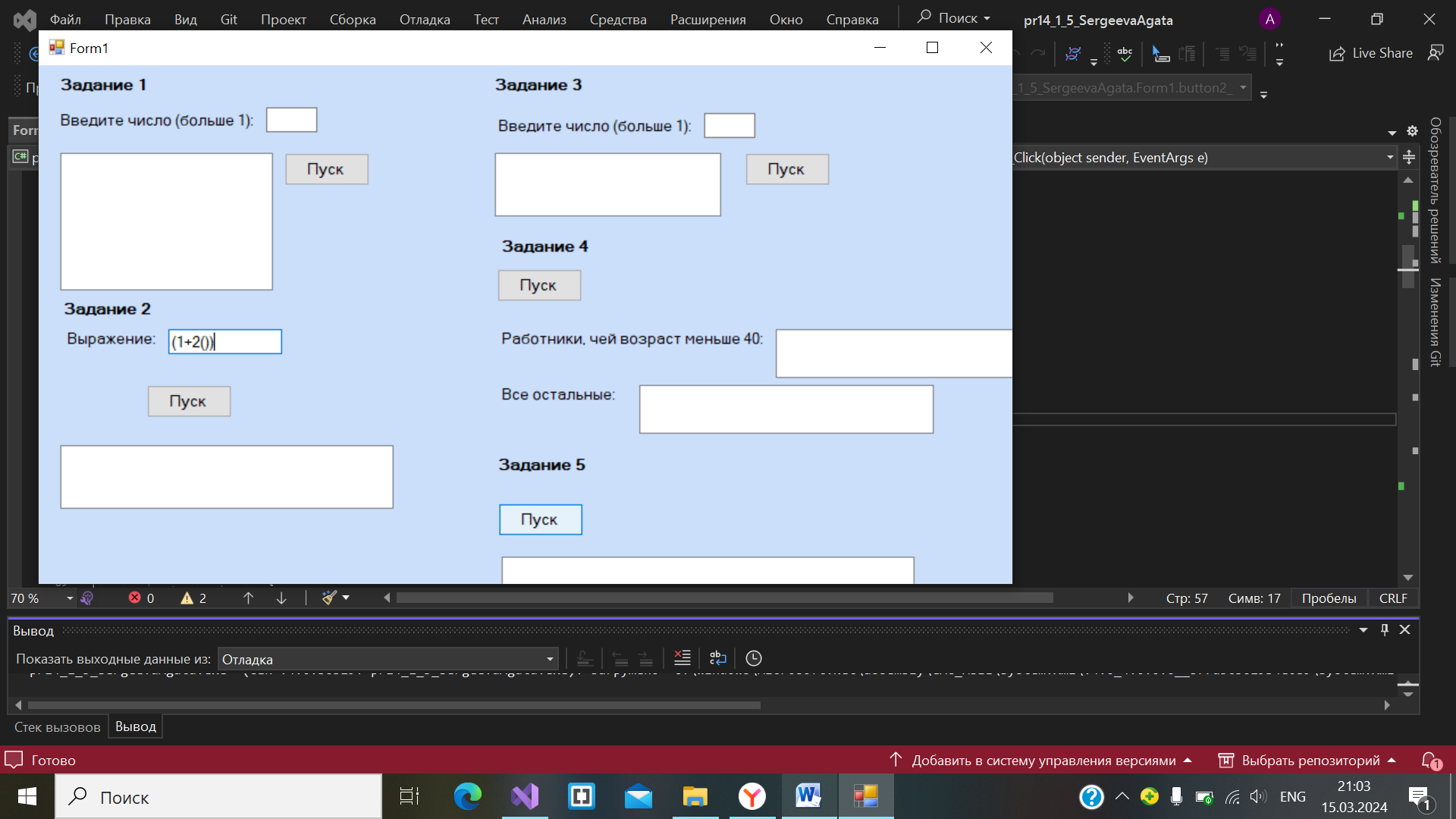
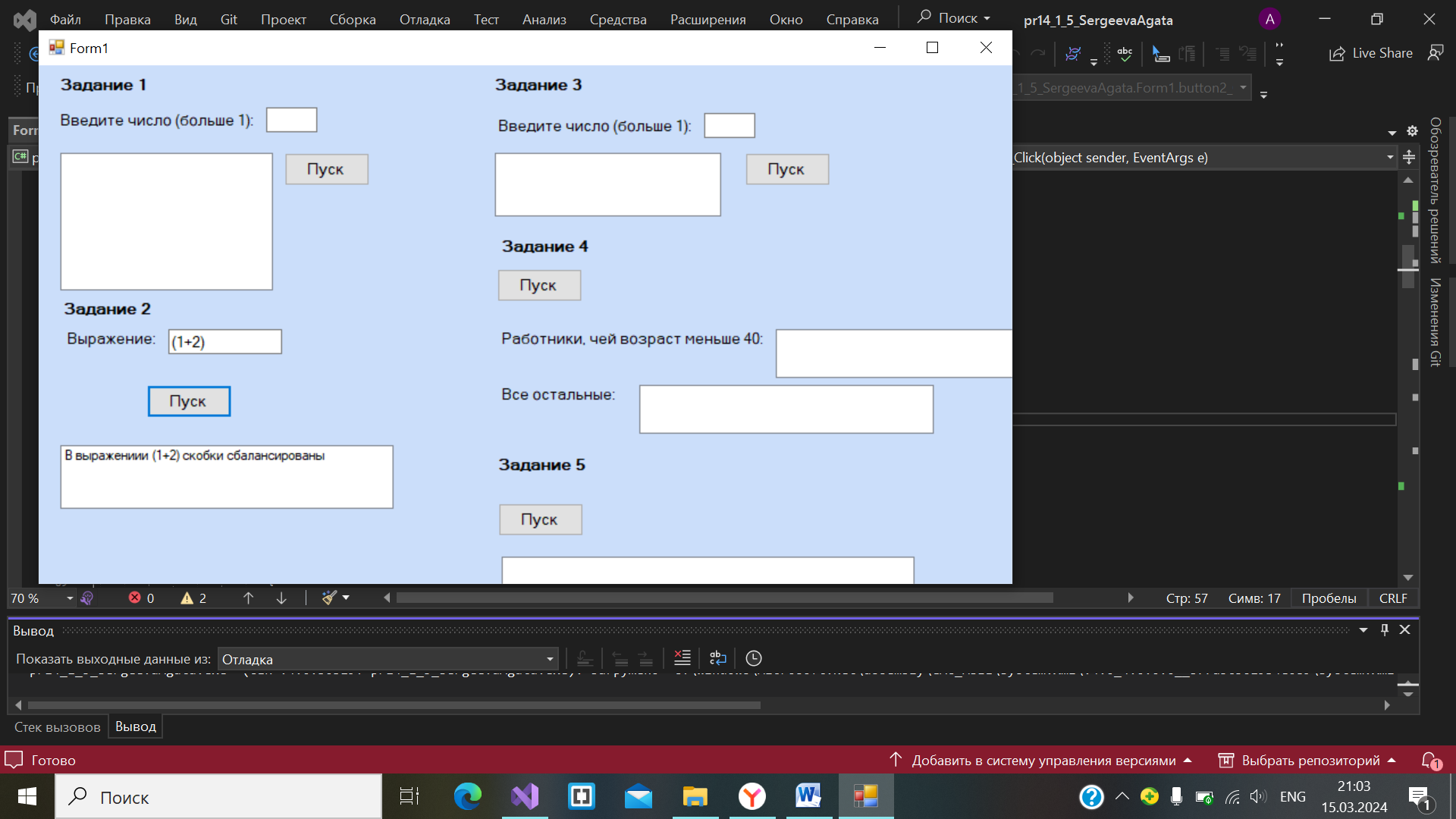
}

else MessageBox.Show("Вы не ввели математическое выражение", "Ошибка", MessageBoxButtons.OK, MessageBoxIcon.Error);

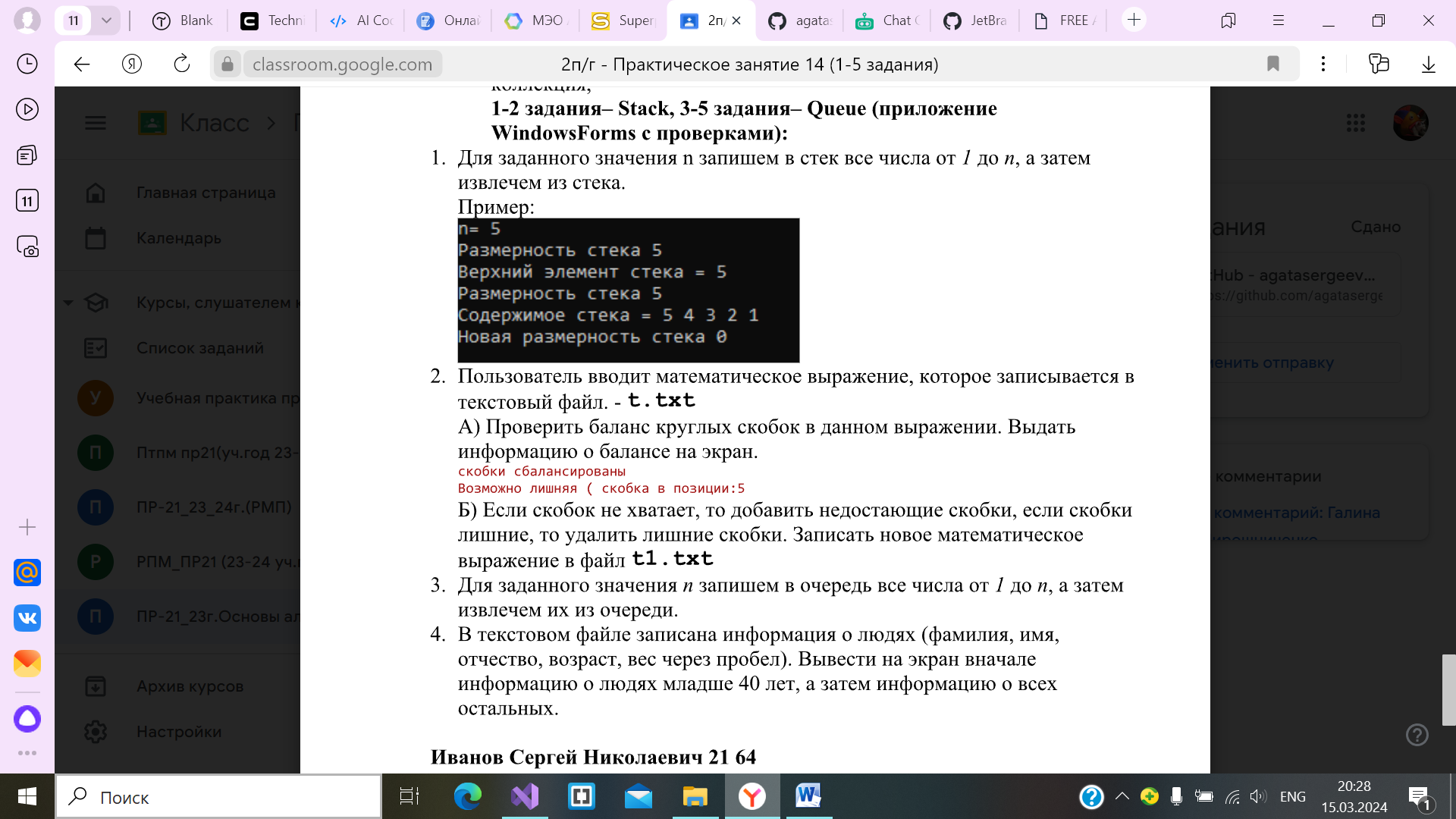
**Тестовые ситуации:**



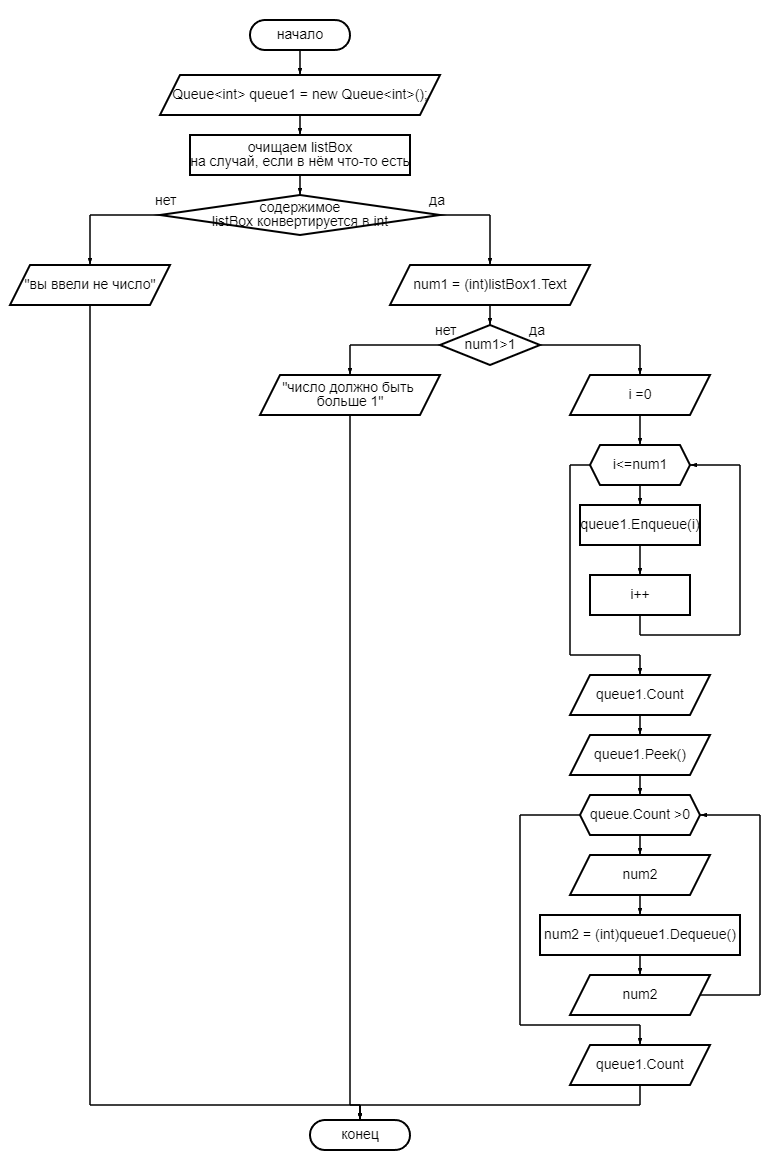
**Задание 3:**



**Входные данные:** n - число, int.

**Выходные данные:** верхний элемент очереди, int; размерности в до и после извлечения, int; вывод содержимого, string.

**Блок-схема:**



**Листинг программы:**

Queue<int> queue1 = new Queue<int>();

listBox3.Items.Clear();

if (int.TryParse((textBox3.Text), out int a))

{

int num1 = Convert.ToInt32(textBox3.Text);

if (num1 > 1)

{

for (int i = 1; i <= num1; i++)

queue1.Enqueue(i);

listBox3.Items.Add($"Размерность очереди: {queue1.Count}");

listBox3.Items.Add($"Верхний элемент очереди: {queue1.Peek()}");

while (queue1.Count > 0)

{

int num2 = (int)queue1.Dequeue();

listBox3.Items.Add(num2);

}

listBox3.Items.Add($"Новая размерность очереди: {queue1.Count}");

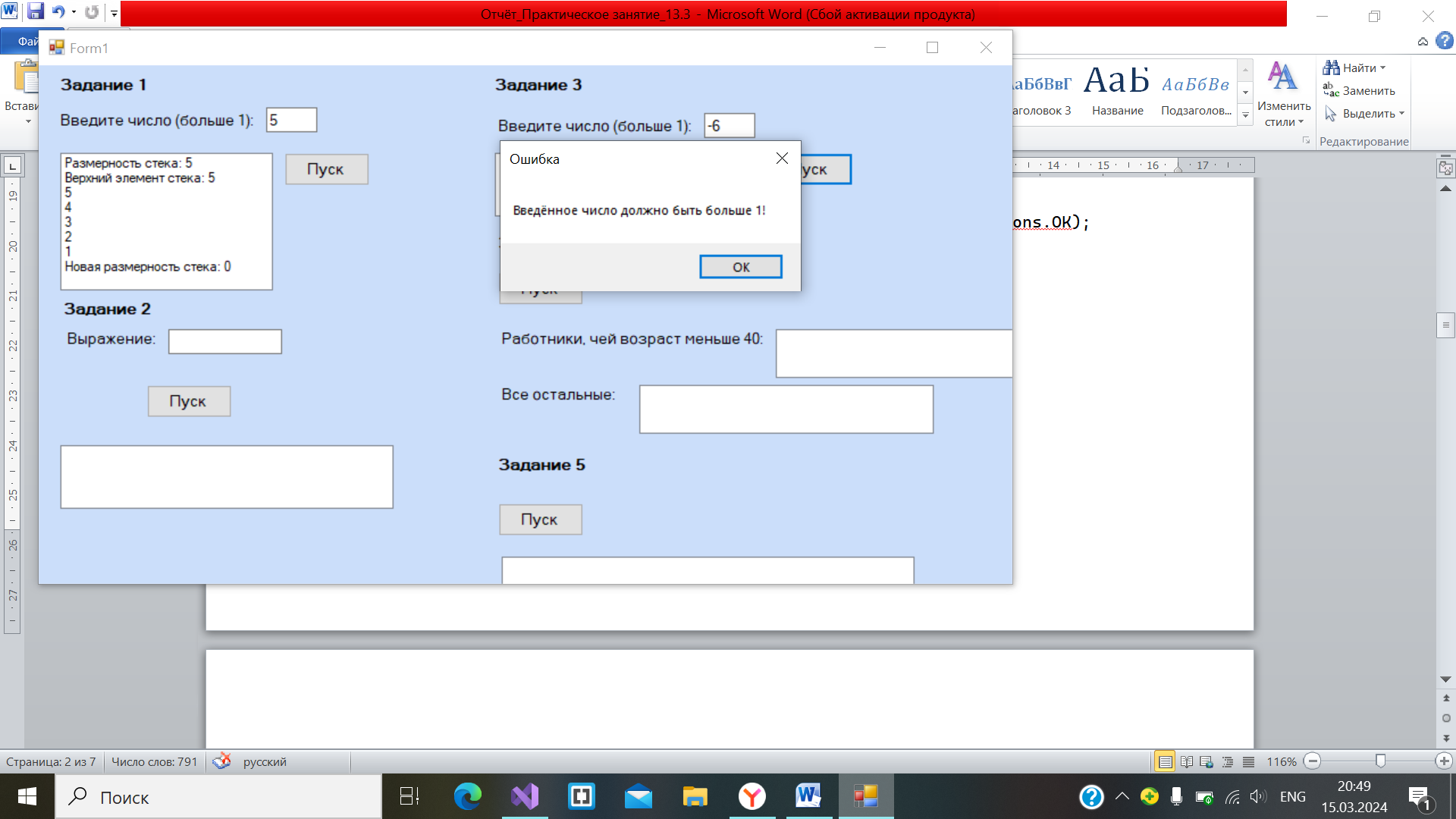
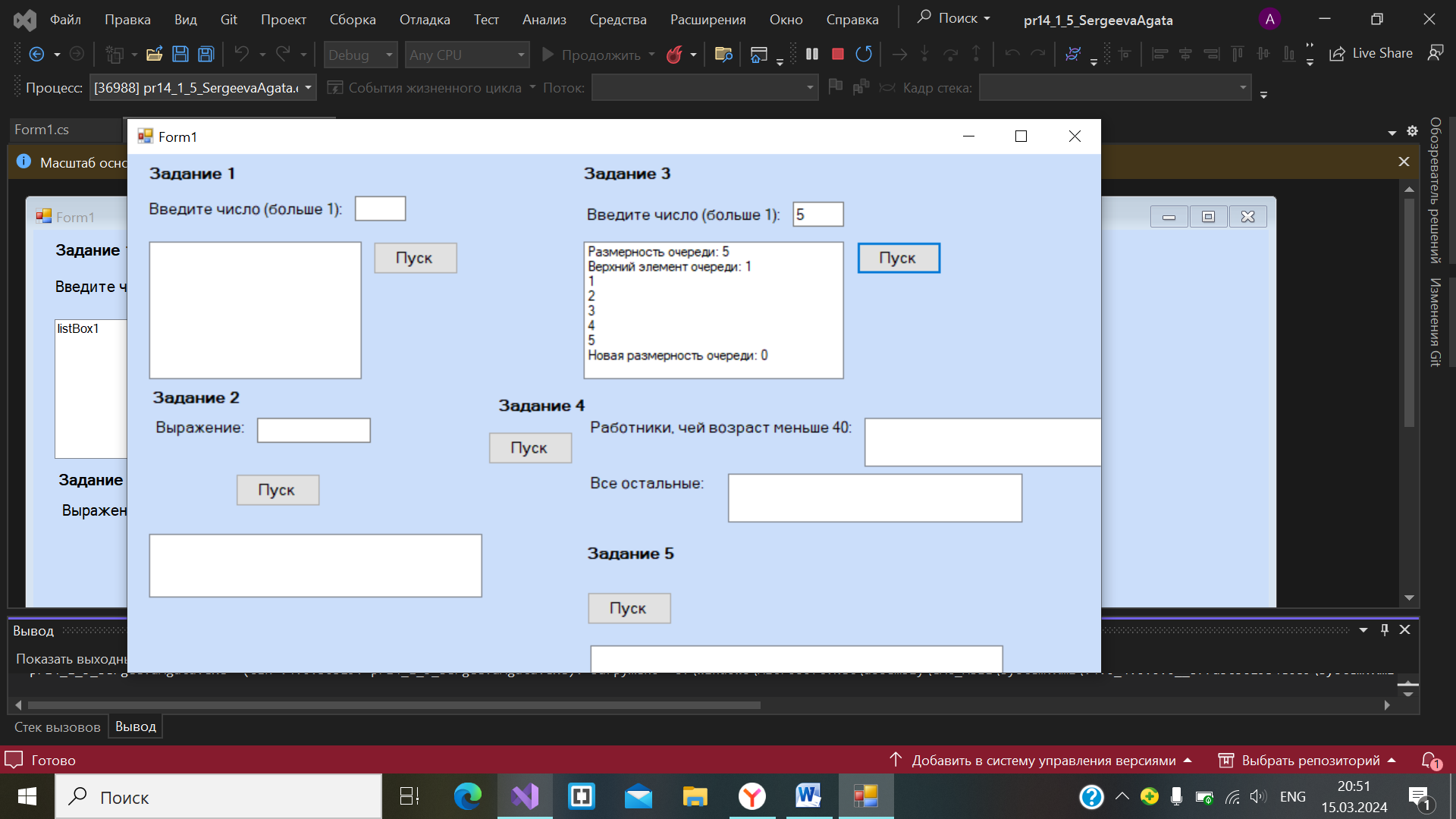
}

else MessageBox.Show("Введённое число должно быть больше 1!", "Ошибка", MessageBoxButtons.OK);

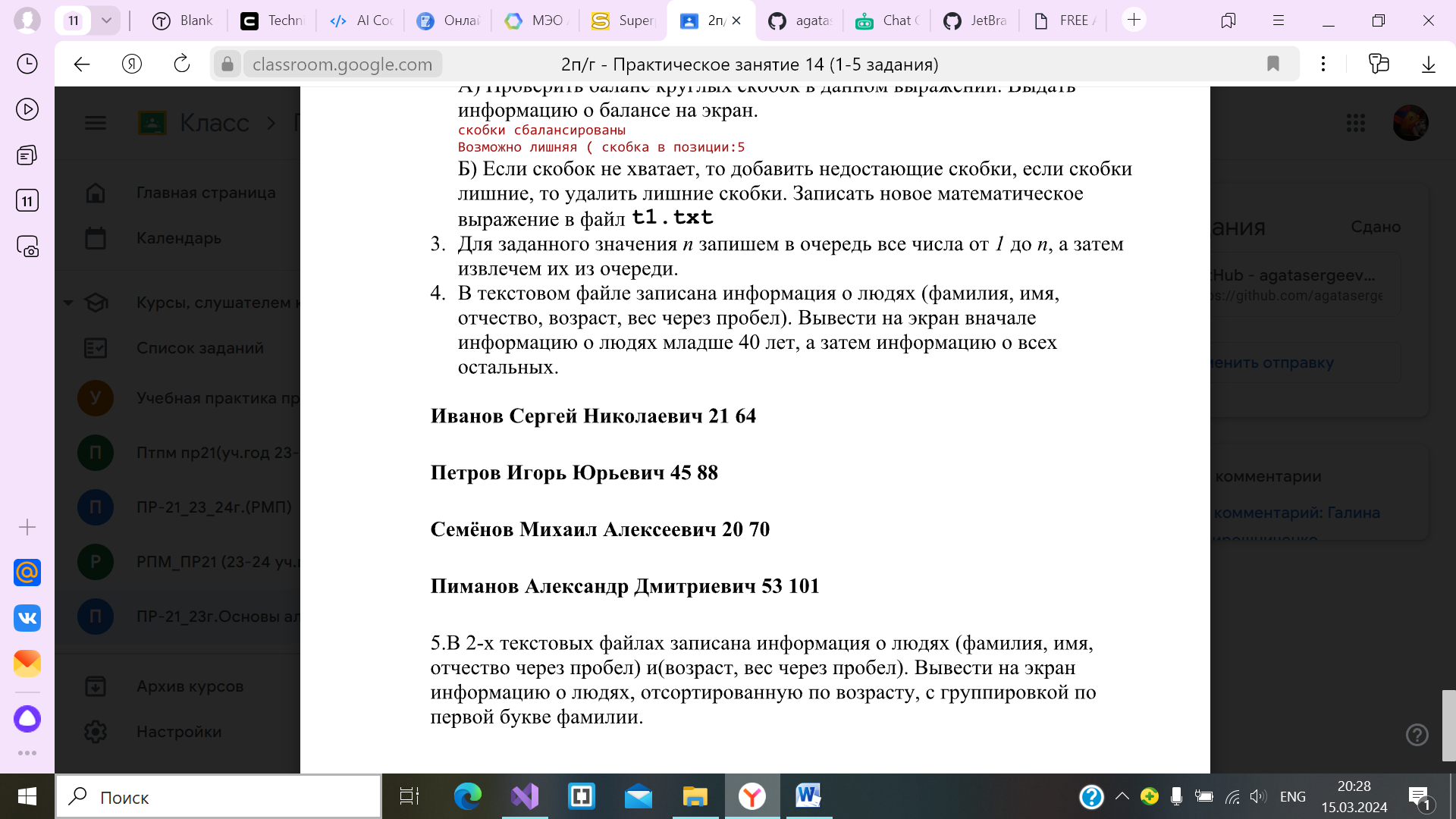
}

else MessageBox.Show("Вы ввели не число!", "Ошибка", MessageBoxButtons.OK);

**Тестовые ситуации:**

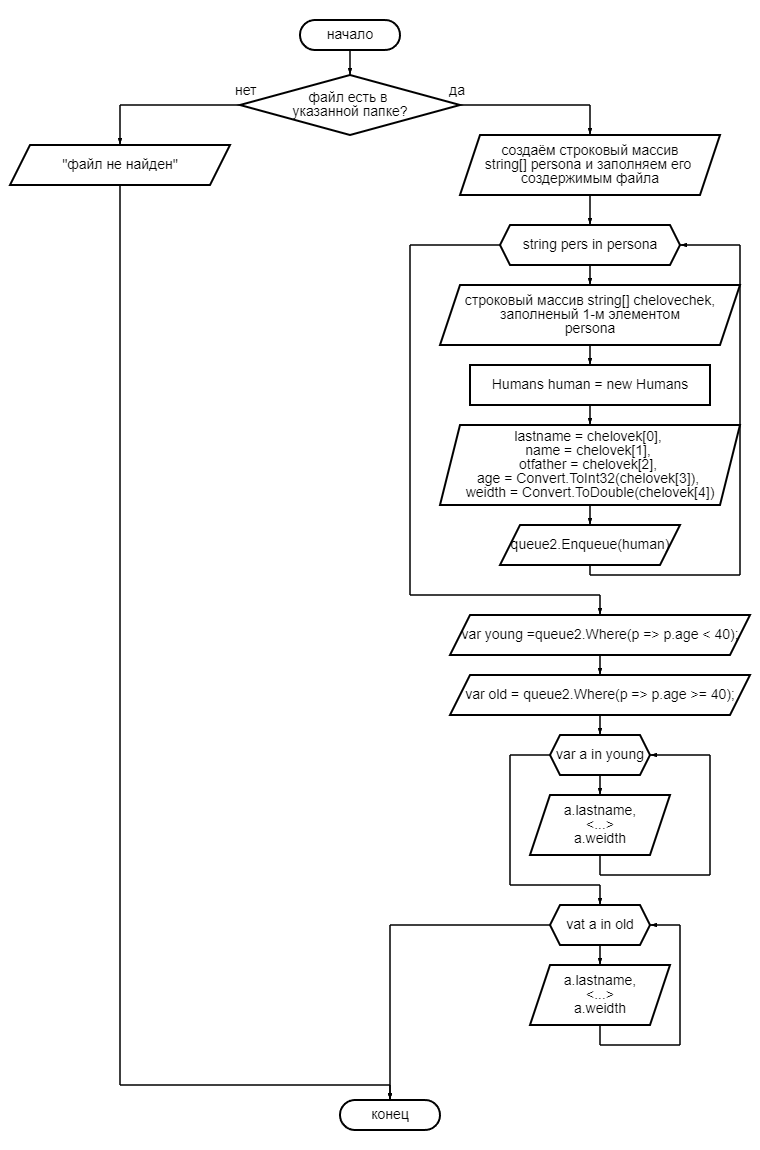
**Задание 4:**



**Входные данные:** данные из файла, string.

**Выходные данные:** информация о людях до и после 40 лет, string.

**Блок-схема:**



**Листинг программы:**

Queue <Humans> queue2 = new Queue<Humans>();

if (!File.Exists("humans.txt"))

{

MessageBox.Show("Файл не найден", "Ошибка", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

else

{

string[] persona = File.ReadAllLines("humans.txt");

foreach (string pers in persona)

{

string[] chelovek = pers.Split(new char[] { ' ' }, StringSplitOptions.RemoveEmptyEntries);

Humans human = new Humans

{

lastname = chelovek[0],

name = chelovek[1],

otfather = chelovek[2],

age = Convert.ToInt32(chelovek[3]),

weidth = Convert.ToDouble(chelovek[4])

};

queue2.Enqueue(human);

}

var young =queue2.Where(p => p.age < 40);

foreach (var a in young)

{

listBox4.Items.Add($"{a.lastname} {a.name} {a.otfather} {a.age} {a.weidth}");

}

var old = queue2.Where(p => p.age >= 40);

foreach (var a in old)

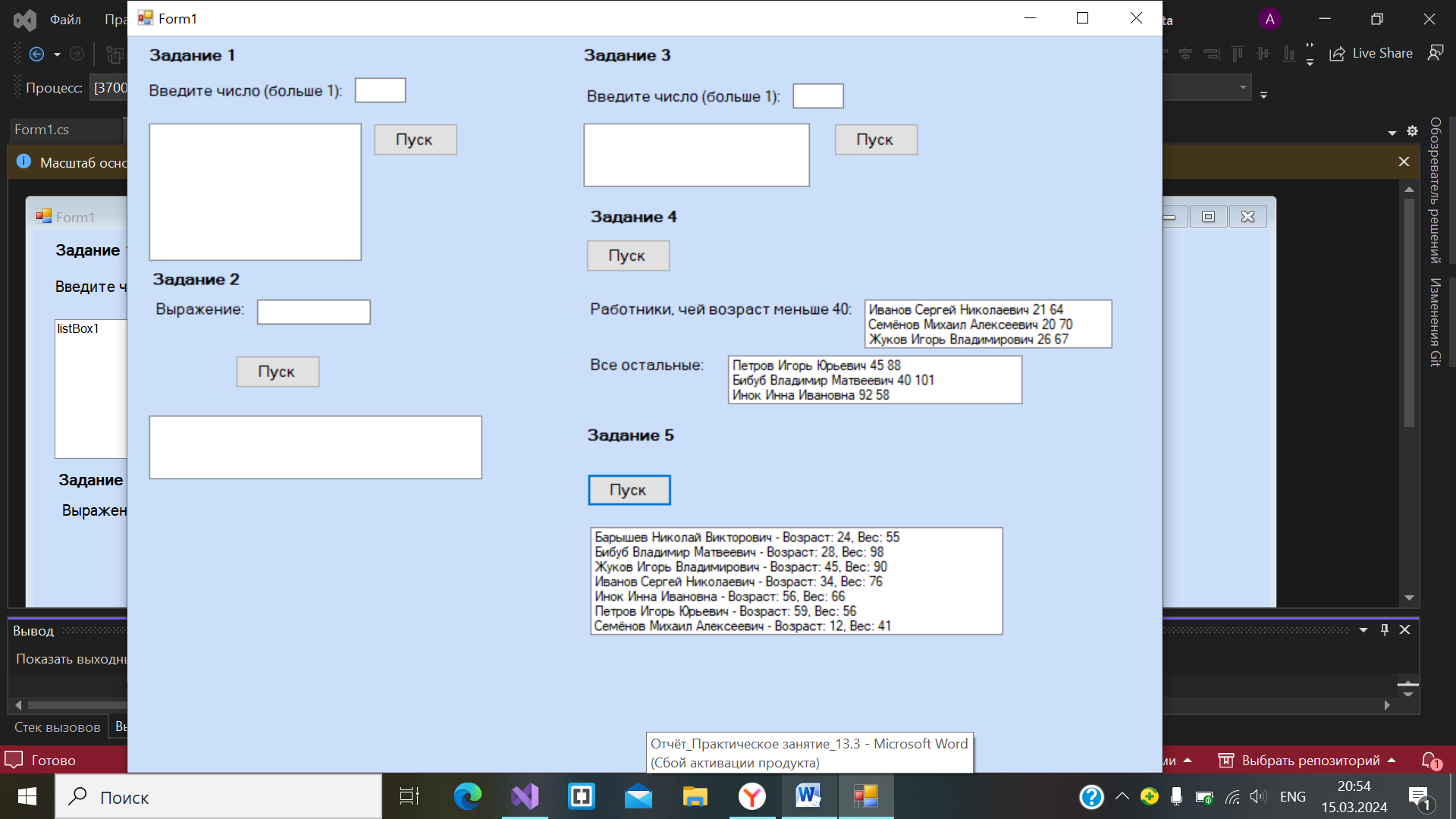
{

listBox5.Items.Add($"{a.lastname} {a.name} {a.otfather} {a.age} {a.weidth}");

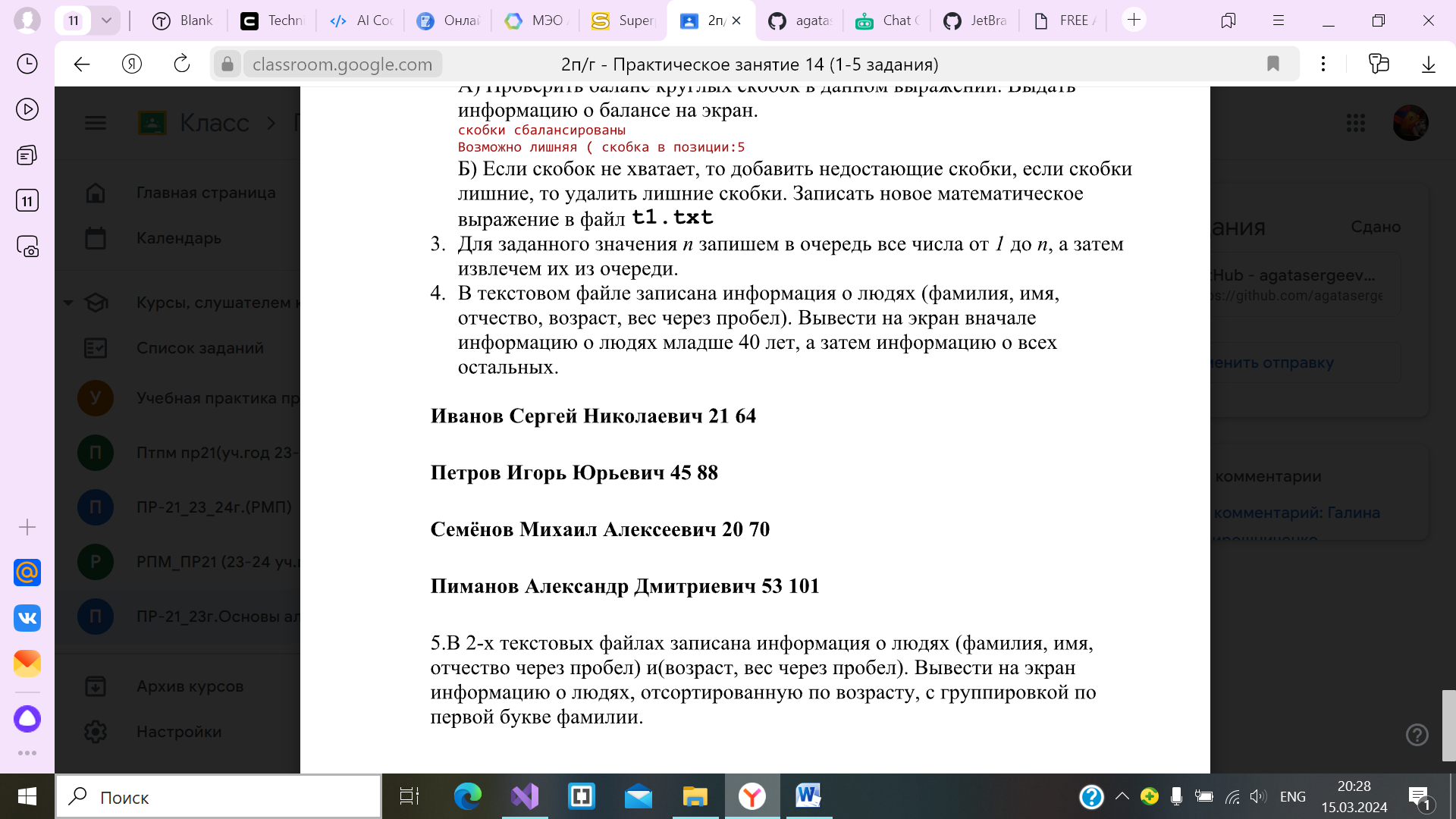
}

}

**Тестовые ситуации:**



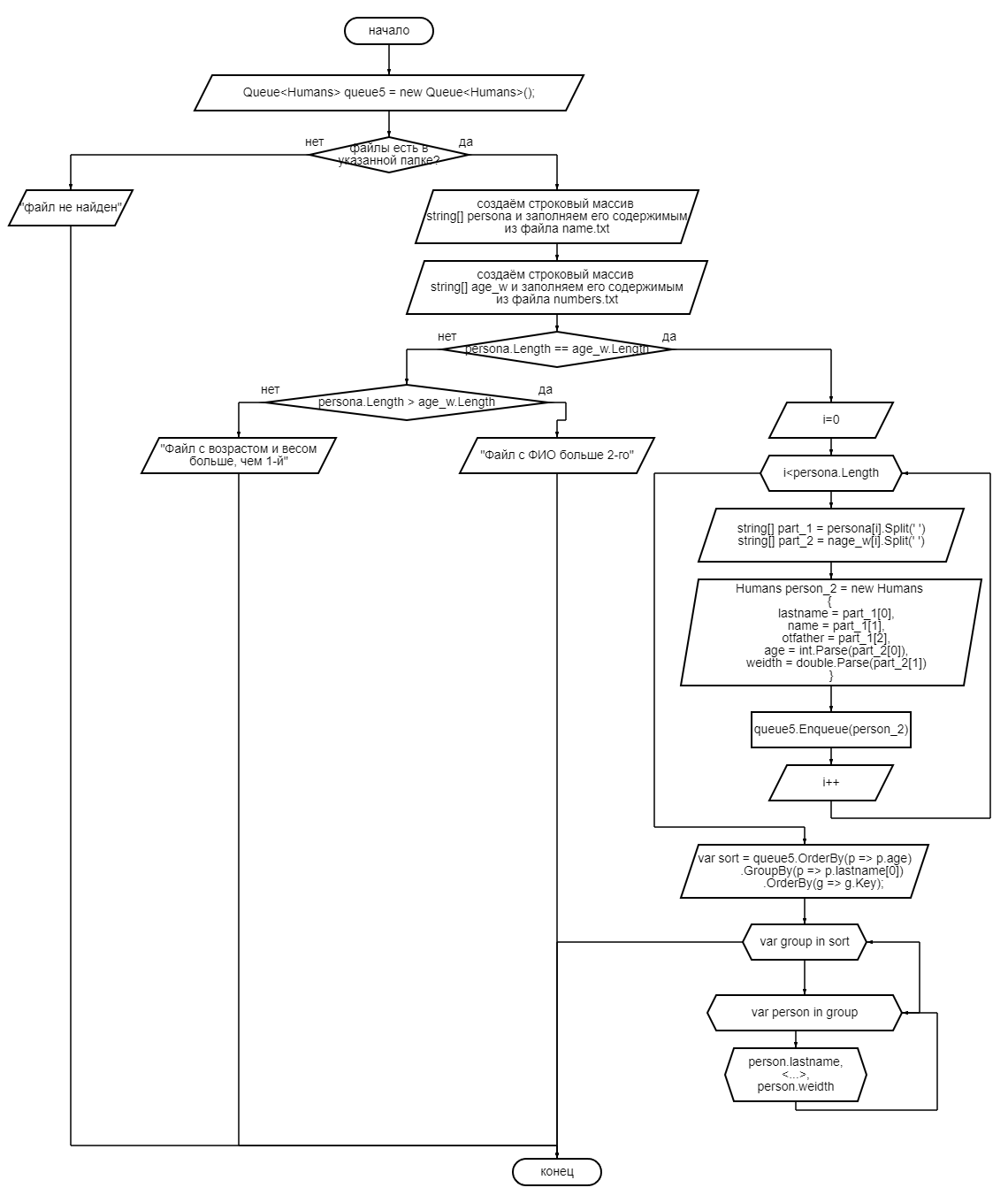
**Задание 5:**



**Входные данные:** данные из 2-х файлов, string

**Выходные данные:** отсортированные данные с группировкой по фамилии, string.

**Блок-схема:**



**Листинг программы:**

Queue<Humans> queue5 = new Queue<Humans>();

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

{

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

{

string[] persona = File.ReadAllLines("name.txt");

string[] age\_w = File.ReadAllLines("numbers.txt");

if (persona.Length == age\_w.Length)

{

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

{

string[] part\_1 = persona[i].Split(new char[] { ' ' }, StringSplitOptions.RemoveEmptyEntries);

string[] part\_2 = age\_w[i].Split(new char[] { ' ' }, StringSplitOptions.RemoveEmptyEntries);

Humans person\_2 = new Humans

{

lastname = part\_1[0],

name = part\_1[1],

otfather = part\_1[2],

age = int.Parse(part\_2[0]),

weidth = double.Parse(part\_2[1])

};

queue5.Enqueue(person\_2);

}

var sort = queue5.OrderBy(p => p.age)

.GroupBy(p => p.lastname[0])

.OrderBy(g => g.Key);

foreach (var group in sort)

{

foreach (var person in group)

{

listBox6.Items.Add($"{person.lastname} {person.name} {person.otfather} - Возраст: {person.age}, Вес: {person.weidth}");

}

}

}

else if (persona.Length > age\_w.Length) MessageBox.Show("СДлина файлов не одинакова! Файл с ФИО сотрудников больше", "Ошибка", MessageBoxButtons.OK, MessageBoxIcon.Error);

else MessageBox.Show("СДлина файлов не одинакова! Файл с возрастом и весом больше!", "Ошибка", MessageBoxButtons.OK, MessageBoxIcon.Error);

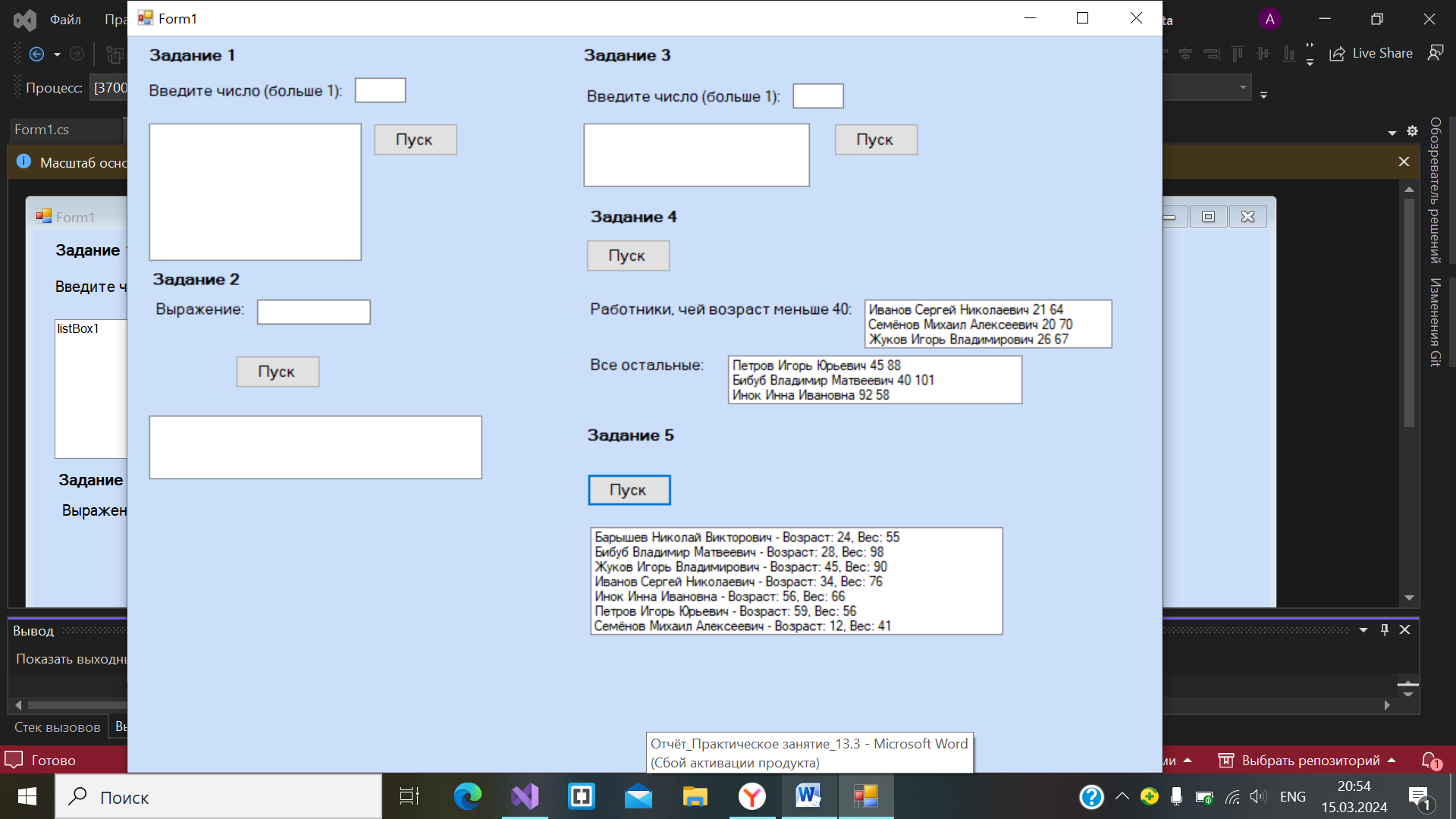
}

else MessageBox.Show("Файл 'age.txt' не найден", "Ошибка", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

else MessageBox.Show("Файл 'name.txt' не найден", "Ошибка", MessageBoxButtons.OK, MessageBoxIcon.Error);

**Тестовые ситуации:**



**Вывод:** научились работать с очередью Queue и стеком Stack в WindowsForms