МИНОБРНАУКИ РФ

ФГБОУ ВО Тверской государственный технический университет

Кафедра “Программное обеспечение”

Отчет по учебной практике

Задание №3

Вариант№21

Выполнил:

Федотов Ярослав Всеволодович

Тверь, 2020

Пункт №1

Код программы С# решение задачи:

using System;

namespace PrackticumCL

{

public class Task3

{

/// <summary>

/// Число результата задачи - свойство.

/// </summary>

public string ResultTaskFactors { get; set; }

/// <summary>

/// Стандартный конструктор класса Task1

/// </summary>

public Task3()

{

}

/// <summary>

/// Констурктор класса Task1

/// </summary>

/// <param name="Odds">На вход подаются коэффициенты для квадратного уравнения.</param>

public Task3(string Odds)

{

EquationFactors(Odds);

}

/// <summary>

/// Констурктор класса Task1

/// </summary>

/// <param name="Odds">На вход подаются коэффициенты для постраения фигур.</param>

/// /// <param name="fig">На вход подается true для выбора перегрузки.</param>

public Task3(string Odds, bool fig)

{

FiguresFactors(Odds, fig);

}

/// <summary>

/// Метод, занимающийся решением квадратного уравнения с коэффициентами поданными на вход.</param>

/// </summary>

/// <param name="Odds">На вход подаются коэффициенты для вычисления корней квадратного уравнения.</param>

private void

EquationFactors(string Odds)

{

string[] Buf = Odds.Split(' ');

double a = double.Parse(Buf[0]);

double b = double.Parse(Buf[1]);

double c = double.Parse(Buf[2]);

ResultTaskFactors = Math.Round((-b + Math.Sqrt(b \* b - 4 \* a \* c)) / (2 \* a), 3).ToString() + " " + Math.Round((-b - Math.Sqrt(b \* b - 4 \* a \* c)) / (2 \* a), 3).ToString();

}

/// <summary>

/// Метод, занимающийся построением фигур.

/// </summary>

/// <param name="Odds">На вход подаются коэффициенты для постраения фигур.</param>

private void

FiguresFactors(string Odds, bool fig)

{

string[] Buf = Odds.Split(' ');

string[] FigursBuilding = new string[int.Parse(Buf[0])];

for (int i = 0; i < int.Parse(Buf[0]); i++)

{

if ((i == 0) | (i == int.Parse(Buf[0]) - 1))

{

for (int j = 0; j < int.Parse(Buf[2]); j++)

{

for (int h = 0; h < int.Parse(Buf[1]); h++)

{

FigursBuilding[i] += "\*";

}

FigursBuilding[i] += " ";

}

}

else

{

int naber = 0;

for (int k = 0; k < (int.Parse(Buf[2]) + int.Parse(Buf[2]) - 1); k++)

{

if (naber == int.Parse(Buf[1]))

{

FigursBuilding[i] += " ";

naber = 0;

}

else

{

for (int g = 0; g < int.Parse(Buf[1]); g++)

{

naber++;

if ((g == 0) | (g == (int.Parse(Buf[1]) - 1)))

{

FigursBuilding[i] += "\*";

}

else

{

FigursBuilding[i] += " ";

}

}

}

}

}

}

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

{

ResultTaskFactors += FigursBuilding[s] + "!";

}

}

}

}

Код программы С# решение задачи windows form:

using System;

using System.Windows.Forms;

using PrackticumCL;

namespace Prakticum

{

public partial class Task1Form : Form

{

Task3 task3 = new Task3();

public Task1Form()

{

InitializeComponent();

}

private void buttonResult\_Click(object sender, EventArgs e)

{

try

{

task3 = new Task3(textBoxInputNumber.Text);

{

textBoxResult.Text = "NOT";

}

else

{

textBoxResult.Text = task3.ResultTaskFactors.ToString();

}

}

catch (Exception)

{

textBoxResult.Text = "incorrect number";

}

}

private void Task1Form\_Load(object sender, EventArgs e)

{

}

private void buttonResultFig\_Click(object sender, EventArgs e)

{

task3 = new Task3(textBoxInputNumber.Text, true);

string[] mas = task3.ResultTaskFactors.Split('!');

for (int i = 0; i < mas.Length - 1; i++)

{

textBoxResult.Text += mas[i] + Environment.NewLine;

}

}

}

}

Псевдоекод:

Используем system;

пространство имён prackticumcl

{

Публичный класс task3

{

/// <summary>

/// число результата задачи - свойство.

/// </summary>

Публичный Строка resulttaskfactors { get; set; }

/// <summary>

/// стандартный конструктор класса task1

/// </summary>

Публичный task3()

{

}

/// <summary>

/// констурктор класса task1

/// </summary>

/// <param name="odds">на вход подаются коэффициенты для квадратного уравнения.</param>

Публичный task3(string odds)

{

equationfactors(odds);

}

/// <summary>

/// констурктор класса task1

/// </summary>

/// <param name="odds">на вход подаются коэффициенты для постраения фигур.</param>

/// /// <param name="fig">на вход подается истина для выбора перегрузки.</param>

Публичный task3(string odds, Булевская переменная fig)

{

figuresfactors(odds, fig);

}

/// <summary>

/// метод, занимающийся решением квадратного уравнения с коэффициентами поданными на вход.</param>

/// </summary>

/// <param name="odds">на вход подаются коэффициенты для вычисления корней квадратного уравнения.</param>

Приватный Функция

equationfactors(string odds)

{

Строковый массив buf = odds.split(' ');

Вещественное число a = double.parse(buf[0]);

Вещественное число b = double.parse(buf[1]);

Вещественное число c = double.parse(buf[2]);

resulttaskfactors = math.round((-b + math.sqrt(b \* b - 4 \* a \* c)) / (2 \* a), 3).tostring() + " " + math.round((-b - math.sqrt(b \* b - 4 \* a \* c)) / (2 \* a), 3).tostring();

}

/// <summary>

/// метод, занимающийся построением фигур.

/// </summary>

/// <param name="odds">на вход подаются коэффициенты для постраения фигур.</param>

Приватный Функция

figuresfactors(string odds, Булевская переменная fig)

{

Строковый массив buf = odds.split(' ');

Строковый массив figursbuilding = новый string[int.parse(buf[0])];

Для (int i = 0; i < int.parse(buf[0]); i++)

{

Если ((i == 0) | (i == int.parse(buf[0]) - 1))

{

Для (int j = 0; j < int.parse(buf[2]); j++)

{

Для (int h = 0; h < int.parse(buf[1]); h++)

{

figursbuilding[i] += "\*";

}

figursbuilding[i] += " ";

}

}

Иначе

{

Целое число naber = 0;

Для (int k = 0; k < (int.parse(buf[2]) + int.parse(buf[2]) - 1); k++)

{

Если (naber == int.parse(buf[1]))

{

figursbuilding[i] += " ";

naber = 0;

}

Иначе

{

Для (int g = 0; g < int.parse(buf[1]); g++)

{

naber++;

Если ((g == 0) | (g == (int.parse(buf[1]) - 1)))

{

figursbuilding[i] += "\*";

}

Иначе

{

figursbuilding[i] += " ";

}

}

}

}

}

}

Для (int s = 0; s < figursbuilding.length; s++)

{

resulttaskfactors += figursbuilding[s] + "!";

}

}

}

}

Пункт №3

Тестирование приложения















ПРИМЕЧАНИЕ:

Из-за того что пробел заниимает меньше места чем "\*" (разница указана красным) вывести ровно не представляется возможным.



В случае замены пробела на "\*" всё ровно

