МИНОБРНАУКИ РФ

ФГБОУ ВО Тверской государственный технический университет

Кафедра “Программное обеспечение”

Отчет по учебной практике

Задание №4

Вариант№5F

Выполнил:

Федотов Ярослав Всеволодович

Тверь, 2020

Пункт №1

Код программы С# решение задачи:

using System;

namespace PrackticumCL

{

public class Task4

{

/// <summary>

/// Число результата задачи - свойство.

/// </summary>

public string ResultTaskFactors { get; set; }

/// <summary>

/// Стандартный конструктор класса Task4

/// </summary>

public Task4()

{

}

/// <summary>

/// Констурктор класса Task4

/// </summary>

/// <param name="S1data">На вход подаётся первая строка исходного файла.</param>

/// <param name="S2data">На вход подаётся вторая строка исходного файла.</param>

public Task4(string[] S1data, string[] S2data)

{

CalculationQuantityDaysFactors(S1data, S2data);

}

/// <summary>

/// Метод, занимающийся вычислением количества дней оставшихся до дня рождения с учетом високстных годов.

/// </summary>

/// <param name="S1data">На вход подаётся первая строка исходного файла для расчёта количества дней до дня рождения.</param>

/// <param name="S2data">На вход подаётся вторая строка исходного файла для расчёта количества дней до дня рождения.</param>

private void

CalculationQuantityDaysFactors(string[] S1data, string[] S2data)

{

bool BreakCheck = false;

int AddYears = 0;

int DaysCounterBirthday = 0;

int DaysCounterNewTime = 0;

int[] ArrMonths = new int[12] {31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31};

if (S1data[0][0] == '0')

{

S1data[0] = S1data[0][1].ToString();

}

if (S1data[1][0] == '0')

{

S1data[1] = S1data[1][1].ToString();

}

if (S2data[0][0] == '0')

{

S2data[0] = S2data[0][1].ToString();

}

if (S2data[1][0] == '0')

{

S2data[1] = S2data[1][1].ToString();

}

int Bdays = int.Parse(S1data[0]);

int Bmonths = int.Parse(S1data[1]);

int days = int.Parse(S2data[0]);

int months = int.Parse(S2data[1]);

int years = int.Parse(S2data[2]);

if ((years % 4 == 0 && !(years % 100 == 0)) | years % 400 == 0)

{

ArrMonths[1] = 29;

}

if (2 == Bmonths && 29 == Bdays)

{

for (int g = 0; g < 5; g++)

{

AddYears++;

if (((years + AddYears) % 4 == 0 && !((years + AddYears) % 100 == 0)) | (years + AddYears) % 400 == 0)

{

break;

}

}

for (int q = 0; q < 12; q++)

{

for (int w = 0; w < ArrMonths[q]; w++)

{

DaysCounterNewTime++;

if (q == (months - 1) && w == days)

{

BreakCheck = true;

break;

}

}

if (BreakCheck == true)

{

break;

}

}

ResultTaskFactors = (365 - DaysCounterNewTime + 365 \* AddYears + 1).ToString();

}

else

{

for (int i = 0; i < 12; i++)

{

for (int k = 0; k < ArrMonths[i]; k++)

{

if (i == (Bmonths - 1) && k == Bdays)

{

BreakCheck = true;

break;

}

DaysCounterBirthday++;

}

if (BreakCheck == true)

{

BreakCheck = false;

break;

}

}

for (int j = 0; j < 12; j++)

{

for (int s = 0; s < ArrMonths[j]; s++)

{

if (j == (months - 1) && s == days)

{

BreakCheck = true;

break;

}

DaysCounterNewTime++;

}

if (BreakCheck == true)

{

break;

}

}

if (DaysCounterNewTime <= DaysCounterBirthday)

{

ResultTaskFactors = (DaysCounterBirthday - DaysCounterNewTime).ToString();

}

else

{

ResultTaskFactors = (365 - DaysCounterNewTime + DaysCounterBirthday).ToString();

}

}

}

}

}

Псевдоекод:

Используем system;

пространство имён prackticumcl

{

Публичный класс task4

{

/// <summary>

/// число результата задачи - свойство.

/// </summary>

Публичный Строка resulttaskfactors { get; set; }

/// <summary>

/// стандартный конструктор класса task4

/// </summary>

Публичный task4()

{

}

/// <summary>

/// констурктор класса task4

/// </summary>

/// <param name="s1data">на вход подаётся первая строка исходного файла.</param>

/// <param name="s2data">на вход подаётся вторая строка исходного файла.</param>

Публичный task4(string[] s1data, Строковый массив s2data)

{

calculationquantitydaysfactors(s1data, s2data);

}

/// <summary>

/// метод, занимающийся вычислением количества дней оставшихся до дня рождения с учетом високстных годов.

/// </summary>

/// <param name="s1data">на вход подаётся первая строка исходного файла для расчёта количества дней до дня рождения.</param>

/// <param name="s2data">на вход подаётся вторая строка исходного файла для расчёта количества дней до дня рождения.</param>

Приватный Функция

calculationquantitydaysfactors(string[] s1data, Строковый массив s2data)

{

Булевская переменная breakcheck = false;

Целое число addyears = 0;

Целое число dayscounterbirthday = 0;

Целое число dayscounternewtime = 0;

Целочисленный массив arrmonths = новый int[12] {31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31};

Если (s1data[0][0] == '0')

{

s1data[0] = s1data[0][1].tostring();

}

Если (s1data[1][0] == '0')

{

s1data[1] = s1data[1][1].tostring();

}

Если (s2data[0][0] == '0')

{

s2data[0] = s2data[0][1].tostring();

}

Если (s2data[1][0] == '0')

{

s2data[1] = s2data[1][1].tostring();

}

Целое число bdays = int.parse(s1data[0]);

Целое число bmonths = int.parse(s1data[1]);

Целое число days = int.parse(s2data[0]);

Целое число months = int.parse(s2data[1]);

Целое число years = int.parse(s2data[2]);

Если ((years % 4 == 0 && !(years % 100 == 0)) | years % 400 == 0)

{

arrmonths[1] = 29;

}

Если (2 == bmonths && 29 == bdays)

{

Для (int g = 0; g < 5; g++)

{

addyears++;

Если (((years + addyears) % 4 == 0 && !((years + addyears) % 100 == 0)) | (years + addyears) % 400 == 0)

{

break;

}

}

Для (int q = 0; q < 12; q++)

{

Для (int w = 0; w < arrmonths[q]; w++)

{

dayscounternewtime++;

Если (q == (months - 1) && w == days)

{

breakcheck = true;

break;

}

}

Если (breakcheck == true)

{

break;

}

}

resulttaskfactors = (365 - dayscounternewtime + 365 \* addyears + 1).tostring();

}

Иначе

{

Для (int i = 0; i < 12; i++)

{

Для (int k = 0; k < arrmonths[i]; k++)

{

Если (i == (bmonths - 1) && k == bdays)

{

breakcheck = true;

break;

}

dayscounterbirthday++;

}

Если (breakcheck == true)

{

breakcheck = false;

break;

}

}

Для (int j = 0; j < 12; j++)

{

Для (int s = 0; s < arrmonths[j]; s++)

{

Если (j == (months - 1) && s == days)

{

breakcheck = true;

break;

}

dayscounternewtime++;

}

Если (breakcheck == true)

{

break;

}

}

Если (dayscounternewtime <= dayscounterbirthday)

{

resulttaskfactors = (dayscounterbirthday - dayscounternewtime).tostring();

}

Иначе

{

resulttaskfactors = (365 - dayscounternewtime + dayscounterbirthday).tostring();

}

}

}

}

}

Код программы С# решение задачи windows form:

using System;

using System.Windows.Forms;

using PrackticumCL;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.IO;

namespace Prakticum

{

public partial class Task1Form : Form

{

Task4 task4 = new Task4();

public Task1Form()

{

InitializeComponent();

}

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

{

try

{

StreamReader sr = new StreamReader("birthday.in.txt");

String[] S1data = sr.ReadLine().Split(' ');

String[] S2data = sr.ReadLine().Split(' ');

sr.Close();

if (int.Parse(S2data[2]) > 3000 | int.Parse(S2data[2]) < 1920)

{

StreamWriter sw = new StreamWriter("birthday.out.txt");

sw.WriteLine("Неверный год");

sw.Close();

}

else

{

task4 = new Task4(S1data, S2data);

StreamWriter sw = new StreamWriter("birthday.out.txt");

sw.WriteLine(task4.ResultTaskFactors);

sw.Close();

}

}

catch (Exception)

{

StreamWriter sw = new StreamWriter("birthday.out.txt");

sw.WriteLine("Неверный формат файла");

sw.Close();

}

}

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

{

}

}

}

Пункт №3

Тестирование приложения





