ФЕДЕРАЛЬНОЕ АГЕНТСТВО ПО ОБРАЗОВАНИЮ

МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ СТРОИТЕЛЬНЫЙ УНИВЕРСИТЕТ

СПЕЦИАЛЬНОСТЬ 2903 «Промышленное и гражданское строительство»

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к курсовому проекту по системотехнике

Выполнила:

студентка ТЭС VI-3 Худякова Н.Ю.

Руководитель:

проф., д.т.н. Гинзбург А.В.

Москва, 2010

**УСЛОВИЕ ЗАДАЧИ**

На строительстве заняты две машины: экскаватор и бульдозер. Машины периодически ломаются.

Математическое ожидание продолжительности рабочего времени для экскаватора 4 часа, для бульдозера 6 часов. Машины работают подряд в две смены - 16 часов в день. Третья смена используется для профилактики и ремонта.

Рабочий день машины всегда начинают в исправном состоянии. После каждого отказа машины подвергаются ремонту слесарями 3-го и 6-го разряда. Слесарь 3-го разряда может ремонтировать в одиночку только бульдозер, 6-го - любую машину. Они могут работать вдвоем.

Математическое ожидание продолжительности ремонта в зависимости от состава бригады:

Бульдозер Экскаватор

Слесарь\_6 2часа 1час

Слесарь\_3 - 2 часа

Бригада(6+3) 1,5 0,25

Все случайные величины распределены по экспоненциальному закону.

Простой машины (в 1 и 2 смену) приносит следующие убытки.

Экскаватор - 500 руб/час.

Бульдозер - 300 руб/час.

Зарплата слесарей:

Слесарь 6 разряда – 100 руб/час

Слесарь 3 разряда - 60 руб/час

Накладные расходы на бригаду – 50 руб/час на время ремонта.

Вопрос: Выгодно ли уволить слесаря 3-го разряда?

**РЕШЕНИЕ**

Пусть бригада всегда состоит из двух человек, т.е. оба слесаря всегда ремонтируют одну машину и не разделяются, если одновременно неисправны обе машины.

Необходимо сравнить стоимость дня при работе одного слесаря СТдня1 и двух слесарей СТдня2.

Формулы для вычисления стоимости дней:

**СТдня1= (общ. tрем.б + общ. tрем.э)х(ЗПс6+НР)+ (общ. tп.б)хУп.б+ (общ. tп.э)хУп.э**

**СТдня2= (общ. tрем.б + общ. tрем.э)х(ЗПс6+ЗПс3+НР)+ (общ. tп.б)хУп.б+ (общ. tп.э)хУп.э**

где:

общ. tрем.б = t(Мрем.б, x), ч – общее за день время ремонта бульдозера;

общ. tрем.э = t(Мрем.э, x), ч – общее за день время ремонта экскаватора;

общ. tп.б, ч – общее за день время простоя бульдозера;

общ. tрем.э, ч – общее за день время простоя экскаватора;

х – случайная величина в интервале [0, 1], распределенная по равномерному закону;

Мр.б = 6 – математическое ожидание времени работы бульдозера, ч;

Мр.э = 4 – математическое ожидание времени работы экскаватора, ч;

М1рем.б = 2 – математическое ожидание времени ремонта бульдозера одним слесарем, ч;

М1рем.э = 1 – математическое ожидание времени ремонта экскаватора одним слесарем, ч;

М2рем.б = 1,5 – математическое ожидание времени ремонта бульдозера двумя слесарями, ч;

М2рем.э = 0,25 – математическое ожидание времени ремонта экскаватора двумя слесарями, ч;

Уп.б = 300 – убытки от простоя бульдозера, руб.;

Уп.э = 500 – убытки от простоя бульдозера. руб.;

ЗПс6 = 100 – зарплата слесаря 6-го разряда, руб.;

ЗПс3 = 60 – зарплата слесаря 3-го разряда, руб.;

НР = 50 – накладные расходы на бригаду, руб.

Математическое ожидание продолжительности того или иного состояния (работы, ремонта) находится в следующей зависимости от случайной величины х:

t(M,x)=(-1)∙М∙ln(1-x)

х=random(0, 1)

Величины общ. tрем.б, общ. tрем.э, общ. tп.б, общ. tрем.э вычисляются по представленному ниже алгоритму. Программа для решения задачи написана на языке программирования С#.

**АЛГОРИТМ**

**ТЕКСТ ПРОГРАММЫ**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Windows.Forms;

using System.IO;

namespace zad

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

//Расчёт времени

public Random x = new Random();

public double t(double M)

{

return (-1) \* M \* Math.Log(1 - x.NextDouble());

}

//Метод (функция) основного расчёта

public void mainCalc()

{

double ZP1 = Convert.ToDouble(txtZP1.Text);

double ZP2 = Convert.ToDouble(txtZP2.Text);

double NR = Convert.ToDouble(txtNR.Text);

double UBP = Convert.ToDouble(txtUBP.Text);

double UEP = Convert.ToDouble(txtUEP.Text);

double Cost = new double();

double TBR = 0;

double TER = 0;

double TBP = 0;

double TEP = 0;

int RndDay = x.Next(100);

int iter = 0;

while (iter < Convert.ToInt32(txtQit.Text))

{

iter++;

double B = new double();

double E = new double();

double[] BR = new double[0];

double[] BJ = new double[1];

double[] ER = new double[0];

double[] EJ = new double[1];

double[] BP = new double[0];

double[] EP = new double[0];

int[] indexBJ = new int[0];

int[] indexEJ = new int[0];

int lenghtBR;

int lenghtBJ;

int lenghtER;

int lenghtEJ;

int lenghtBP;

int lenghtEP;

int lenghtIBJ;

int lenghtIEJ;

B = t(Convert.ToDouble(txtMathExpBj.Text));

E = t(Convert.ToDouble(txtMathExpEj.Text));

BJ[0] = B;

EJ[0] = E;

switch (1)

{

// CASE 1 ### CASE 1 ### CASE 1 ### CASE 1

case 1:

if ((B > 16) && (E > 16))

{

//############ Конец 1 ###############

Array.Resize(ref BJ, 1);

Array.Resize(ref EJ, 1);

Array.Resize(ref BR, 1);

Array.Resize(ref ER, 1);

Array.Resize(ref BP, 1);

Array.Resize(ref EP, 1);

BJ[0] = 16;

EJ[0] = 16;

BR[0] = 0;

ER[0] = 0;

BP[0] = 0;

EP[0] = 0;

goto case 7;

}

else if ((B < 16) && (E >= 16))

{

lenghtBR = BR.Length;

Array.Resize(ref BR, lenghtBR + 1);

lenghtBR = BR.Length;

BR[lenghtBR - 1] = t(Convert.ToDouble(txtMathExpBr.Text));

B = B + BR[lenghtBR - 1];

lenghtBJ = BJ.Length;

int j = 0;

int k;

k = lenghtBJ;

int l;

l = lenghtBR;

if (B < 16)

{

double[] temp = { Convert.ToDouble(txtMathExpBr.Text), Convert.ToDouble(txtMathExpBj.Text) };

while (B < 16)

{

j = 1 - j;

if (j == 1)

{

k++;

Array.Resize(ref BJ, k);

BJ[k - 1] = t(temp[j]);

B = B + BJ[k - 1];

}

else

{

l++;

Array.Resize(ref BR, l);

BR[l - 1] = t(temp[j]);

B = B + BR[l - 1];

}

}

}

if ((B > 16) && (j == 1))

{

//############### Конец 2 ################

lenghtEJ = EJ.Length;

EJ[lenghtEJ - 1] = EJ[lenghtEJ - 1] - (E - 16);

lenghtBJ = BJ.Length;

BJ[lenghtBJ - 1] = BJ[lenghtBJ - 1] - (B - 16);

goto case 7;

}

else if ((B > 16) && (j == 0))

{

if (B > 24)

{

//############### Конец 3 ##############

lenghtEJ = EJ.Length;

EJ[lenghtEJ - 1] = EJ[lenghtEJ - 1] - (E - 16);

lenghtBR = BR.Length;

BR[lenghtBR - 1] = BR[lenghtBR - 1] - (B - 24);

lenghtER = ER.Length;

Array.Resize(ref ER, lenghtER + 1);

lenghtER = ER.Length;

ER[lenghtER - 1] = 0;

goto case 7;

}

else

{

//############### Конец 4 ##############

lenghtEJ = EJ.Length;

EJ[lenghtEJ - 1] = EJ[lenghtEJ - 1] - (E - 16);

goto case 7;

}

}

}

else

{

if ((B >= 16) && (E < 16))

{

lenghtER = ER.Length;

Array.Resize(ref ER, lenghtER + 1);

lenghtER = ER.Length;

ER[lenghtER - 1] = t(Convert.ToDouble(txtMathExpEr.Text));

E = E + ER[lenghtER - 1];

lenghtEJ = EJ.Length;

int j = 0;

int k;

k = lenghtEJ;

int l;

l = lenghtER;

if (E < 16)

{

double[] temp = { Convert.ToDouble(txtMathExpEr.Text), Convert.ToDouble(txtMathExpEj.Text) };

while (E < 16)

{

j = 1 - j;

if (j == 1)

{

k++;

Array.Resize(ref EJ, k);

EJ[k - 1] = t(temp[j]);

E = E + EJ[k - 1];

}

else

{

l++;

Array.Resize(ref ER, l);

ER[l - 1] = t(temp[j]);

E = E + ER[l - 1];

}

}

}

if ((E > 16) && (j == 1))

{

//############# Конец 5 ##############

lenghtBJ = BJ.Length;

BJ[lenghtBJ - 1] = BJ[lenghtBJ - 1] - (B - 16);

lenghtEJ = EJ.Length;

EJ[lenghtEJ - 1] = EJ[lenghtEJ - 1] - (E - 16);

goto case 7;

}

else if ((E > 16) && (j == 0))

{

if (E > 24)

{

//############# Конец 6 #############

lenghtBJ = BJ.Length;

BJ[lenghtBJ - 1] = BJ[lenghtBJ - 1] - (B - 16);

lenghtER = ER.Length;

ER[lenghtER - 1] = ER[lenghtER - 1] - (E - 24);

goto case 7;

}

else

{

//############# Конец 7 ##############

lenghtBJ = BJ.Length;

BJ[lenghtBJ - 1] = BJ[lenghtBJ - 1] - (B - 16);

goto case 7;

}

}

}

else

{

if (B < E)

{

goto case 2;

}

//############## Конец B<E #############

if (B > E)

{

goto case 3;

}

}

}

break;

// CASE 2 ### CASE 2 ### CASE 2 ### CASE 2

case 2:

if (B < E)

{

lenghtBR = BR.Length;

Array.Resize(ref BR, lenghtBR + 1);

lenghtBR = BR.Length;

BR[lenghtBR - 1] = t(Convert.ToDouble(txtMathExpBr.Text));

B = B + BR[lenghtBR - 1];

if (B < 16)

{

goto case 4;

}

else

{

if (B < 24)

{

lenghtER = ER.Length;

Array.Resize(ref ER, lenghtER + 1);

lenghtER = ER.Length;

ER[lenghtER - 1] = t(Convert.ToDouble(txtMathExpEr.Text));

if (ER[lenghtER - 1] < (24 - B))

{

lenghtEP = EP.Length;

Array.Resize(ref EP, lenghtEP + 1);

lenghtEP = EP.Length;

EP[lenghtEP - 1] = 16 - E;

lenghtEJ = EJ.Length;

lenghtIEJ = indexEJ.Length;

Array.Resize(ref indexEJ, lenghtIEJ + 1);

lenghtIEJ = indexEJ.Length;

indexEJ[lenghtIEJ - 1] = lenghtEJ - 1;

goto case 7;

}

else

{

lenghtEP = EP.Length;

Array.Resize(ref EP, lenghtEP + 1);

lenghtEP = EP.Length;

EP[lenghtEP - 1] = 16 - E;

lenghtEJ = EJ.Length;

lenghtIEJ = indexEJ.Length;

Array.Resize(ref indexEJ, lenghtIEJ + 1);

lenghtIEJ = indexEJ.Length;

indexEJ[lenghtIEJ - 1] = lenghtEJ - 1;

ER[lenghtER - 1] = 24 - B;

goto case 7;

}

}

else

{

lenghtEP = EP.Length;

Array.Resize(ref EP, lenghtEP + 1);

lenghtEP = EP.Length;

EP[lenghtEP - 1] = 16 - E;

lenghtEJ = EJ.Length;

lenghtIEJ = indexEJ.Length;

Array.Resize(ref indexEJ, lenghtIEJ + 1);

lenghtIEJ = indexEJ.Length;

indexEJ[lenghtIEJ - 1] = lenghtEJ - 1;

lenghtBR = BR.Length;

BR[lenghtBR - 1] = BR[lenghtBR - 1] - (B - 24);

goto case 7;

}

}

}

goto case 3;

// CASE 3 ### CASE 3 ### CASE 3 ### CASE 3

case 3:

if (B > E)

{

lenghtER = ER.Length;

Array.Resize(ref ER, lenghtER + 1);

lenghtER = ER.Length;

ER[lenghtER - 1] = t(Convert.ToDouble(txtMathExpEr.Text));

E = E + ER[lenghtER - 1];

if (E < 16)

{

goto case 5;

}

else

{

if (E < 24)

{

lenghtBR = BR.Length;

Array.Resize(ref BR, lenghtBR + 1);

lenghtBR = BR.Length;

BR[lenghtBR - 1] = t(Convert.ToDouble(txtMathExpBr.Text));

if (BR[lenghtBR - 1] < (24 - E))

{

lenghtBP = BP.Length;

Array.Resize(ref BP, lenghtBP + 1);

lenghtBP = BP.Length;

BP[lenghtBP - 1] = 16 - B;

lenghtBJ = BJ.Length;

lenghtIBJ = indexBJ.Length;

Array.Resize(ref indexBJ, lenghtIBJ + 1);

lenghtIBJ = indexBJ.Length;

indexBJ[lenghtIBJ - 1] = lenghtBJ - 1;

goto case 7;

}

else

{

lenghtBP = BP.Length;

Array.Resize(ref BP, lenghtBP + 1);

lenghtBP = BP.Length;

BP[lenghtBP - 1] = 16 - B;

lenghtBJ = BJ.Length;

lenghtIBJ = indexBJ.Length;

Array.Resize(ref indexBJ, lenghtIBJ + 1);

lenghtIBJ = indexBJ.Length;

indexBJ[lenghtIBJ - 1] = lenghtBJ - 1;

BR[lenghtBR - 1] = 24 - E;

goto case 7;

}

}

else

{

lenghtBP = BP.Length;

Array.Resize(ref BP, lenghtBP + 1);

lenghtBP = BP.Length;

BP[lenghtBP - 1] = 16 - B;

lenghtBJ = BJ.Length;

lenghtIBJ = indexBJ.Length;

Array.Resize(ref indexBJ, lenghtIBJ + 1);

lenghtIBJ = indexBJ.Length;

indexBJ[lenghtIBJ - 1] = lenghtBJ - 1;

lenghtER = ER.Length;

ER[lenghtER - 1] = ER[lenghtER - 1] - (E - 24);

goto case 7;

}

}

}

goto case 7;

// CASE 4 ### CASE 4 ### CASE 4 ### CASE 4

case 4:

if (B < E)

{

lenghtBJ = BJ.Length;

Array.Resize(ref BJ, lenghtBJ + 1);

lenghtBJ = BJ.Length;

BJ[lenghtBJ - 1] = t(Convert.ToDouble(txtMathExpBj.Text));

B = B + BJ[lenghtBJ - 1];

if (B > 16)

{

lenghtER = ER.Length;

Array.Resize(ref ER, lenghtER + 1);

lenghtER = ER.Length;

ER[lenghtER - 1] = t(Convert.ToDouble(txtMathExpEr.Text));

E = E + ER[lenghtER - 1];

int j;

j = 0;

lenghtEJ = EJ.Length;

int k = lenghtEJ;

int l = lenghtER;

if (E < 16)

{

double[] temp2 = { Convert.ToDouble(txtMathExpEr.Text), Convert.ToDouble(txtMathExpEj.Text) };

while (E < 16)

{

j = 1 - j;

if (j == 1)

{

k++;

Array.Resize(ref EJ, k);

EJ[k - 1] = t(temp2[j]);

E = E + EJ[k - 1];

}

else

{

l++;

Array.Resize(ref ER, l);

ER[l - 1] = t(temp2[j]);

E = E + ER[l - 1];

}

}

}

if ((E > 16) && (j == 1))

{

//############### Конец 8 ################

lenghtEJ = EJ.Length;

EJ[lenghtEJ - 1] = EJ[lenghtEJ - 1] - (E - 16);

lenghtBJ = BJ.Length;

BJ[lenghtBJ - 1] = BJ[lenghtBJ - 1] - (B - 16);

goto case 7;

}

else if ((E > 16) && (j == 0))

{

if (E > 24)

{

//################### Конец 9 #############

lenghtBJ = BJ.Length;

BJ[lenghtBJ - 1] = BJ[lenghtBJ - 1] - (B - 16);

lenghtER = ER.Length;

ER[lenghtER - 1] = ER[lenghtER - 1] - (E - 24);

goto case 7;

}

else

{

//################## Конец 10 ##############

lenghtBJ = BJ.Length;

BJ[lenghtBJ - 1] = BJ[lenghtBJ - 1] - (B - 16);

goto case 7;

}

}

}

else

{

goto case 6;

}

}

else

{

lenghtEP = EP.Length;

Array.Resize(ref EP, lenghtEP + 1);

lenghtEP = EP.Length;

EP[lenghtEP - 1] = B - E;

lenghtEJ = EJ.Length;

lenghtIEJ = indexEJ.Length;

Array.Resize(ref indexEJ, lenghtIEJ + 1);

lenghtIEJ = indexEJ.Length;

indexEJ[lenghtIEJ - 1] = lenghtEJ - 1;

lenghtBJ = BJ.Length;

Array.Resize(ref BJ, lenghtBJ + 1);

lenghtBJ = BJ.Length;

BJ[lenghtBJ - 1] = t(Convert.ToDouble(txtMathExpBj.Text));

B = B + BJ[lenghtBJ - 1];

lenghtER = ER.Length;

Array.Resize(ref ER, lenghtER + 1);

lenghtER = ER.Length;

ER[lenghtER - 1] = t(Convert.ToDouble(txtMathExpEr.Text));

E = E + EP[lenghtEP - 1] + ER[lenghtER - 1];

if ((B > 16) && (E > 16))

{

if (E < 24)

{

//############## Конец 11 ###############

lenghtBJ = BJ.Length;

BJ[lenghtBJ - 1] = BJ[lenghtBJ - 1] - (B - 16);

goto case 7;

}

else

{

//############## Конец 12 ###############

lenghtBJ = BJ.Length;

BJ[lenghtBJ - 1] = BJ[lenghtBJ - 1] - (B - 16);

lenghtER = ER.Length;

ER[lenghtER - 1] = ER[lenghtER - 1] - (E - 24);

goto case 7;

}

}

else

{

if ((B > 16) && (E < 16))

{

double[] temp2 = { Convert.ToDouble(txtMathExpEr.Text), Convert.ToDouble(txtMathExpEj.Text) };

lenghtEJ = EJ.Length;

int k;

k = lenghtEJ;

lenghtER = ER.Length;

int l;

l = lenghtER;

int j = 0;

while (E < 16)

{

j = 1 - j;

if (j == 1)

{

k++;

Array.Resize(ref EJ, k);

EJ[k - 1] = t(temp2[j]);

E = E + EJ[k - 1];

}

else

{

l++;

Array.Resize(ref ER, l);

ER[l - 1] = t(temp2[j]);

E = E + ER[l - 1];

}

}

if ((E > 16) && (j == 0))

{

if (E < 24)

{

//############## Конец 13 ###############

lenghtBJ = BJ.Length;

BJ[lenghtBJ - 1] = BJ[lenghtBJ - 1] - (B - 16);

goto case 7;

}

else

{

//############## Конец 14 ###############

lenghtBJ = BJ.Length;

BJ[lenghtBJ - 1] = BJ[lenghtBJ - 1] - (B - 16);

lenghtER = ER.Length;

ER[lenghtER - 1] = ER[lenghtER - 1] - (E - 24);

goto case 7;

}

}

else if ((E > 16) && (j == 1))

{

//############## Конец 15 ###############

lenghtBJ = BJ.Length;

BJ[lenghtBJ - 1] = BJ[lenghtBJ - 1] - (B - 16);

lenghtEJ = EJ.Length;

EJ[lenghtEJ - 1] = EJ[lenghtEJ - 1] - (E - 16);

goto case 7;

}

}

else

{

if ((B < 16) && (E >= 16))

{

if (E < 24)

{

lenghtBR = BR.Length;

Array.Resize(ref BR, lenghtBR + 1);

lenghtBR = BR.Length;

BR[lenghtBR - 1] = t(Convert.ToDouble(txtMathExpBr.Text));

if (BR[lenghtBR - 1] < (24 - E))

{

//############## Конец 16 ###############

lenghtBP = BP.Length;

Array.Resize(ref BP, lenghtBP + 1);

lenghtBP = BP.Length;

BP[lenghtBP - 1] = 16 - B;

lenghtBJ = BJ.Length;

lenghtIBJ = indexBJ.Length;

Array.Resize(ref indexBJ, lenghtIBJ + 1);

lenghtIBJ = indexBJ.Length;

indexBJ[lenghtIBJ - 1] = lenghtBJ - 1;

goto case 7;

}

else

{

//############## Конец 17 ###############

lenghtBR = BR.Length;

BR[lenghtBR - 1] = 24 - E;

lenghtBP = BP.Length;

Array.Resize(ref BP, lenghtBP + 1);

lenghtBP = BP.Length;

BP[lenghtBP - 1] = 16 - B;

lenghtBJ = BJ.Length;

lenghtIBJ = indexBJ.Length;

Array.Resize(ref indexBJ, lenghtIBJ + 1);

lenghtIBJ = indexBJ.Length;

indexBJ[lenghtIBJ - 1] = lenghtBJ - 1;

goto case 7;

}

}

else

{

//############## Конец 18 ###############

lenghtER = ER.Length;

ER[lenghtER - 1] = ER[lenghtER - 1] - (E - 24);

lenghtBP = BP.Length;

Array.Resize(ref BP, lenghtBP + 1);

lenghtBP = BP.Length;

BP[lenghtBP - 1] = 16 - B;

lenghtBJ = BJ.Length;

lenghtIBJ = indexBJ.Length;

Array.Resize(ref indexBJ, lenghtIBJ + 1);

lenghtIBJ = indexBJ.Length;

indexBJ[lenghtIBJ - 1] = lenghtBJ - 1;

goto case 7;

}

}

else

{

goto case 5;

}

}

}

}

goto case 3;

// CASE 5 ### CASE 5 ### CASE 5 ### CASE 5

case 5:

if (E < B)

{

lenghtEJ = EJ.Length;

Array.Resize(ref EJ, lenghtEJ + 1);

lenghtEJ = EJ.Length;

EJ[lenghtEJ - 1] = t(Convert.ToDouble(txtMathExpEj.Text));

E = E + EJ[lenghtEJ - 1];

if (E > 16)

{

lenghtBR = BR.Length;

Array.Resize(ref BR, lenghtBR + 1);

lenghtBR = BR.Length;

BR[lenghtBR - 1] = t(Convert.ToDouble(txtMathExpBr.Text));

B = B + BR[lenghtBR - 1];

int j;

j = 0;

lenghtBJ = BJ.Length;

int k = lenghtBJ;

int l = lenghtBR;

if (B < 16)

{

double[] temp2 = { Convert.ToDouble(txtMathExpBr.Text), Convert.ToDouble(txtMathExpBj.Text) };

while (B < 16)

{

j = 1 - j;

if (j == 1)

{

k++;

Array.Resize(ref BJ, k);

BJ[k - 1] = t(temp2[j]);

B = B + BJ[k - 1];

}

else

{

l++;

Array.Resize(ref BR, l);

BR[l - 1] = t(temp2[j]);

B = B + BR[l - 1];

}

}

}

if ((B > 16) && (j == 1))

{

//############### Конец 19 ################

lenghtBJ = BJ.Length;

BJ[lenghtBJ - 1] = BJ[lenghtBJ - 1] - (B - 16);

lenghtEJ = EJ.Length;

EJ[lenghtEJ - 1] = EJ[lenghtEJ - 1] - (E - 16);

goto case 7;

}

else if ((B > 16) && (j == 0))

{

if (B > 24)

{

//################### Конец 20 #############

lenghtEJ = EJ.Length;

EJ[lenghtEJ - 1] = EJ[lenghtEJ - 1] - (E - 16);

lenghtBR = BR.Length;

BR[lenghtBR - 1] = BR[lenghtBR - 1] - (B - 24);

goto case 7;

}

else

{

//################## Конец 21 ##############

lenghtEJ = EJ.Length;

EJ[lenghtEJ - 1] = EJ[lenghtEJ - 1] - (E - 16);

goto case 7;

}

}

}

else

{

goto case 6;

}

}

else

{

lenghtBP = BP.Length;

Array.Resize(ref BP, lenghtBP + 1);

lenghtBP = BP.Length;

BP[lenghtBP - 1] = E - B;

lenghtBJ = BJ.Length;

lenghtIBJ = indexBJ.Length;

Array.Resize(ref indexBJ, lenghtIBJ + 1);

lenghtIBJ = indexBJ.Length;

indexBJ[lenghtIBJ - 1] = lenghtBJ - 1;

lenghtEJ = EJ.Length;

Array.Resize(ref EJ, lenghtEJ + 1);

lenghtEJ = EJ.Length;

EJ[lenghtEJ - 1] = t(Convert.ToDouble(txtMathExpEj.Text));

E = E + EJ[lenghtEJ - 1];

lenghtBR = BR.Length;

Array.Resize(ref BR, lenghtBR + 1);

lenghtBR = BR.Length;

BR[lenghtBR - 1] = t(Convert.ToDouble(txtMathExpBr.Text));

B = B + BP[lenghtBP - 1] + BR[lenghtBR - 1];

if ((E > 16) && (B > 16))

{

if (B < 24)

{

//############## Конец 22 ###############

lenghtEJ = EJ.Length;

EJ[lenghtEJ - 1] = EJ[lenghtEJ - 1] - (E - 16);

goto case 7;

}

else

{

//############## Конец 23 ###############

lenghtEJ = EJ.Length;

EJ[lenghtEJ - 1] = EJ[lenghtEJ - 1] - (E - 16);

lenghtBR = BR.Length;

BR[lenghtBR - 1] = BR[lenghtBR - 1] - (B - 24);

goto case 7;

}

}

else

{

if ((E >= 16) && (B < 16))

{

double[] temp2 = { Convert.ToDouble(txtMathExpBr.Text), Convert.ToDouble(txtMathExpBj.Text) };

lenghtBJ = BJ.Length;

int k;

k = lenghtBJ;

lenghtBR = BR.Length;

int l;

l = lenghtBR;

int j = 0;

while (B < 16)

{

j = 1 - j;

if (j == 1)

{

k++;

Array.Resize(ref BJ, k);

BJ[k - 1] = t(temp2[j]);

B = B + BJ[k - 1];

}

else

{

l++;

Array.Resize(ref BR, l);

BR[l - 1] = t(temp2[j]);

B = B + BR[l - 1];

}

}

if ((B > 16) && (j == 0))

{

if (B < 24)

{

//############## Конец 24 ###############

lenghtEJ = EJ.Length;

EJ[lenghtEJ - 1] = EJ[lenghtEJ - 1] - (E - 16);

goto case 7;

}

else

{

//############## Конец 25 ###############

lenghtEJ = EJ.Length;

EJ[lenghtEJ - 1] = EJ[lenghtEJ - 1] - (E - 16);

lenghtBR = BR.Length;

BR[lenghtBR - 1] = BR[lenghtBR - 1] - (B - 24);

goto case 7;

}

}

else if ((B > 16) && (j == 1))

{

//############## Конец 26 ###############

lenghtEJ = EJ.Length;

EJ[lenghtEJ - 1] = EJ[lenghtEJ - 1] - (E - 16);

lenghtBJ = BJ.Length;

BJ[lenghtBJ - 1] = BJ[lenghtBJ - 1] - (B - 16);

goto case 7;

}

}

else

{

if ((E < 16) && (B >= 16))

{

if (B < 24)

{

lenghtER = ER.Length;

Array.Resize(ref ER, lenghtER + 1);

lenghtER = ER.Length;

ER[lenghtER - 1] = t(Convert.ToDouble(txtMathExpEr.Text));

if (ER[lenghtER - 1] < (24 - B))

{

//############## Конец 27 ###############

lenghtEP = EP.Length;

Array.Resize(ref EP, lenghtEP + 1);

lenghtEP = EP.Length;

EP[lenghtEP - 1] = 16 - E;

lenghtEJ = EJ.Length;

lenghtIEJ = indexEJ.Length;

Array.Resize(ref indexEJ, lenghtIEJ + 1);

lenghtIEJ = indexEJ.Length;

indexEJ[lenghtIEJ - 1] = lenghtEJ - 1;

goto case 7;

}

else

{

//############## Конец 28 ###############

lenghtER = ER.Length;

ER[lenghtER - 1] = 24 - B;

lenghtEP = EP.Length;

Array.Resize(ref EP, lenghtEP + 1);

lenghtEP = EP.Length;

EP[lenghtEP - 1] = 16 - E;

lenghtEJ = EJ.Length;

lenghtIEJ = indexEJ.Length;

Array.Resize(ref indexEJ, lenghtIEJ + 1);

lenghtIEJ = indexEJ.Length;

indexEJ[lenghtIEJ - 1] = lenghtEJ - 1;

goto case 7;

}

}

else

{

//############## Конец 29 ###############

lenghtBR = BR.Length;

BR[lenghtBR - 1] = BR[lenghtBR - 1] - (B - 24);

lenghtEP = EP.Length;

Array.Resize(ref EP, lenghtEP + 1);

lenghtEP = EP.Length;

EP[lenghtEP - 1] = 16 - E;

lenghtEJ = EJ.Length;

lenghtIEJ = indexEJ.Length;

Array.Resize(ref indexEJ, lenghtIEJ + 1);

lenghtIEJ = indexEJ.Length;

indexEJ[lenghtIEJ - 1] = lenghtEJ - 1;

goto case 7;

}

}

else

{

goto case 4;

}

}

}

}

goto case 7;

// CASE 6 ### CASE 6 ### CASE 6 ### CASE 6

case 6:

if (B < E)

{

goto case 2;

}

else

{

goto case 3;

}

// CASE 7 ### CASE 7 ### CASE 7 ### CASE 7

case 7:

break;

}

//################## Расчёт среднего ##################

if (BR.Length > 0)

{

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

{

TBR = TBR + BR[i];

}

}

if (ER.Length > 0)

{

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

{

TER = TER + ER[i];

}

}

if (BP.Length > 0)

{

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

{

TBP = TBP + BP[i];

}

}

if (EP.Length > 0)

{

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

{

TEP = TEP + EP[i];

}

}

//################## Проверка работы программы ######################

rtxbRout.AppendText("\n" + "День " + iter + "\n" + "\n");

// ########### Бульдозер ############

if (BR.Length > 0)

{

for (int a = 0; a < BJ.Length; a++)

{

rtxbRout.AppendText("Бульдозер работал: " + Convert.ToString(BJ[a]) + " часов. Всего " + Convert.ToString(BJ.Length) + " раз." + "\n");

}

for (int a = 0; a < BR.Length; a++)

{

rtxbRout.AppendText("Ремонт занял: " + Convert.ToString(BR[a]) + "\n");

}

if (BP.Length > 0)

{

for (int iterfor2 = 0; iterfor2 < BP.Length; iterfor2++)

{

rtxbRout.AppendText("\n" + "Время простоя: " + BP[iterfor2] + "\n");

}

}

}

else

{

for (int a = 0; a < BJ.Length; a++)

{

rtxbRout.AppendText("Бульдозер работал: " + Convert.ToString(BJ[a]) + " часов. Всего " + Convert.ToString(BJ.Length) + " раз." + "\n");

}

if (BP.Length > 0)

{

for (int iterfor2 = 0; iterfor2 < BP.Length; iterfor2++)

{

rtxbRout.AppendText("\n" + "Время простоя: " + BP[iterfor2] + "\n");

}

}

}

double sumBJ = 0;

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

{

sumBJ = sumBJ + BJ[i];

}

rtxbRout.AppendText("\n" + "Общее время работы: " + Convert.ToString(sumBJ) + "\n");

double sumBR = 0;

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

{

sumBR = sumBR + BR[i];

}

rtxbRout.AppendText("Общее время ремонта: " + Convert.ToString(sumBR) + "\n");

double sumBP = 0;

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

{

sumBP = sumBP + BP[i];

}

rtxbRout.AppendText("Общее время простоя: " + Convert.ToString(sumBP) + "\n");

rtxbRout.AppendText("---------------------------------------------------------------" + "\n");

//########## Экскаватор #############

if (ER.Length > 0)

{

for (int a = 0; a < EJ.Length; a++)

{

rtxbRout.AppendText("Экскаватор работал: " + Convert.ToString(EJ[a]) + " часов. Всего " + Convert.ToString(EJ.Length) + " раз." + "\n");

}

for (int a = 0; a < ER.Length; a++)

{

rtxbRout.AppendText("Ремонт занял: " + Convert.ToString(ER[a]) + "\n");

}

if (EP.Length > 0)

{

for (int iterfor2 = 0; iterfor2 < EP.Length; iterfor2++)

{

rtxbRout.AppendText("\n" + "Время простоя: " + EP[iterfor2] + "\n");

}

}

}

else

{

for (int a = 0; a < EJ.Length; a++)

{

rtxbRout.AppendText("Экскаватор работал: " + Convert.ToString(EJ[a]) + " часов. Всего " + Convert.ToString(EJ.Length) + " раз." + "\n");

}

if (EP.Length > 0)

{

for (int iterfor2 = 0; iterfor2 < EP.Length; iterfor2++)

{

rtxbRout.AppendText("\n" + "Время простоя: " + EP[iterfor2] + "\n");

}

}

}

double sumEJ = 0;

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

{

sumEJ = sumEJ + EJ[i];

}

rtxbRout.AppendText("\n" + "Общее время работы: " + Convert.ToString(sumEJ) + "\n");

double sumER = 0;

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

{

sumER = sumER + ER[i];

}

rtxbRout.AppendText("Общее время ремонта: " + Convert.ToString(sumER) + "\n");

double sumEP = 0;

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

{

sumEP = sumEP + EP[i];

}

rtxbRout.AppendText("Общее время простоя: " + Convert.ToString(sumEP) + "\n");

rtxbRout.AppendText("#############################################################" + "\n");

// ################ Вывод Расписания ################

if (iter == RndDay)

{

rtxbRasp.AppendText(" " + Convert.ToString(RndDay) + " день." + "\n");

rtxbRasp.AppendText("Бульдозер" + "\n" + "\n");

rtxbRaspE.AppendText("\n" + "Экскаватор" + "\n" + "\n");

rtxbRasp.AppendText("0 : 00" + " Начал работать бульдозер" + "\n");

rtxbRaspE.AppendText("0 : 00" + " Начал работать экскаватор" + "\n");

double timeODB = 0;

double timeODE = 0;

int maxi;

int mini;

if (BJ.Length > EJ.Length)

{

maxi = BJ.Length;

mini = EJ.Length;

}

else

{

maxi = EJ.Length;

mini = BJ.Length;

}

int prostoyB = 0;

int prostoyE = 0;

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

{

// ############ Работа бульдозера ###########

if (i < BJ.Length)

{

timeODB = timeODB + BJ[i];

rtxbRasp.AppendText(Convert.ToString(Math.Floor(timeODB)) + " : ");

if (Math.Floor(Math.Abs(60 \* Math.Round((timeODB - Math.Floor(timeODB)), 2))) < 10)

{

if (i == (BJ.Length - 1))

{

rtxbRasp.AppendText("0" + Convert.ToString(Math.Floor(Math.Abs(60 \* Math.Round((timeODB - Math.Floor(timeODB)), 2)))) + " Бульдозер закончил работу. " + "\n");

}

else

{

rtxbRasp.AppendText("0" + Convert.ToString(Math.Floor(Math.Abs(60 \* Math.Round((timeODB - Math.Floor(timeODB)), 2)))) + " Сломался бульдозер. " + "\n");

}

}

else

{

if (i == (BJ.Length - 1))

{

if (Math.Floor(Math.Abs(60 \* Math.Round((timeODB - Math.Floor(timeODB)), 2))) == 60)

{

rtxbRasp.Undo();

rtxbRasp.AppendText("16 : 00" + " Бульдозер закончил работу. " + "\n");

}

else

{

rtxbRasp.AppendText(Convert.ToString(Math.Floor(Math.Abs(60 \* Math.Round((timeODB - Math.Floor(timeODB)), 2)))) + " Бульдозер закончил работу. " + "\n");

}

}

else

{

rtxbRasp.AppendText(Convert.ToString(Math.Floor(Math.Abs(60 \* Math.Round((timeODB - Math.Floor(timeODB)), 2)))) + " Сломался бульдозер. " + "\n");

}

}

}

for (int foriter = 0; foriter < indexBJ.Length; foriter++ )

{

if (indexBJ[foriter] == i)

{

timeODB = timeODB + BP[prostoyB];

if (Math.Floor(Math.Abs(60 \* Math.Round((timeODB - Math.Floor(timeODB)), 2))) < 10)

{

rtxbRasp.AppendText(Convert.ToString(Math.Floor(timeODB)) + " : 0" + Convert.ToString(Math.Floor(Math.Abs(60 \* Math.Round((timeODB - Math.Floor(timeODB)), 2)))) + " Бульдозер начали ремонтировать." + "\n");

}

else

{

rtxbRasp.AppendText(Convert.ToString(Math.Floor(timeODB)) + " : " + Convert.ToString(Math.Floor(Math.Abs(60 \* Math.Round((timeODB - Math.Floor(timeODB)), 2)))) + " Бульдозер начали ремонтировать." + "\n");

}

prostoyB++;

}

}

// ############ Работа Экскаватора ###########

if (i < EJ.Length)

{

timeODE = timeODE + EJ[i];

rtxbRaspE.AppendText(Convert.ToString(Math.Floor(timeODE)) + " : ");

if (Math.Floor(Math.Abs(60 \* Math.Round((timeODE - Math.Floor(timeODE)), 2))) < 10)

{

if (i == (EJ.Length - 1))

{

rtxbRaspE.AppendText("0" + Convert.ToString(Math.Floor(Math.Abs(60 \* Math.Round((timeODE - Math.Floor(timeODE)), 2)))) + " Экскаватор закончил работу. " + "\n");

}

else

{

rtxbRaspE.AppendText("0" + Convert.ToString(Math.Floor(Math.Abs(60 \* Math.Round((timeODE - Math.Floor(timeODE)), 2)))) + " Сломался экскаватор. " + "\n");

}

}

else

{

if (i == (EJ.Length - 1))

{

if (Math.Floor(Math.Abs(60 \* Math.Round((timeODE - Math.Floor(timeODE)), 2))) == 60)

{

rtxbRaspE.Undo();

rtxbRaspE.AppendText("16 : 00" + " Экскаватор закончил работу. " + "\n");

}

else

{

rtxbRaspE.AppendText(Convert.ToString(Math.Floor(Math.Abs(60 \* Math.Round((timeODE - Math.Floor(timeODE)), 2)))) + " Экскаватор закончил работу. " + "\n");

}

}

else

{

rtxbRaspE.AppendText(Convert.ToString(Math.Floor(Math.Abs(60 \* Math.Round((timeODE - Math.Floor(timeODE)), 2)))) + " Сломался экскаватор. " + "\n");

}

}

}

for (int foriter = 0; foriter < indexEJ.Length; foriter++)

{

if (indexEJ[foriter] == i)

{

timeODE = timeODE + EP[prostoyE];

if (Math.Floor(Math.Abs(60 \* Math.Round((timeODE - Math.Floor(timeODE)), 2))) < 10)

{

rtxbRaspE.AppendText(Convert.ToString(Math.Floor(timeODE)) + " : 0" + Convert.ToString(Math.Floor(Math.Abs(60 \* Math.Round((timeODE - Math.Floor(timeODE)), 2)))) + " Экскаватор начали ремонтировать." + "\n");

}

else

{

rtxbRaspE.AppendText(Convert.ToString(Math.Floor(timeODE)) + " : " + Convert.ToString(Math.Floor(Math.Abs(60 \* Math.Round((timeODE - Math.Floor(timeODE)), 2)))) + " Экскаватор начали ремонтировать." + "\n");

}

prostoyE++;

}

}

// ############ Ремонт Бульдозера ###########

if (i < BR.Length)

{

timeODB = timeODB + BR[i];

rtxbRasp.AppendText(Convert.ToString(Math.Floor(timeODB)) + " : ");

if (Math.Floor(timeODB) < 16)

{

if (Math.Floor(Math.Abs(60 \* Math.Round((timeODB - Math.Floor(timeODB)), 2))) < 10)

{

rtxbRasp.AppendText("0" + Convert.ToString(Math.Floor(Math.Abs(60 \* Math.Round((timeODB - Math.Floor(timeODB)), 2)))) + " Бульдозер начал работу. " + "\n");

}

else

{

rtxbRasp.AppendText(Convert.ToString(Math.Floor(Math.Abs(60 \* Math.Round((timeODB - Math.Floor(timeODB)), 2)))) + " Бульдозер начал работу. " + "\n");

}

}

else

{

if (Math.Floor(Math.Abs(60 \* Math.Round((timeODB - Math.Floor(timeODB)), 2))) < 10)

{

rtxbRasp.AppendText("0" + Convert.ToString(Math.Floor(Math.Abs(60 \* Math.Round((timeODB - Math.Floor(timeODB)), 2)))) + " Бульдозер закончили ремонтировать. " + "\n");

}

else

{

rtxbRasp.AppendText(Convert.ToString(Math.Floor(Math.Abs(60 \* Math.Round((timeODB - Math.Floor(timeODB)), 2)))) + " Бульдозер закончили ремонтировать. " + "\n");

}

}

}

// ############ Ремонт Экскаватора ###########

if (i < ER.Length)

{

timeODE = timeODE + ER[i];

rtxbRaspE.AppendText(Convert.ToString(Math.Floor(timeODE)) + " : ");

if (Math.Floor(timeODE) < 16)

{

if (Math.Floor(Math.Abs(60 \* Math.Round((timeODE - Math.Floor(timeODE)), 2))) < 10)

{

rtxbRaspE.AppendText("0" + Convert.ToString(Math.Floor(Math.Abs(60 \* Math.Round((timeODE - Math.Floor(timeODE)), 2)))) + " Экскаватор начал работу. " + "\n");

}

else

{

rtxbRaspE.AppendText(Convert.ToString(Math.Floor(Math.Abs(60 \* Math.Round((timeODE - Math.Floor(timeODE)), 2)))) + " Экскаватор начал работу. " + "\n");

}

}

else

{

if (Math.Floor(Math.Abs(60 \* Math.Round((timeODE - Math.Floor(timeODE)), 2))) < 10)

{

rtxbRaspE.AppendText("0" + Convert.ToString(Math.Floor(Math.Abs(60 \* Math.Round((timeODE - Math.Floor(timeODE)), 2)))) + " Экскаватор закончили ремонтировать. " + "\n");

}

else

{

rtxbRaspE.AppendText(Convert.ToString(Math.Floor(Math.Abs(60 \* Math.Round((timeODE - Math.Floor(timeODE)), 2)))) + " Экскаватор закончили ремонтировать. " + "\n");

}

}

}

}

}

}

//############## Расчёт выгодности #############

int QI = Convert.ToInt32(txtQit.Text);

Cost = (TBR / QI + TER / QI) \* (ZP1 + ZP2 + NR) + TBP / QI \* UBP + TEP / QI \* UEP;

if (ZP2 == 0)

{

txtCofday1.Text = Convert.ToString(Cost);

}

else

{

txtCofday2.Text = Convert.ToString(Cost);

}

}

//##############Обработчики событий##########################

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

{

rtxbRout.Clear();

rtxbRasp.Clear();

rtxbRaspE.Clear();

mainCalc();

}

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

{

lblCheck.Text = "";

rtxbRout.Clear();

rtxbRasp.Clear();

rtxbRaspE.Clear();

txtCofday1.Clear();

txtCofday2.Clear();

}

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

{

Form1.ActiveForm.Close();

}

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

{

SaveFileDialog dlgSave = new SaveFileDialog();

try

{

dlgSave.DefaultExt = "rtf";

dlgSave.Title = "Сохранить решение";

dlgSave.Filter = "RTF Files (\*.rtf)|\*.rtf";

if (dlgSave.ShowDialog() == DialogResult.OK)

{

rtxbRout.SaveFile(dlgSave.FileName, RichTextBoxStreamType.RichNoOleObjs);

}

}

catch (Exception errorMsg)

{

MessageBox.Show(errorMsg.Message);

}

}

private void rdBtnOne\_CheckedChanged(object sender, EventArgs e)

{

txtZP2.Text = "0";

txtZP2.Visible = false;

lblZP2.Visible = false;

}

private void rdBtnTwo\_CheckedChanged(object sender, EventArgs e)

{

txtZP2.Visible = true;

lblZP2.Visible = true;

txtZP2.Text = "60";

}

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

{

if ((txtCofday1.Text == "") || (txtCofday2.Text == ""))

{

lblCheck.Text = "Сначала нужно выполнить расчёт!!!";

}

else

{

if (Convert.ToDouble(txtCofday2.Text) > Convert.ToDouble(txtCofday1.Text))

{

lblCheck.Text = "Слесаря третьего разряда нужно уволить.";

}

else

{

lblCheck.Text = "Слесаря третьего разряда не нужно увольнять.";

}

}

}

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

{

txtCofday1.Visible = true;

txtCofday2.Visible = true;

lblCOne.Visible = true;

lblCTwo.Visible = true;

}

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

{

txtCofday1.Visible = false;

txtCofday2.Visible = false;

lblCOne.Visible = false;

lblCTwo.Visible = false;

}

}

}