Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Вятский государственный университет»

Колледж ВятГУ

**ОТЧЕТ**

**ПО ДОМАШНЕЙ КОНТРОЛЬНОЙ РАБОТЕ №8**

**«ОСНОВЫ СОБЫТИЙНО-ОРИЕНТИРОВАННОГО ПРОГРАММИРОВАНИЯ»**

**ПО ДИСЦИПЛИНЕ «РАЗРАБОТКА КОДА ИНФОРМАЦИОННЫХ СИСТЕМ»**

Выполнил: студент учебной группы

ИСПк-203-52-00

Куликов Иван Алексеевич

Преподаватель:

Сергеева Елизавета Григорьевна

Киров, 2023

**Цель работы:** получение базовых навыков реализации приложений с графическим интерфейсом пользователя на основе событийно-ориентированной парадигмы.

**Задание (Вариант 12):**

1. Реализовать приложение «Калькулятор простых и сложных процентов».
2. За период начисления процентов должен приниматься месяц.

3. Предусмотреть возможность вывода пояснительной информации по расчётам.

**Код программы:**

unit main;

{$mode objfpc}{$H+}

interface

uses

Classes, SysUtils, Forms, Controls, Graphics, Dialogs, StdCtrls, ComCtrls,

ExtCtrls, Math;

type

{ TWindow }

TWindow = class(TForm)

CheckBox\_Hard: TCheckBox;

CheckBox\_Simple: TCheckBox;

Fon: TImage;

Label1: TLabel;

Per\_text: TLabel;

Raschitat: TButton;

Clear\_Button: TButton;

Input\_sum: TEdit;

Input\_per: TEdit;

Input\_mon: TEdit;

Sum\_text: TLabel;

Info\_output: TMemo;

Strelochki: TUpDown;

UpDown1: TUpDown;

procedure FonClick(Sender: TObject);

procedure Mon\_textClick(Sender: TObject);

procedure RaschitatClick(Sender: TObject);

procedure Clear\_ButtonClick(Sender: TObject);

private

public

end;

var

Window: TWindow;

implementation

{$R \*.lfm}

{ TWindow }

procedure TWindow.Mon\_textClick(Sender: TObject);

begin

end;

procedure TWindow.FonClick(Sender: TObject);

begin

end;

procedure TWindow.RaschitatClick(Sender: TObject);

var

initialSum, interestRate: Double;

monthes:Integer;

simpleInterest, complexInterest, totalSimple, totalComplex: Double;

begin

if (Length(Input\_sum.Text) = 0) or (Length(Input\_per.Text) = 0) or (Length(Input\_mon.Text) = 0) then

begin

ShowMessage('Заполните все поля для ввода!')

end

else

begin

initialSum := StrToFloat(Input\_sum.Text);

interestRate := StrToFloat(Input\_per.Text) / 100;

monthes := StrToInt(Input\_mon.Text);

simpleInterest := initialSum \* interestRate \* monthes;

totalSimple := initialSum + simpleInterest;

complexInterest := initialSum \* Power(1 + interestRate, monthes) - initialSum;

totalComplex := initialSum + complexInterest;

if (simpleInterest < 5.0E-324) or (simpleInterest > 1.7E308) or

(totalSimple < 5.0E-324) or (totalSimple > 1.7E308) or

(complexInterest < 5.0E-324) or (complexInterest > 1.7E308) or

(totalComplex < 5.0E-324) or (totalComplex > 1.7E308) then ShowMessage('Слишком большой набор входных данных')

else

begin

Info\_output.Lines.Clear;

if (CheckBox\_Simple.Checked = False) and (CheckBox\_Hard.Checked = False) then begin ShowMessage('Необходимо поставить хотя бы одну галочку!'); end;

if CheckBox\_Simple.Checked = True then

begin

Info\_output.Lines.Add(Format('Простые проценты: %f', [simpleInterest]));

Info\_output.Lines.Add(Format('Окончательная сумма с простыми процентами: %f', [totalSimple]));

end;

if CheckBox\_Hard.Checked = True then

begin

Info\_output.Lines.Add(Format('Сложные проценты: %f', [complexInterest]));

Info\_output.Lines.Add(Format('Окончательная сумма со сложными процентами: %f', [totalComplex]));

end;

end;

end;

end;

procedure TWindow.Clear\_ButtonClick(Sender: TObject);

var monthes, per:Integer;

begin

Input\_sum.Text:='';

per:=StrToInt(Input\_per.Text);

per:=0;

Input\_per.Text:=IntToStr(per);

monthes:=StrToInt(Input\_mon.Text);

monthes:=0;

Input\_mon.Text:=Inttostr(monthes);

Info\_output.Text:='';

end;

end.

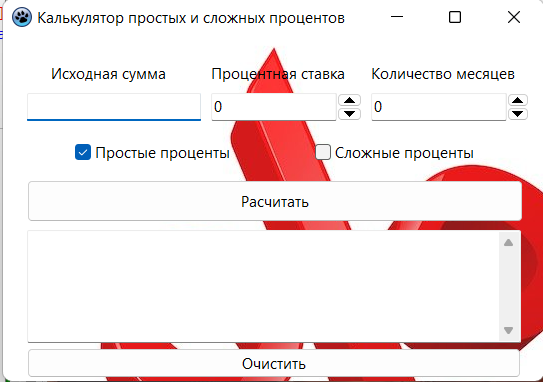
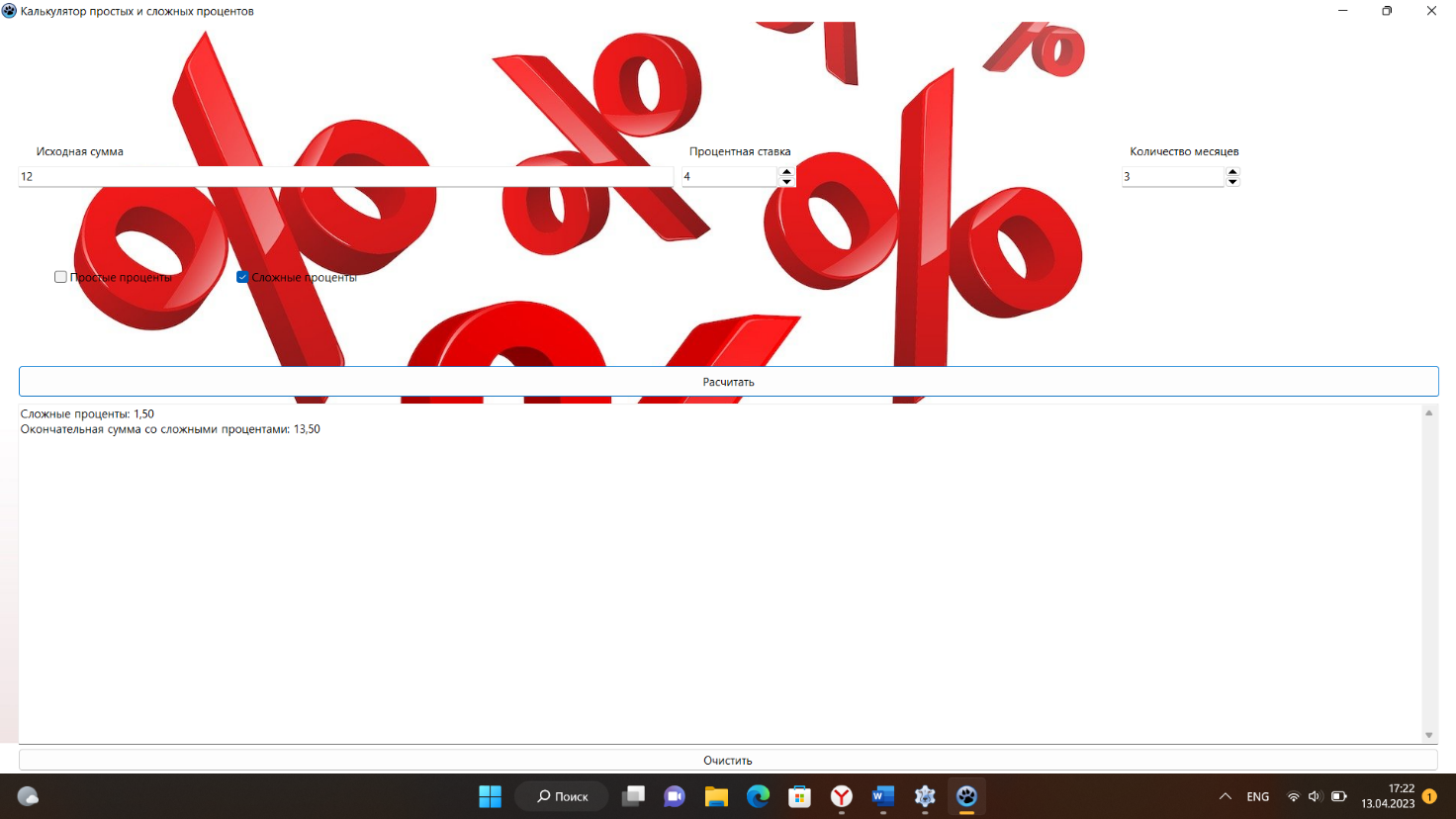
**Результат выполнения программы**

Рисунок 2 - Масштабирование программы

Рисунок 1 - Результат выполнения программы

**Вывод:**

Исходя из общего положения и пользуясь в качестве довода и логического обоснования совокупность ранее упомянутых эмпирических знаний можно сделать закономерный вывод, что в ходе работы мы использовали знания, полученные нами на столь замечательной и бесподобной дисциплине, как Разработка кода информационных систем. После домашней контрольной работы №8 мы можем смело сказать, что полученные на лекционных занятиях знания и умения в полной мере пригодились для грамотного и быстрого выполнения данной работы. Исходя из этого можно смело утверждать, что благодаря ДКР №8 возможно стать настоящим Лазарус-разработчиком.

В ходе выполнения домашней контрольной работы №8 проблем не возникло.

Подводя итоги, можно смело утверждать, что были получены базовые навыки реализации приложений с графическим интерфейсом пользователя на основе событийно-ориентированной парадигмы на Lazarus.