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-ОРИЕНТИРОВАНИГО ПРОГРАММИРОВАНИЯ»**

**ПО МДК 05.02 РАЗРАБОТКА КОДА ИНФОРМАЦИОННЫХ СИСТЕМ**

Выполнила: студентка учебной группы ИСПк-203-52-00

Гюльмамедова Лэйла эльмир-кызы

Преподаватель:

Сергеева Елизавета Григорьевна

Киров

2023

В отчете должны отображаться:

1. Цель работы
2. Скриншот задания (с вариантом)
3. Описание алгоритма
4. Код программы
5. Результат выполнения программы
6. Вывод

**1. Цель работы**

Получение базовых навыков реализации приложений с графическим интерфейсом пользователя на основе событийно–ориентированной пирамиды.

**2. Задание**

Вариант: 14

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

Задание:

1. Реализовать приложение «калькулятор простых и сложных процентов»;
2. За период начисления процентов должен приниматься день;
3. Предусмотреть возможность вывода пояснительной информации по расчётам.

**3. Описание алгоритма**

В программе используются следующие элементы:

* TEdit – для ввода информации которую нужно рассчитать
* Memo – для вывода рассчитанной информации
* TButton – триггер для расчета информации
* TImage – картинки для вывода формул расчета
* TPageControl – вкладки для переключения расчёта между простыми и сложными процентами
* TLabel – для подписи элементов TEdit чтобы пользователь знал какие данные куда он вводит

**4. Код программы**

unit unitPr;

{$mode objfpc}{$H+}

interface

uses

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

ExtCtrls, Math, Types;

type

{ TFont }

TFont = class(TForm)

ButtonRashetHard: TButton;

ButtonRaschet: TButton;

EditSrockHard: TEdit;

EditsrocHard: TEdit;

EditdepozitHard: TEdit;

EditProcent: TEdit;

EditMonth: TEdit;

EditDepozit: TEdit;

Imageformease: TImage;

Imageformease2: TImage;

Imageform2: TImage;

Imageform: TImage;

LabelSrokHard: TLabel;

LabelDepozitHard: TLabel;

LabelProcentHard: TLabel;

LabelFallProcentHard: TLabel;

LabelFallSummaHard: TLabel;

LabeProcent: TLabel;

LabelMonth: TLabel;

LabelDepozit: TLabel;

LabelItog: TLabel;

LabelProcentitog: TLabel;

MemoFallProcentHard: TMemo;

MemoFallSummaHard: TMemo;

MemoItog: TMemo;

MemoProcentitog: TMemo;

PageControl1: TPageControl;

EaseProcent: TTabSheet;

HardProcent: TTabSheet;

procedure ButtonRashetHardClick(Sender: TObject);

procedure ButtonRaschetClick(Sender: TObject);

procedure EditSrockHardChange(Sender: TObject);

procedure EditsrocHardChange(Sender: TObject);

procedure EditdepozitHardChange(Sender: TObject);

procedure EditProcentChange(Sender: TObject);

procedure EditMonthChange(Sender: TObject);

procedure EditDepozitChange(Sender: TObject);

procedure Label1Click(Sender: TObject);

procedure LabelItogClick(Sender: TObject);

procedure MemoItogChange(Sender: TObject);

procedure MemoProcentitogChange(Sender: TObject);

procedure HardProcentContextPopup(Sender: TObject; MousePos: TPoint;

var Handled: Boolean);

private

public

end;

var

Font: TFont;

a,b,c,i,d,f,g,s,e,otv:real;

implementation

{$R \*.lfm}

{ TFont }

procedure TFont.LabelItogClick(Sender: TObject);

begin

end;

procedure TFont.MemoItogChange(Sender: TObject);

begin

end;

procedure TFont.MemoProcentitogChange(Sender: TObject);

begin

end;

procedure TFont.HardProcentContextPopup(Sender: TObject; MousePos: TPoint;

var Handled: Boolean);

begin

end;

procedure TFont.EditProcentChange(Sender: TObject);

begin

if not(EditProcent.text = '') then

a:=strtofloat(EditProcent.text);

end;

procedure TFont.ButtonRaschetClick(Sender: TObject);

begin

i:=(c\*(a/12)\*b)/100;

i:=RoundTo(i,-2);

MemoProcentitog.text:=floattostr(i);

d:=i+c;

d:=RoundTo(d,-2);

MemoItog.text:=floattostr(d);

end;

procedure TFont.ButtonRashetHardClick(Sender: TObject);

begin

e:= s \* Power(1 + (f / 100 / 12), g);

e:=RoundTo(e,-2);

MemoFallSummaHard.text:=floattostr(e);

otv:= e-s;

otv:=RoundTo(otv,-2);

MemoFallProcentHard.text:=floattostr(otv);

end;

procedure TFont.EditSrockHardChange(Sender: TObject);

begin

if not(EditSrockHard.text = '') then

f:=strtofloat(EditSrockHard.text);

end;

procedure TFont.EditsrocHardChange(Sender: TObject);

begin

if not(EditsrocHard.text = '') then

g:=strtofloat(EditsrocHard.text);

end;

procedure TFont.EditdepozitHardChange(Sender: TObject);

begin

if not(EditdepozitHard.text = '') then

s:=strtofloat(EditdepozitHard.text);

end;

procedure TFont.EditMonthChange(Sender: TObject);

begin

if not(EditMonth.text = '') then

b:=strtofloat(EditMonth.text);

end;

procedure TFont.EditDepozitChange(Sender: TObject);

begin

if not(EditDepozit.text = '') then

c:=strtofloat(EditDepozit.text);

end;

procedure TFont.Label1Click(Sender: TObject);

begin

end;

end.

**5. Результат выполнения программы**

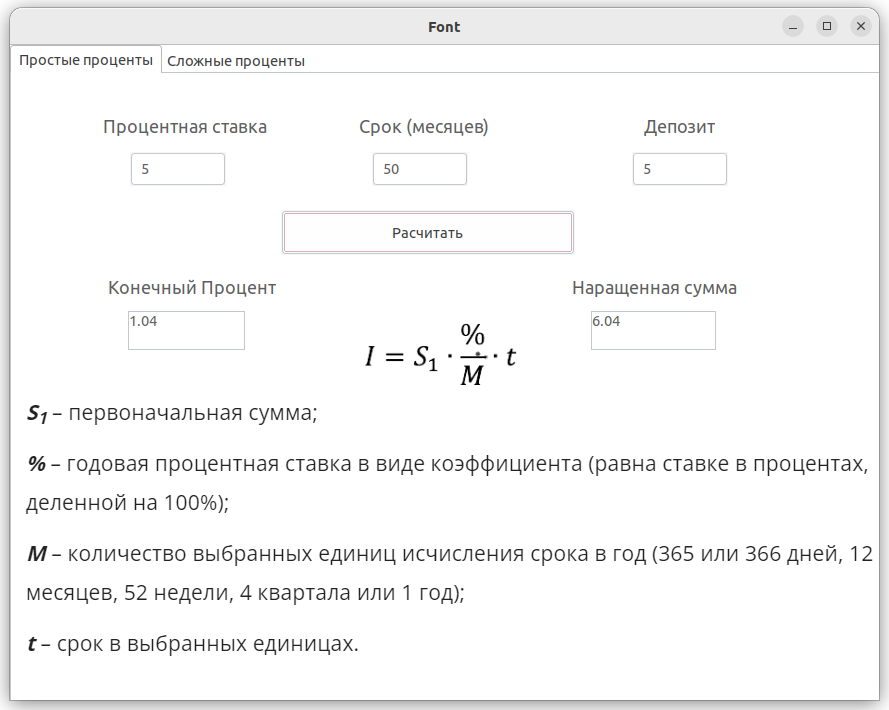


Рис 1 – Результаты выполнения программы 1-3

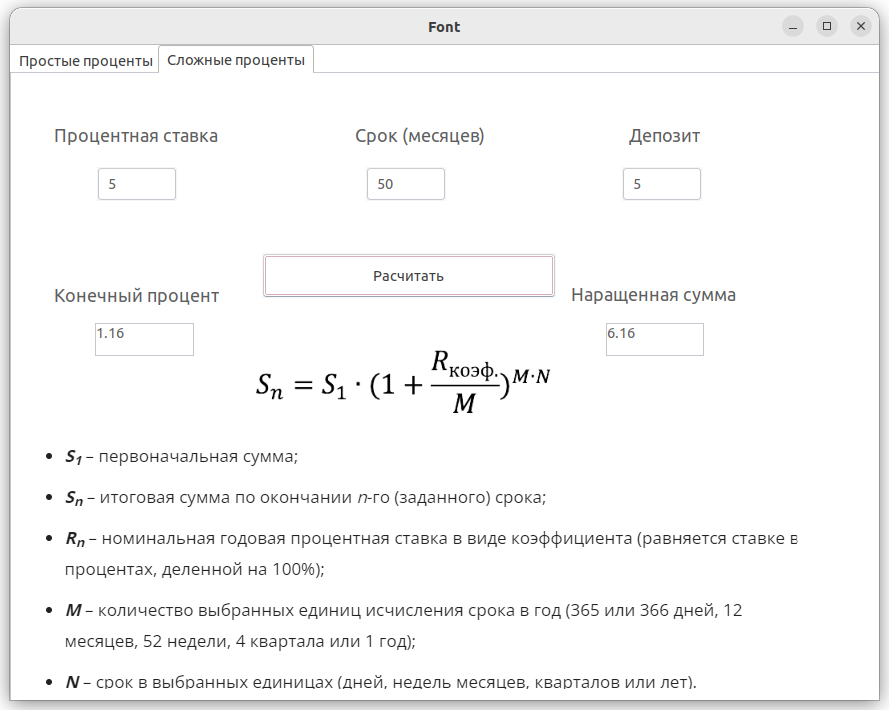


Рис 2 – Результаты выполнения программы 1-3

**6. Вывод**

В этой домашней контрольной работе были реализован калькулятор простых и сложных процентов. При написании калькулятора были использованы различные математические функции и операции, что позволило реализовать все необходимые возможности для работы с процентами.

Также в этой программе присутствует удобный и интуитивно понятный пользовательский интерфейс, который позволяет пользователям быстро освоиться с программой и начать использовать ее для своих нужд.

Расчет простых и сложных процентов разделен по вкладкам чтобы пользователь не путался при расчете разных процентов

В программе реализована возможность масштабирования элементов и минимально возможный размер окна.

В программе присутствуют картинки, которые в виде формул подробно поясняют как проводится расчет.