Univerzitet u Kragujevcu
Fakultet inženjerskih nauka

Seminarski rad iz predmeta
Softverski inženjering

Tema:
Aplikacija koja predstavlja matematički kalkulator i UML dijagrami

Student:
Nikola Mitrevski 603/2017

Predmetni profesor:
Prof. Dr. Nenad Filipović

Kragujevac 2020.

Sadržaj:

1 Postavka zadatka i detaljan opis aplikacije

1.1 Definisanje zadatka

Osnovni zadatak je pravljenje korisničke aplikacije koja simulira rad matematičkog kalkulatora.

Kalkulator treba da računa pored osnovnih matematičkih operacija(sabiranje, oduzimanje, množenje i deljenje) i napredne funkcije(logaritam, stepenovanje itd).

1.2 Opis korišćenja aplikacije

Na početku korisnik treba da bude u mogućnosti da izabere jedan od sledeća dva režima: standardni ili naučni režim.

Standardni režim treba da pruža korisniku korišćenje samo osnovnih operacija, dok naučni režim treba da pruža korisniku pored osnovnih operacija i neke napredne operacije.

Nakon biranja režima, korisnik treba da bude u mogućnosti da unese izraz za koji želi da dobije rezultat.

Kada korisnik završi sa zadavanjem izraza, on bi trebalo da bude u mogućnosti da pritisne znak jednakosti, nakon čega mu aplikacija prikazuje rezultat.

Pored zadavanja izraza, korisnik treba da bude u mogućnosti da uređuje izraz.

Kada korisnik završi sa svojim potrebama, on treba da bude u mogućnosti da na neki način može da zatvori aplikaciju.

2 Opis delova programa sa samim izvornim kodom

2.1 Uvod

Za izradu ovog projektnog zadatka korišćeno je razvojno okruženje Visual Studio, programski jezik C++.

Prilikom projektovanja je neophodno korišćenje dodatnog paketa koji se zove „System“ zbog klasa koje se nalaze u njemu, koje su potrebne za razvoj korisničke aplikacije.

Linija koda ispod predstavlja način na koji se koristi prostor(paket) „System“.

```
using namespace System;
```

2.2 Datoteka “HomeMode.cpp”

Ova datoteka sadrži poziv funkcije “main” čiji je zadatak da pokrene aplikaciju.

```
#include "HomeMode.h"
using namespace System;

[STAThread]

void main(array<String^>^ args)
{
Application::EnableVisualStyles();
Application::SetCompatibleTextRenderingDefault(false);
MathCalculator::HomeMode home;
Application::Run(% home);
}
```

2.3 Datoteka “HomeMode.h”

Ova datoteka sadrži klasu ”HomeMode” koja nasleđuje klasu ”Form”(iz paketa ”System”) čija se istanca kreira nakon pokretanja programa.

Klasa ”HomeMode” sadrži konstruktor čiji je zadatak da uradi početno inicijalizovanje.

Takođe klasa “HomeMode” sadrži sledeće metode: “standardMode_Click“, “scientificMode_Click“, “descriptionMode_Click“ i “exit_Click“.

Zadatak prve tri metode je da urade instanciranje neke od sledećih klasa: “StandardMode”, "ScientificMode" ili "DescriptionMode", respektivno u zavisnosti od potrebe.

Potreba se vrši događajem “klika“ pomoću miša na odgovarajuće dugme prozora aplikacije.

Pored instanciranja, zadatak prve tri metode je da sakriju trenutan prozor “Home“ i prikažu odgovarajući prozor(Standard, Scientific ili Description).

Zadatak poslednje metode “exit_Click“ je da zatvori aplikaciju i ona se poziva pomoću događaja “klik“.

```
#include "StandardMode.h"
#include "ScientificMode.h"
#include "DescriptionMode.h"

#pragma once
namespace MathCalculator {
using namespace System;
using namespace System::ComponentModel;
using namespace System::Collections;
using namespace System::Windows::Forms;
using namespace System::Data;
using namespace System::Drawing;

/// <summary>
/// Summary for HomeMode
/// </summary>
public ref class HomeMode : public System::Windows::Forms::Form
{
public:
HomeMode(void) {
InitializeComponent();
//
//TODO: Add the constructor code here
//
}

protected:
/// <summary>
/// Clean up any resources being used.
/// </summary>
~HomeMode() {
if (components) {
delete components;
}
}
private: System::Windows::Forms::PictureBox^ picture;
private: System::Windows::Forms::Button^ standardMode;
private: System::Windows::Forms::Label^ title;
private: System::Windows::Forms::Button^ scientificMode;
private: System::Windows::Forms::Button^ descriptionMode;
```

```
private: System::Windows::Forms::Button^ exit;

private:
/// <summary>
/// Required designer variable.
/// </summary>
System::ComponentModel::Container ^components;

#pragma region Windows Form Designer generated code
/// <summary>
/// Required method for Designer support - do not modify
/// the contents of this method with the code editor.
/// </summary>
void InitializeComponent(void) {
System::ComponentModel::ComponentResourceManager^ resources = (gcnew System::ComponentModel::ComponentResourceManager(HomeMode::typeid));
this->picture = (gcnew System::Windows::Forms::PictureBox());
this->standardMode = (gcnew System::Windows::Forms::Button());
this->title = (gcnew System::Windows::Forms::Label());
this->scientificMode = (gcnew System::Windows::Forms::Button());
this->descriptionMode = (gcnew System::Windows::Forms::Button());
this->exit = (gcnew System::Windows::Forms::Button());
(cli::safe_cast<System::ComponentModel::ISupportInitialize^>(this->picture))->BeginInit();
this->SuspendLayout();
//
// picture
//
this->picture->Image = (cli::safe_cast<System::Drawing::Image^>(resources->GetObject(L"picture.Image")));
this->picture->InitialImage = (cli::safe_cast<System::Drawing::Image^>(resources->GetObject(L"picture.InitialImage")));
this->picture->Location = System::Drawing::Point(14, 14);
this->picture->Name = L"picture";
this->picture->Size = System::Drawing::Size(80, 80);
this->picture->SizeMode = System::Windows::Forms::PictureBoxSizeMode::Zoom;
this->picture->TabIndex = 0;
this->picture->TabStop = false;
//
// standardMode
//
this->standardMode->Font = (gcnew System::Drawing::Font(L"Arial Narrow", 20.25F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(238)));
```

```
this->standardMode->Location = System::Drawing::Point(59, 113);
this->standardMode->Name = L"standardMode";
this->standardMode->Size = System::Drawing::Size(192, 44);
this->standardMode->TabIndex = 1;
this->standardMode->Text = L"Standard Mode";
this->standardMode->UseVisualStyleBackColor = true;
this->standardMode->Click += gcnew System::EventHandler(this, &HomeMode::standardMode_Click);
//
// title
//
this->title->AutoSize = true;
this->title->Font = (gcnew System::Drawing::Font(L"Arial Narrow", 24, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(238)));
this->title->Location = System::Drawing::Point(100, 39);
this->title->Name = L"title";
this->title->Size = System::Drawing::Size(203, 37);
this->title->TabIndex = 4;
this->title->Text = L"Math Calculator";
//
// scientificMode
//
this->scientificMode->Font = (gcnew System::Drawing::Font(L"Arial Narrow", 20.25F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(238)));
this->scientificMode->Location = System::Drawing::Point(59, 173);
this->scientificMode->Name = L"scientificMode";
this->scientificMode->Size = System::Drawing::Size(192, 44);
this->scientificMode->TabIndex = 5;
this->scientificMode->Text = L"Scientific Mode";
this->scientificMode->UseVisualStyleBackColor = true;
this->scientificMode->Click += gcnew System::EventHandler(this, &HomeMode::scientificMode_Click);
//
// descriptionMode
//
this->descriptionMode->Font = (gcnew System::Drawing::Font(L"Arial Narrow", 20.25F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(238)));
this->descriptionMode->Location = System::Drawing::Point(59, 233);
this->descriptionMode->Name = L"descriptionMode";
this->descriptionMode->Size = System::Drawing::Size(192, 44);
```

```
this->descriptionMode->TabIndex = 6;
this->descriptionMode->Text = L"Description Mode";
this->descriptionMode->UseVisualStyleBackColor = true;
this->descriptionMode->Click += gcnew System::EventHandler(this, &HomeMode::descriptionMode_Click);
//
// exit
//
this->exit->Font = (gcnew System::Drawing::Font(L"Arial Narrow", 20.25F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(238)));
this->exit->Location = System::Drawing::Point(59, 294);
this->exit->Name = L"exit";
this->exit->Size = System::Drawing::Size(192, 44);
this->exit->TabIndex = 7;
this->exit->Text = L"Exit";
this->exit->UseVisualStyleBackColor = true;
this->exit->Click += gcnew System::EventHandler(this, &HomeMode::exit_Click);
//
// HomeMode
//
this->AutoScaleDimensions = System::Drawing::SizeF(6, 13);
this->AutoScaleMode = System::Windows::Forms::AutoScaleMode::Font;
this->ClientSize = System::Drawing::Size(312, 357);
this->Controls->Add(this->exit);
this->Controls->Add(this->descriptionMode);
this->Controls->Add(this->scientificMode);
this->Controls->Add(this->title);
this->Controls->Add(this->standardMode);
this->Controls->Add(this->picture);
this->FormBorderStyle = System::Windows::Forms::FormBorderStyle::FixedSingle;
this->Icon = (cli::safe_cast<System::Drawing::Icon^>(resources->GetObject(L"$this.Icon")));
this->MaximizeBox = false;
this->Name = L"HomeMode";
this->StartPosition = System::Windows::Forms::FormStartPosition::CenterScreen;
this->Text = L"Home";
(cli::safe_cast<System::ComponentModel::ISupportInitialize^>(this->picture))->EndInit();
this->ResumeLayout(false);
this->PerformLayout();

}
#pragma endregion
```

```
private: System::Void standardMode_Click(System::Object^ sender, System::EventArgs^ e)
{
StandardMode^ sm = gcnew StandardMode;
this->Hide();
sm->ShowDialog();
this->Show();
}

private: System::Void scientificMode_Click(System::Object^ sender, System::EventArgs^ e)
{
ScientificMode^ sm = gcnew ScientificMode;
this->Hide();
sm->ShowDialog();
this->Show();
}

private: System::Void descriptionMode_Click(System::Object^ sender, System::EventArgs^
e) {
DescriptionMode^ dm = gcnew DescriptionMode;
this->Hide();
dm->ShowDialog();
this->Show();
}

private: System::Void exit_Click(System::Object^ sender, System::EventArgs^ e) {
Application::Exit();
}
};
}
```

2.4 Datoteka “Description.h”

Ova datoteka sadrži klasu ”Description” koja nasleđuje klasu ”Form”(iz paketa ”System”), koja sadrži konstruktor čiji je zadatak da uradi početno inicijalizovanje.

Takođe klasa “Description” sadrži metodu ”okClick” koja se poziva na dogadjaj ”klik” i čiji je zadatak da zatvori trenutan prozor “Description”.

```
#pragma once

namespace MathCalculator {
using namespace System;
using namespace System::ComponentModel;
using namespace System::Collections;
using namespace System::Windows::Forms;
```

```
using namespace System::Data;
using namespace System::Drawing;

/// <summary>
/// Summary for DescriptionMode
/// </summary>
public ref class DescriptionMode : public System::Windows::Forms::Form
{
public:
DescriptionMode(void) {
InitializeComponent();
//
//TODO: Add the constructor code here
//
}

protected:
/// <summary>
/// Clean up any resources being used.
/// </summary>
~DescriptionMode() {
if (components) {
delete components;
}
}
private: System::Windows::Forms::Button^ ok;
private: System::Windows::Forms::Label^ contact;
private: System::Windows::Forms::Label^ copyright;
private: System::Windows::Forms::Label^ version;
private: System::Windows::Forms::Label^ system;
private: System::Windows::Forms::PictureBox^ picture;
private: System::Windows::Forms::TextBox^ email;

private:
/// <summary>
/// Required designer variable.
/// </summary>
System::ComponentModel::Container ^components;

#pragma region Windows Form Designer generated code
/// <summary>
/// Required method for Designer support - do not modify
```

```
/// the contents of this method with the code editor.
/// </summary>
void InitializeComponent(void) {
System::ComponentModel::ComponentResourceManager^ resources = (gcnew System::ComponentModel::ComponentResourceManager(DescriptionMode::typeid));
this->ok = (gcnew System::Windows::Forms::Button());
this->contact = (gcnew System::Windows::Forms::Label());
this->copyright = (gcnew System::Windows::Forms::Label());
this->version = (gcnew System::Windows::Forms::Label());
this->system = (gcnew System::Windows::Forms::Label());
this->picture = (gcnew System::Windows::Forms::PictureBox());
this->email = (gcnew System::Windows::Forms::TextBox());
(cli::safe_cast<System::ComponentModel::ISupportInitialize^>(this->picture))->BeginInit();
this->SuspendLayout();
//
// ok
//
this->ok->Location = System::Drawing::Point(439, 232);
this->ok->Name = L"ok";
this->ok->Size = System::Drawing::Size(83, 33);
this->ok->TabIndex = 13;
this->ok->Text = L"OK";
this->ok->UseVisualStyleBackColor = true;
this->ok->Click += gcnew System::EventHandler(this, &DescriptionMode::okClick);
//
// contact
//
this->contact->AutoSize = true;
this->contact->Font = (gcnew System::Drawing::Font(L"Century", 12, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(238)));
this->contact->Location = System::Drawing::Point(107, 121);
this->contact->Name = L"contact";
this->contact->Size = System::Drawing::Size(75, 20);
this->contact->TabIndex = 12;
this->contact->Text = L"Contact: ";
//
// copyright
//
this->copyright->AutoSize = true;
this->copyright->Font = (gcnew System::Drawing::Font(L"Century", 12, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(238)));
```

```
this->copyright->Location = System::Drawing::Point(107, 91);
this->copyright->Name = L"copyright";
this->copyright->Size = System::Drawing::Size(421, 20);
this->copyright->TabIndex = 11;
this->copyright->Text = L"Copyright ©  2020 Nikola Mitrevski. All rights reserved.";
//
// version
//
this->version->AutoSize = true;
this->version->Font = (gcnew System::Drawing::Font(L"Century", 12, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(238)));
this->version->Location = System::Drawing::Point(107, 62);
this->version->Name = L"version";
this->version->Size = System::Drawing::Size(92, 20);
this->version->TabIndex = 10;
this->version->Text = L"Version 1.0";
//
// system
//
this->system->AutoSize = true;
this->system->Font = (gcnew System::Drawing::Font(L"Century", 12, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(238)));
this->system->Location = System::Drawing::Point(107, 33);
this->system->Name = L"system";
this->system->Size = System::Drawing::Size(147, 20);
this->system->TabIndex = 9;
this->system->Text = L"Microsoft Windows";
//
// picture
//
this->picture->Image = (cli::safe_cast<System::Drawing::Image^>(resources->GetObject(L"picture.Image")));
this->picture->Location = System::Drawing::Point(12, 46);
this->picture->Name = L"picture";
this->picture->Size = System::Drawing::Size(80, 80);
this->picture->SizeMode = System::Windows::Forms::PictureBoxSizeMode::StretchImage;
this->picture->TabIndex = 8;
this->picture->TabStop = false;
//
// email
//
```

```
this->email->BackColor = System::Drawing::SystemColors::Control;
this->email->Cursor = System::Windows::Forms::Cursors::IBeam;
this->email->Font = (gcnew System::Drawing::Font(L"Century", 12, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(238)));
this->email->Location = System::Drawing::Point(188, 118);
this->email->Name = L"email";
this->email->ReadOnly = true;
this->email->Size = System::Drawing::Size(250, 27);
this->email->TabIndex = 7;
this->email->Text = L"nikola.mitrevski1998@gmail.com";
this->email->WordWrap = false;
//
// DescriptionMode
//
this->AutoScaleDimensions = System::Drawing::SizeF(6, 13);
this->AutoScaleMode = System::Windows::Forms::AutoScaleMode::Font;
this->ClientSize = System::Drawing::Size(537, 299);
this->Controls->Add(this->ok);
this->Controls->Add(this->contact);
this->Controls->Add(this->copyright);
this->Controls->Add(this->version);
this->Controls->Add(this->system);
this->Controls->Add(this->picture);
this->Controls->Add(this->email);
this->FormBorderStyle = System::Windows::Forms::FormBorderStyle::FixedSingle;
this->Icon = (cli::safe_cast<System::Drawing::Icon^>(resources->GetObject(L"$this.Icon")));
this->MaximizeBox = false;
this->MinimizeBox = false;
this->Name = L"DescriptionMode";
this->StartPosition = System::Windows::Forms::FormStartPosition::CenterParent;
this->Text = L"AboutCalculator";
(cli::safe_cast<System::ComponentModel::ISupportInitialize^>(this->picture))->EndInit();
this->ResumeLayout(false);
this->PerformLayout();
}
#pragma endregion
private: System::Void okClick(System::Object^ sender, System::EventArgs^ e) {
this->Close();
}
};
}
```

2.5 Datoteka “StandardMode.h”

Ova datoteka sadrži klasu ”StandardMode” koja nasleđuje klasu ”Form”(iz paketa ”System”), koja sadrži konstruktor čiji je zadatak da uradi početno inicijalizovanje.

Klasa “StandardMode” pored konstruktora sadrži sledeće metode:

- metoda ”standardMode_Load” pored konstruktora vrši takođe početno podešavanje i poziva se automatski nakon konstruktora;
- metoda ”home_Click” ima namenu da zatvori trenutan prozor “StandardMode” i poziva se pomoću događaja “klik“;
- metoda ”history_Click” ima namenu da prikaže istoriju računanja i poziva se pomoću događaja “klik“;
- metoda ”exit_Click” ima namenu da zatvori aplikaciju i poziva se pomoću događaja “klik“;
- metode ”btnBackspace_Click”, ”btnCE_Click”, ”btnC_Click” imaju namene da uređuju izraz i pozivaju se pomoću događaja “klik“;
- metoda ”btnPlusMinus_Click” ima namenu da promeni znak broja i poziva se pomoću događaja “klik“;
- metoda ”btnNumber_Click” ima namenu da privremeno sačuva i prikaže kliknut broj koji će se koristiti prilikom računanja i poziva se pomoću događaja “klik“;
- metoda ”btnAritmeticOp_Click” ima namenu da privremeno sačuva i prikaže kliknutu operaciju koja će se koristiti prilikom računanja i poziva se pomoću događaja “klik“;
- metoda ”btnEqually_Click” ima namenu da izračuna zadat izraz, privremeno sačuva rezultat i prikaže ga i poziva se pomoću događaja “klik“;
- metoda ”btnDecimalPoint_Click” ima namenu da doda decimalnu tačku broju i poziva se pomoću događaja “klik“;
- metoda ”copy_Click” ima namenu da kopira (privremeno sačuva) broj i poziva se pomoću događaja “klik“;
- metoda ”paste_Click” ima namenu da prikaže privremeno sačuvan broj koji se moze koristiti u računu i poziva se pomoću događaja “klik“.

```
#pragma once

namespace MathCalculator {
using namespace System;
using namespace System::ComponentModel;
using namespace System::Collections;
using namespace System::Windows::Forms;
using namespace System::Data;
using namespace System::Drawing;

/// <summary>
/// Summary for StandardMode
```

```
/// </summary>
public ref class StandardMode : public System::Windows::Forms::Form
{
public:
StandardMode(void) {
InitializeComponent();
//
//TODO: Add the constructor code here
//
}

protected:
/// <summary>
/// Clean up any resources being used.
/// </summary>
~StandardMode() {
if (components) {
delete components;
}
}
public: System::Windows::Forms::ListBox^ historyList;

public: System::Windows::Forms::Button^ btnDecimalPoint;
public: System::Windows::Forms::Button^ btnEqually;
public: System::Windows::Forms::Button^ btnDivide;
public: System::Windows::Forms::Button^ btnZero;
public: System::Windows::Forms::Button^ btnTwo;
public: System::Windows::Forms::Button^ btnThree;
public: System::Windows::Forms::Button^ btnMultiplication;
public: System::Windows::Forms::Button^ btnOne;
public: System::Windows::Forms::Button^ btnFive;
public: System::Windows::Forms::Button^ btnSix;
public: System::Windows::Forms::Button^ btnSubtraction;
public: System::Windows::Forms::Button^ btnFour;
public: System::Windows::Forms::Button^ btnEight;
public: System::Windows::Forms::Button^ btnNine;
public: System::Windows::Forms::Button^ btnAddition;
public: System::Windows::Forms::Button^ btnSeven;
public: System::Windows::Forms::Button^ btnCE;
public: System::Windows::Forms::Button^ btnC;
public: System::Windows::Forms::Button^ btnPlusMinus;
public: System::Windows::Forms::Button^ btnBackspace;
```

```
public: System::Windows::Forms::TextBox^ display;
public: System::Windows::Forms::MenuStrip^ menu;
public: System::Windows::Forms::ToolStripMenuItem^ file;
public: System::Windows::Forms::ToolStripMenuItem^ history;
public: System::Windows::Forms::ToolStripMenuItem^ exit;
public: System::Windows::Forms::ToolStripMenuItem^ edit;
public: System::Windows::Forms::ToolStripMenuItem^ copy;
public: System::Windows::Forms::ToolStripMenuItem^ paste;
public: System::Windows::Forms::ToolStripMenuItem^ home;
public: System::Windows::Forms::Label^ lblShow;

public:
/// <summary>
/// Required designer variable.
/// </summary>
System::ComponentModel::Container ^components;

#pragma region Windows Form Designer generated code
/// <summary>
/// Required method for Designer support - do not modify
/// the contents of this method with the code editor.
/// </summary>
void InitializeComponent(void) {
System::ComponentModel::ComponentResourceManager^ resources = (gcnew System::ComponentModel::ComponentResourceManager(StandardMode::typeid));
this->historyList = (gcnew System::Windows::Forms::ListBox());
this->btnDecimalPoint = (gcnew System::Windows::Forms::Button());
this->btnEqually = (gcnew System::Windows::Forms::Button());
this->btnDivide = (gcnew System::Windows::Forms::Button());
this->btnZero = (gcnew System::Windows::Forms::Button());
this->btnTwo = (gcnew System::Windows::Forms::Button());
this->btnThree = (gcnew System::Windows::Forms::Button());
this->btnMultiplication = (gcnew System::Windows::Forms::Button());
this->btnOne = (gcnew System::Windows::Forms::Button());
this->btnFive = (gcnew System::Windows::Forms::Button());
this->btnSix = (gcnew System::Windows::Forms::Button());
this->btnSubtraction = (gcnew System::Windows::Forms::Button());
this->btnFour = (gcnew System::Windows::Forms::Button());
this->btnEight = (gcnew System::Windows::Forms::Button());
this->btnNine = (gcnew System::Windows::Forms::Button());
this->btnAddition = (gcnew System::Windows::Forms::Button());
```

```
this->btnSeven = (gcnew System::Windows::Forms::Button());
this->btnCE = (gcnew System::Windows::Forms::Button());
this->btnC = (gcnew System::Windows::Forms::Button());
this->btnPlusMinus = (gcnew System::Windows::Forms::Button());
this->btnBackspace = (gcnew System::Windows::Forms::Button());
this->display = (gcnew System::Windows::Forms::TextBox());
this->menu = (gcnew System::Windows::Forms::MenuStrip());
this->file = (gcnew System::Windows::Forms::ToolStripMenuItem());
this->home = (gcnew System::Windows::Forms::ToolStripMenuItem());
this->history = (gcnew System::Windows::Forms::ToolStripMenuItem());
this->exit = (gcnew System::Windows::Forms::ToolStripMenuItem());
this->edit = (gcnew System::Windows::Forms::ToolStripMenuItem());
this->copy = (gcnew System::Windows::Forms::ToolStripMenuItem());
this->paste = (gcnew System::Windows::Forms::ToolStripMenuItem());
this->lblShow = (gcnew System::Windows::Forms::Label());
this->menu->SuspendLayout();
this->SuspendLayout();
//
// historyList
//
this->historyList->FormattingEnabled = true;
this->historyList->Location = System::Drawing::Point(12, 447);
this->historyList->Name = L"historyList";
this->historyList->Size = System::Drawing::Size(286, 134);
this->historyList->TabIndex = 72;
//
// btnDecimalPoint
//
this->btnDecimalPoint->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif",
14.25F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(238)));
this->btnDecimalPoint->Location = System::Drawing::Point(85, 375);
this->btnDecimalPoint->Name = L"btnDecimalPoint";
this->btnDecimalPoint->Size = System::Drawing::Size(67, 63);
this->btnDecimalPoint->TabIndex = 70;
this->btnDecimalPoint->Text = L".";
this->btnDecimalPoint->UseVisualStyleBackColor = true;
this->btnDecimalPoint->Click += gcnew System::EventHandler(this,
&StandardMode::btnDecimalPoint_Click);
//
// btnEqually
//
```

```
this->btnEqually->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 14.25F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(238)));
this->btnEqually->Location = System::Drawing::Point(158, 375);
this->btnEqually->Name = L"btnEqually";
this->btnEqually->Size = System::Drawing::Size(67, 63);
this->btnEqually->TabIndex = 69;
this->btnEqually->Text = L"=";
this->btnEqually->UseVisualStyleBackColor = true;
this->btnEqually->Click += gcnew System::EventHandler(this, &StandardMode::btnEqually_Click);
//
// btnDivide
//
this->btnDivide->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 14.25F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(238)));
this->btnDivide->Location = System::Drawing::Point(231, 375);
this->btnDivide->Name = L"btnDivide";
this->btnDivide->Size = System::Drawing::Size(67, 63);
this->btnDivide->TabIndex = 68;
this->btnDivide->Text = L"/";
this->btnDivide->UseVisualStyleBackColor = true;
this->btnDivide->Click += gcnew System::EventHandler(this, &StandardMode::btnAritmeticOp_Click);
//
// btnZero
//
this->btnZero->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 14.25F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(238)));
this->btnZero->Location = System::Drawing::Point(11, 375);
this->btnZero->Name = L"btnZero";
this->btnZero->Size = System::Drawing::Size(67, 63);
this->btnZero->TabIndex = 67;
this->btnZero->Text = L"0";
this->btnZero->UseVisualStyleBackColor = true;
this->btnZero->Click += gcnew System::EventHandler(this, &StandardMode::btnNumber_Click);
//
// btnTwo
//
```

```
this->btnTwo->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 14.25F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(238)));
this->btnTwo->Location = System::Drawing::Point(84, 306);
this->btnTwo->Name = L"btnTwo";
this->btnTwo->Size = System::Drawing::Size(67, 63);
this->btnTwo->TabIndex = 66;
this->btnTwo->Text = L"2";
this->btnTwo->UseVisualStyleBackColor = true;
this->btnTwo->Click += gcnew System::EventHandler(this, &StandardMode::btnNumber_Click);
//
// btnThree
//
this->btnThree->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 14.25F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(238)));
this->btnThree->Location = System::Drawing::Point(157, 307);
this->btnThree->Name = L"btnThree";
this->btnThree->Size = System::Drawing::Size(67, 63);
this->btnThree->TabIndex = 65;
this->btnThree->Text = L"3";
this->btnThree->UseVisualStyleBackColor = true;
this->btnThree->Click += gcnew System::EventHandler(this, &StandardMode::btnNumber_Click);
//
// btnMultiplication
//
this->btnMultiplication->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 14.25F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(238)));
this->btnMultiplication->Location = System::Drawing::Point(230, 306);
this->btnMultiplication->Name = L"btnMultiplication";
this->btnMultiplication->Size = System::Drawing::Size(67, 63);
this->btnMultiplication->TabIndex = 64;
this->btnMultiplication->Text = L"*";
this->btnMultiplication->UseVisualStyleBackColor = true;
this->btnMultiplication->Click += gcnew System::EventHandler(this, &StandardMode::btnAritmeticOp_Click);
//
// btnOne
//
```

```
this->btnOne->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 14.25F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(238)));
this->btnOne->Location = System::Drawing::Point(11, 306);
this->btnOne->Name = L"btnOne";
this->btnOne->Size = System::Drawing::Size(67, 63);
this->btnOne->TabIndex = 63;
this->btnOne->Text = L"1";
this->btnOne->UseVisualStyleBackColor = true;
this->btnOne->Click += gcnew System::EventHandler(this, &StandardMode::btnNumber_Click);
//
// btnFive
//
this->btnFive->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 14.25F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(238)));
this->btnFive->Location = System::Drawing::Point(84, 237);
this->btnFive->Name = L"btnFive";
this->btnFive->Size = System::Drawing::Size(67, 63);
this->btnFive->TabIndex = 62;
this->btnFive->Text = L"5";
this->btnFive->UseVisualStyleBackColor = true;
this->btnFive->Click += gcnew System::EventHandler(this, &StandardMode::btnNumber_Click);
//
// btnSix
//
this->btnSix->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 14.25F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(238)));
this->btnSix->Location = System::Drawing::Point(157, 237);
this->btnSix->Name = L"btnSix";
this->btnSix->Size = System::Drawing::Size(67, 63);
this->btnSix->TabIndex = 61;
this->btnSix->Text = L"6";
this->btnSix->UseVisualStyleBackColor = true;
this->btnSix->Click += gcnew System::EventHandler(this, &StandardMode::btnNumber_Click);
//
// btnSubtraction
//
```

```
this->btnSubtraction->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 14.25F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(238)));
this->btnSubtraction->Location = System::Drawing::Point(230, 237);
this->btnSubtraction->Name = L"btnSubtraction";
this->btnSubtraction->Size = System::Drawing::Size(67, 63);
this->btnSubtraction->TabIndex = 60;
this->btnSubtraction->Text = L"-";
this->btnSubtraction->UseVisualStyleBackColor = true;
this->btnSubtraction->Click += gcnew System::EventHandler(this, &StandardMode::btnAritmeticOp_Click);
//
// btnFour
//
this->btnFour->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 14.25F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(238)));
this->btnFour->Location = System::Drawing::Point(11, 237);
this->btnFour->Name = L"btnFour";
this->btnFour->Size = System::Drawing::Size(67, 63);
this->btnFour->TabIndex = 59;
this->btnFour->Text = L"4";
this->btnFour->UseVisualStyleBackColor = true;
this->btnFour->Click += gcnew System::EventHandler(this, &StandardMode::btnNumber_Click);
//
// btnEight
//
this->btnEight->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 14.25F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(238)));
this->btnEight->Location = System::Drawing::Point(85, 168);
this->btnEight->Name = L"btnEight";
this->btnEight->Size = System::Drawing::Size(67, 63);
this->btnEight->TabIndex = 58;
this->btnEight->Text = L"8";
this->btnEight->UseVisualStyleBackColor = true;
this->btnEight->Click += gcnew System::EventHandler(this, &StandardMode::btnNumber_Click);
//
// btnNine
//
```

```
this->btnNine->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 14.25F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(238)));
this->btnNine->Location = System::Drawing::Point(158, 168);
this->btnNine->Name = L"btnNine";
this->btnNine->Size = System::Drawing::Size(67, 63);
this->btnNine->TabIndex = 57;
this->btnNine->Text = L"9";
this->btnNine->UseVisualStyleBackColor = true;
this->btnNine->Click += gcnew System::EventHandler(this, &StandardMode::btnNumber_Click);
//
// btnAddition
//
this->btnAddition->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 14.25F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(238)));
this->btnAddition->Location = System::Drawing::Point(231, 168);
this->btnAddition->Name = L"btnAddition";
this->btnAddition->Size = System::Drawing::Size(67, 63);
this->btnAddition->TabIndex = 56;
this->btnAddition->Text = L"+";
this->btnAddition->UseVisualStyleBackColor = true;
this->btnAddition->Click += gcnew System::EventHandler(this, &StandardMode::btnAritmeticOp_Click);
//
// btnSeven
//
this->btnSeven->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 14.25F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(238)));
this->btnSeven->Location = System::Drawing::Point(11, 168);
this->btnSeven->Name = L"btnSeven";
this->btnSeven->Size = System::Drawing::Size(67, 63);
this->btnSeven->TabIndex = 55;
this->btnSeven->Text = L"7";
this->btnSeven->UseVisualStyleBackColor = true;
this->btnSeven->Click += gcnew System::EventHandler(this, &StandardMode::btnNumber_Click);
//
// btnCE
//
```

```
this->btnCE->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 14.25F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(238)));
this->btnCE->Location = System::Drawing::Point(85, 99);
this->btnCE->Name = L"btnCE";
this->btnCE->Size = System::Drawing::Size(67, 63);
this->btnCE->TabIndex = 54;
this->btnCE->Text = L"CE";
this->btnCE->UseVisualStyleBackColor = true;
this->btnCE->Click += gcnew System::EventHandler(this, &StandardMode::btnCE_Click);
//
// btnC
//
this->btnC->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 14.25F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(238)));
this->btnC->Location = System::Drawing::Point(158, 99);
this->btnC->Name = L"btnC";
this->btnC->Size = System::Drawing::Size(67, 63);
this->btnC->TabIndex = 53;
this->btnC->Text = L"C";
this->btnC->UseVisualStyleBackColor = true;
this->btnC->Click += gcnew System::EventHandler(this, &StandardMode::btnC_Click);
//
// btnPlusMinus
//
this->btnPlusMinus->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 20.25F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(238)));
this->btnPlusMinus->Location = System::Drawing::Point(231, 99);
this->btnPlusMinus->Name = L"btnPlusMinus";
this->btnPlusMinus->Size = System::Drawing::Size(67, 63);
this->btnPlusMinus->TabIndex = 52;
this->btnPlusMinus->Text = L"±";
this->btnPlusMinus->UseVisualStyleBackColor = true;
this->btnPlusMinus->Click += gcnew System::EventHandler(this, &StandardMode::btnPlusMinus_Click);
//
// btnBackspace
//
this->btnBackspace->Font = (gcnew System::Drawing::Font(L"Wingdings", 14.25F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(2)));
```

```
this->btnBackspace->Location = System::Drawing::Point(12, 99);
this->btnBackspace->Name = L"btnBackspace";
this->btnBackspace->Size = System::Drawing::Size(67, 63);
this->btnBackspace->TabIndex = 51;
this->btnBackspace->Text = L"□";
this->btnBackspace->UseVisualStyleBackColor = true;
this->btnBackspace->Click += gcnew System::EventHandler(this, &StandardMode::btnBackspace_Click);
//
// display
//
this->display->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 14.25F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(238)));
this->display->Location = System::Drawing::Point(12, 28);
this->display->Multiline = true;
this->display->Name = L"display";
this->display->ReadOnly = true;
this->display->Size = System::Drawing::Size(287, 60);
this->display->TabIndex = 71;
this->display->TextAlign = System::Windows::Forms::HorizontalAlignment::Right;
//
// menu
//
this->menu->Items->AddRange(gcnew cli::array< System::Windows::Forms::ToolStripItem^ >(2) { this->file, this->edit });
this->menu->Location = System::Drawing::Point(0, 0);
this->menu->Name = L"menu";
this->menu->Size = System::Drawing::Size(309, 24);
this->menu->TabIndex = 50;
this->menu->Text = L"menu";
//
// file
//
this->file->DropDownItems->AddRange(gcnew cli::array< System::Windows::Forms::ToolStripItem^ >(3) {
this->home, this->history,
this->exit
});
this->file->Name = L"file";
this->file->Size = System::Drawing::Size(37, 20);
this->file->Text = L"File";
//
```

```
// home
//
this->home->Name = L"home";
this->home->Size = System::Drawing::Size(112, 22);
this->home->Text = L"Home";
this->home->Click += gcnew System::EventHandler(this, &StandardMode::home_Click);
//
// history
//
this->history->Name = L"history";
this->history->Size = System::Drawing::Size(112, 22);
this->history->Text = L"History";
this->history->Click += gcnew System::EventHandler(this, &StandardMode::history_Click);
//
// exit
//
this->exit->Name = L"exit";
this->exit->Size = System::Drawing::Size(112, 22);
this->exit->Text = L"Exit";
this->exit->Click += gcnew System::EventHandler(this, &StandardMode::exit_Click);
//
// edit
//
this->edit->DropDownItems->AddRange(gcnew cli::array< System::Windows::Forms::ToolStripItem^ >(2) { this->copy, this->paste });
this->edit->Name = L"edit";
this->edit->Size = System::Drawing::Size(39, 20);
this->edit->Text = L"Edit";
//
// copy
//
this->copy->Name = L"copy";
this->copy->Size = System::Drawing::Size(102, 22);
this->copy->Text = L"Copy";
this->copy->Click += gcnew System::EventHandler(this, &StandardMode::copy_Click);
//
// paste
//
this->paste->Name = L"paste";
this->paste->Size = System::Drawing::Size(102, 22);
this->paste->Text = L"Paste";
this->paste->Click += gcnew System::EventHandler(this, &StandardMode::paste_Click);
```

```
//
// lblShow
//
this->lblShow->AutoSize = true;
this->lblShow->BackColor = System::Drawing::SystemColors::Control;
this->lblShow->Location = System::Drawing::Point(15, 31);
this->lblShow->Name = L"lblShow";
this->lblShow->Size = System::Drawing::Size(29, 13);
this->lblShow->TabIndex = 73;
this->lblShow->Text = L"label";
//
// StandardMode
//
this->AutoScaleDimensions = System::Drawing::SizeF(6, 13);
this->AutoScaleMode = System::Windows::Forms::AutoScaleMode::Font;
this->ClientSize = System::Drawing::Size(309, 447);
this->Controls->Add(this->lblShow);
this->Controls->Add(this->historyList);
this->Controls->Add(this->btnDecimalPoint);
this->Controls->Add(this->btnEqually);
this->Controls->Add(this->btnDivide);
this->Controls->Add(this->btnZero);
this->Controls->Add(this->btnTwo);
this->Controls->Add(this->btnThree);
this->Controls->Add(this->btnMultiplication);
this->Controls->Add(this->btnOne);
this->Controls->Add(this->btnFive);
this->Controls->Add(this->btnSix);
this->Controls->Add(this->btnSubtraction);
this->Controls->Add(this->btnFour);
this->Controls->Add(this->btnEight);
this->Controls->Add(this->btnNine);
this->Controls->Add(this->btnAddition);
this->Controls->Add(this->btnSeven);
this->Controls->Add(this->btnCE);
this->Controls->Add(this->btnC);
this->Controls->Add(this->btnPlusMinus);
this->Controls->Add(this->btnBackspace);
this->Controls->Add(this->display);
this->Controls->Add(this->menu);
this->FormBorderStyle = System::Windows::Forms::FormBorderStyle::FixedSingle;
```

```
this->Icon = (cli::safe_cast<System::Drawing::Icon^>(resources->GetObject(L"$this.Icon")));
this->MaximizeBox = false;
this->Name = L"StandardMode";
this->StartPosition = System::Windows::Forms::FormStartPosition::CenterParent;
this->Text = L"Standard";
this->Load += gcnew System::EventHandler(this, &StandardMode::standardMode_Load);
this->menu->ResumeLayout(false);
this->menu->PerformLayout();
this->ResumeLayout(false);
this->PerformLayout();
}
#pragma endregion
public: double iFirstnum;
public: double iSecondnum;
public: double iResult;
public: double a;
public: String^ iOperator;

public: String^ copyX = "0";
public: String^ pasteX;

bool state = true;

//pogledati i popraviti metodu ispod
public: System::Void standardMode_Load(System::Object^ sender, System::EventArgs^ e) {
this->Width = 325;
this->Height = 485;
display->Text = "0";
lblShow->Text = "0";
historyList->Items->Clear();
history->Checked = false;
}

private: System::Void home_Click(System::Object^ sender, System::EventArgs^ e) {
this->Close();
}

public: System::Void history_Click(System::Object^ sender, System::EventArgs^ e) {
history->Checked = state;
if (history->Checked == true) {
historyList->Visible = true;
```

```
this->Height = 631;
state = false;
}
else if (history->Checked == false) {
historyList->Visible = false;
this->Height = 485;
state = true;
}
}

public: System::Void exit_Click(System::Object^ sender, System::EventArgs^ e) {
this->Close();
Application::Exit();
}

// Button Backspace
public: System::Void btnBackspace_Click(System::Object^ sender, System::EventArgs^ e) {
if (display->Text->Length > 0) {
display->Text = display->Text->Remove(display->Text->Length - 1, 1);
}
if ((display->Text == "") || (display->Text == "-")) {
display->Text = "0";
}
}

// Button CE
public: System::Void btnCE_Click(System::Object^ sender, System::EventArgs^ e) {
display->Text = "0";
}

// Button C
public: System::Void btnC_Click(System::Object^ sender, System::EventArgs^ e) {
lblShow->Text = "0";
display->Text = "0";
}

// Button PlusMinus
public: System::Void btnPlusMinus_Click(System::Object^ sender, System::EventArgs^ e) {
if (display->Text->Contains("-")) {
display->Text = display->Text->Remove(0, 1);
}
else {
```

```
display->Text = "-" + display->Text;
}
}

// Buttons Numbers(0, 1, 2, 3, 4, 5, 6, 7, 8, 9)
public: System::Void btnNumber_Click(System::Object^ sender, System::EventArgs^ e) {
Button^ Numbers = safe_cast<Button^>(sender);
if (display->Text == "0") {
display->Text = Numbers->Text;
}
else {
display->Text = display->Text + Numbers->Text;
}
}

//Buttons Arithmetic Operations(+, -, *, /)
public: System::Void btnAritmeticOp_Click(System::Object^ sender, System::EventArgs^ e)
{
Button^ op = safe_cast<Button^>(sender);
iFirstnum = Double::Parse(display->Text);
display->Text = "0";
iOperator = op->Text;
lblShow->Text = System::Convert::ToString(iFirstnum) + " " + iOperator;
}

// Button Equal
public: System::Void btnEqually_Click(System::Object^ sender, System::EventArgs^ e) {
iSecondnum = Double::Parse(display->Text);
if (iOperator == "+") {
iResult = iFirstnum + iSecondnum;
display->Text = System::Convert::ToString(iResult);
historyList->Items->Add(lblShow->Text + " " + System::Convert::ToString(iSecondnum));
}
else if (iOperator == "-") {
iResult = iFirstnum - iSecondnum;
display->Text = System::Convert::ToString(iResult);
historyList->Items->Add(lblShow->Text + " " + System::Convert::ToString(iSecondnum));
}
else if (iOperator == "*") {
iResult = iFirstnum * iSecondnum;
display->Text = System::Convert::ToString(iResult);
historyList->Items->Add(lblShow->Text + " " + System::Convert::ToString(iSecondnum));
```

```
}
else if (iOperator == "/") {
iResult = iFirstnum / iSecondnum;
display->Text = System::Convert::ToString(iResult);
historyList->Items->Add(lblShow->Text + " " + System::Convert::ToString(iSecondnum));
}
else if (iOperator == "Mod") {
int iFirst, iSecond, iRes;
iFirst = Convert::ToInt32(iFirstnum);
iSecond = Convert::ToInt32(iSecondnum);
iRes = iFirst % iSecond;
display->Text = System::Convert::ToString(iRes);
historyList->Items->Add(lblShow->Text + " " + System::Convert::ToString(iSecondnum));
}
else if (iOperator == "Pow") {
iResult = Math::Pow(iFirstnum, iSecondnum);
display->Text = System::Convert::ToString(iResult);
historyList->Items->Add(lblShow->Text + " " + System::Convert::ToString(iSecondnum));
}
}

// Button Decimal Point
public: System::Void btnDecimalPoint_Click(System::Object^ sender, System::EventArgs^ e) {
if (!display->Text->Contains(",")) {
display->Text = display->Text + ",";
}
}

//Copy And Paste Options
public: System::Void copy_Click(System::Object^ sender, System::EventArgs^ e) {
copyX = display->Text;
}

public: System::Void paste_Click(System::Object^ sender, System::EventArgs^ e) {
pasteX = copyX;
display->Text = pasteX;
}
};
}
```

2.6 Datoteka “ScientificMode.h”

Ova datoteka sadrži klasu ”ScientificMode” koja nasleđuje klasu ”StandardMode” (iz paketa ”MathCalculator”), koja sadrži konstruktor čiji je zadatak da uradi početno inicijalizovanje.

Takođe klasa ”ScientificMode” nasleđuje javne metode i članove klase ”StandardMode”.

Klasa “ScientificMode” pored konstruktora sadrži sledeće metode:

- metoda ”scientificMode_Load” pored konstruktora vrši takođe početno podešavanje i poziva se automatski nakon konstruktora;
- metoda ”btnPI_Click” ima namenu da dodeli izrazu vrednost PI i poziva se pomoću događaja “klik“;
- metoda ”btnLog_Click” ima namenu da izračuna logaritamsku vrednost broja i poziva se pomoću događaja “klik“;
- metoda ”btnQuadrate_Click” ima namenu da izračuna kvadratnu vrednost broja i poziva se pomoću događaja “klik“;
- metoda ”btnSinh_Click” ima namenu da izračuna hiperbolički sinunus ugla broja dat u radijanima i poziva se pomoću događaja “klik“;
- metoda ”btnSin_Click” ima namenu da izračuna sinunus ugla broja dat u radijanima i poziva se pomoću događaja “klik“;
- metoda ”btnSqrt_Click” ima namenu da izračuna kvadratni koren vrednosti broja i poziva se pomoću događaja “klik“;
- metoda ”btnCub_Click” ima namenu da izračuna kubni stepen vrednosti broja i poziva se pomoću događaja “klik“;
- metoda ”btnCosh_Click” ima namenu da izračuna hiperbolički kosinus ugla broja dat u radijanima i poziva se pomoću događaja “klik“;
- metoda ”btnCos_Click” ima namenu da izračuna kosinus ugla broja dat u radijanima i poziva se pomoću događaja “klik“;
- metoda ”btnBin_Click” ima namenu da izračuna binarnu vrednost broja i poziva se pomoću događaja “klik“;
- metoda ”btnTanh_Click” ima namenu da izračuna hiperbolički tangens ugla broja dat u radijanima i poziva se pomoću događaja “klik“;
- metoda ”btnTan_Click” ima namenu da izračuna tangens ugla broja dat u radijanima i poziva se pomoću događaja “klik“;
- metoda ”btnHex_Click” ima namenu da izračuna heksadecimalnu vrednost broja i poziva se pomoću događaja “klik“;
- metoda ”btnFactorial_Click” ima namenu da izračuna faktorijal vrednosti broja i poziva se pomoću događaja “klik“;
- metoda ”btnPercent_Click” ima namenu da izračuna procentualnu vrednost broja i poziva se pomoću događaja “klik“;
- metoda ”btnOct_Click” ima namenu da izračuna oktalnu vrednost broja i poziva se pomoću događaja “klik“;
- metoda ”btnReciprocity_Click” ima namenu da izračuna reciprocitacnu vrednost broja i poziva se pomoću događaja “klik“.

```
#include "StandardMode.h"

#pragma once

namespace MathCalculator {
using namespace System;
using namespace System::ComponentModel;
using namespace System::Collections;
using namespace System::Windows::Forms;
using namespace System::Data;
using namespace System::Drawing;

/// <summary>
/// Summary for ScientificMode
/// </summary>
public ref class ScientificMode : public StandardMode
{

public:
ScientificMode(void) {
InitializeComponent();
//
//TODO: Add the constructor code here
//
}

protected:
/// <summary>
/// Clean up any resources being used.
/// </summary>
~ScientificMode() {
if (components) {
delete components;
}
}

private: System::Windows::Forms::Button^ btnReciprocity;
```

```
private: System::Windows::Forms::Button^ btnCub;
private: System::Windows::Forms::Button^ btnMod;
private: System::Windows::Forms::Button^ btnOct;
private: System::Windows::Forms::Button^ btnPercent;
private: System::Windows::Forms::Button^ btnPow;
private: System::Windows::Forms::Button^ btnTan;
private: System::Windows::Forms::Button^ btnHex;
private: System::Windows::Forms::Button^ btnLn;
private: System::Windows::Forms::Button^ btnTanh;
private: System::Windows::Forms::Button^ btnCos;
private: System::Windows::Forms::Button^ btnBin;
private: System::Windows::Forms::Button^ btnLog;
private: System::Windows::Forms::Button^ btnCosh;
private: System::Windows::Forms::Button^ btnSin;
private: System::Windows::Forms::Button^ btnFactorial;
private: System::Windows::Forms::Button^ btnSqrt;
private: System::Windows::Forms::Button^ btnSinh;
private: System::Windows::Forms::Button^ btnQuadrate;
private: System::Windows::Forms::Button^ btnPI;

private:
/// <summary>
/// Required designer variable.
/// </summary>
System::ComponentModel::Container ^components;

#pragma region Windows Form Designer generated code
/// <summary>
/// Required method for Designer support - do not modify
/// the contents of this method with the code editor.
/// </summary>
void InitializeComponent(void) {
this->btnReciprocity = (gcnew System::Windows::Forms::Button());
this->btnCub = (gcnew System::Windows::Forms::Button());
this->btnMod = (gcnew System::Windows::Forms::Button());
this->btnOct = (gcnew System::Windows::Forms::Button());
this->btnPercent = (gcnew System::Windows::Forms::Button());
this->btnPow = (gcnew System::Windows::Forms::Button());
```

```
this->btnTan = (gcnew System::Windows::Forms::Button());
this->btnHex = (gcnew System::Windows::Forms::Button());
this->btnLn = (gcnew System::Windows::Forms::Button());
this->btnTanh = (gcnew System::Windows::Forms::Button());
this->btnCos = (gcnew System::Windows::Forms::Button());
this->btnBin = (gcnew System::Windows::Forms::Button());
this->btnLog = (gcnew System::Windows::Forms::Button());
this->btnCosh = (gcnew System::Windows::Forms::Button());
this->btnSin = (gcnew System::Windows::Forms::Button());
this->btnFactorial = (gcnew System::Windows::Forms::Button());
this->btnSqrt = (gcnew System::Windows::Forms::Button());
this->btnSinh = (gcnew System::Windows::Forms::Button());
this->btnQuadrate = (gcnew System::Windows::Forms::Button());
this->btnPI = (gcnew System::Windows::Forms::Button());
this->SuspendLayout();
//
// btnReciprocity
//
this->btnReciprocity->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 14.25F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(238)));
this->btnReciprocity->Location = System::Drawing::Point(536, 374);
this->btnReciprocity->Name = L"btnReciprocity";
this->btnReciprocity->Size = System::Drawing::Size(67, 63);
this->btnReciprocity->TabIndex = 73;
this->btnReciprocity->Text = L"1/x";
this->btnReciprocity->UseVisualStyleBackColor = true;
this->btnReciprocity->Click += gcnew System::EventHandler(this, &ScientificMode::btnReciprocity_Click);
//
// btnCub
//
this->btnCub->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 14.25F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(238)));
this->btnCub->Location = System::Drawing::Point(536, 168);
this->btnCub->Name = L"btnCub";
```

```
this->btnCub->Size = System::Drawing::Size(67, 63);
this->btnCub->TabIndex = 72;
this->btnCub->Text = L"x^3";
this->btnCub->UseVisualStyleBackColor = true;
this->btnCub->Click += gcnew System::EventHandler(this, &ScientificMode::btnCub_Click);
//
// btnMod
//
this->btnMod->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 14.25F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(238)));
this->btnMod->Location = System::Drawing::Point(389, 375);
this->btnMod->Name = L"btnMod";
this->btnMod->Size = System::Drawing::Size(67, 63);
this->btnMod->TabIndex = 90;
this->btnMod->Text = L"Mod";
this->btnMod->UseVisualStyleBackColor = true;
this->btnMod->Click += gcnew System::EventHandler(this, &ScientificMode::btnAritmeticOp_Click);
//
// btnOct
//
this->btnOct->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 14.25F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(238)));
this->btnOct->Location = System::Drawing::Point(462, 375);
this->btnOct->Name = L"btnOct";
this->btnOct->Size = System::Drawing::Size(67, 63);
this->btnOct->TabIndex = 89;
this->btnOct->Text = L"Oct";
this->btnOct->UseVisualStyleBackColor = true;
this->btnOct->Click += gcnew System::EventHandler(this, &ScientificMode::btnOct_Click);
//
// btnPercent
//
```

```
this->btnPercent->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 14.25F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(238)));
this->btnPercent->Location = System::Drawing::Point(315, 375);
this->btnPercent->Name = L"btnPercent";
this->btnPercent->Size = System::Drawing::Size(67, 63);
this->btnPercent->TabIndex = 88;
this->btnPercent->Text = L"%";
this->btnPercent->UseVisualStyleBackColor = true;
this->btnPercent->Click += gcnew System::EventHandler(this, &ScientificMode::btnPercent_Click);
//
// btnPow
//
this->btnPow->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 14.25F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(238)));
this->btnPow->Location = System::Drawing::Point(536, 237);
this->btnPow->Name = L"btnPow";
this->btnPow->Size = System::Drawing::Size(67, 63);
this->btnPow->TabIndex = 87;
this->btnPow->Text = L"Pow";
this->btnPow->UseVisualStyleBackColor = true;
this->btnPow->Click += gcnew System::EventHandler(this, &ScientificMode::btnAritmeticOp_Click);
//
// btnTan
//
this->btnTan->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 14.25F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(238)));
this->btnTan->Location = System::Drawing::Point(388, 306);
this->btnTan->Name = L"btnTan";
this->btnTan->Size = System::Drawing::Size(67, 63);
this->btnTan->TabIndex = 86;
this->btnTan->Text = L"Tan";
this->btnTan->UseVisualStyleBackColor = true;
```

```
this->btnTan->Click += gcnew System::EventHandler(this, &ScientificMode::btnTan_Click);
//
// btnHex
//
this->btnHex->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 14.25F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(238)));
this->btnHex->Location = System::Drawing::Point(461, 306);
this->btnHex->Name = L"btnHex";
this->btnHex->Size = System::Drawing::Size(67, 63);
this->btnHex->TabIndex = 85;
this->btnHex->Text = L"Hex";
this->btnHex->UseVisualStyleBackColor = true;
this->btnHex->Click += gcnew System::EventHandler(this, &ScientificMode::btnHex_Click);
//
// btnLn
//
this->btnLn->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 14.25F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(238)));
this->btnLn->Location = System::Drawing::Point(462, 99);
this->btnLn->Name = L"btnLn";
this->btnLn->Size = System::Drawing::Size(67, 63);
this->btnLn->TabIndex = 84;
this->btnLn->Text = L"ln";
this->btnLn->UseVisualStyleBackColor = true;
this->btnLn->Click += gcnew System::EventHandler(this, &ScientificMode::btnLn_Click);
//
// btnTanh
//
this->btnTanh->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 14.25F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(238)));
this->btnTanh->Location = System::Drawing::Point(315, 306);
this->btnTanh->Name = L"btnTanh";
```

```
this->btnTanh->Size = System::Drawing::Size(67, 63);
this->btnTanh->TabIndex = 83;
this->btnTanh->Text = L"Tanh";
this->btnTanh->UseVisualStyleBackColor = true;
this->btnTanh->Click += gcnew System::EventHandler(this, &ScientificMode::btnTanh_Click);
//
// btnCos
//
this->btnCos->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 14.25F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(238)));
this->btnCos->Location = System::Drawing::Point(388, 237);
this->btnCos->Name = L"btnCos";
this->btnCos->Size = System::Drawing::Size(67, 63);
this->btnCos->TabIndex = 82;
this->btnCos->Text = L"Cos";
this->btnCos->UseVisualStyleBackColor = true;
this->btnCos->Click += gcnew System::EventHandler(this, &ScientificMode::btnCos_Click);
//
// btnBin
//
this->btnBin->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 14.25F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(238)));
this->btnBin->Location = System::Drawing::Point(461, 237);
this->btnBin->Name = L"btnBin";
this->btnBin->Size = System::Drawing::Size(67, 63);
this->btnBin->TabIndex = 81;
this->btnBin->Text = L"Bin";
this->btnBin->UseVisualStyleBackColor = true;
this->btnBin->Click += gcnew System::EventHandler(this, &ScientificMode::btnBin_Click);
//
// btnLog
//
```

```
this->btnLog->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 14.25F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(238)));
this->btnLog->Location = System::Drawing::Point(389, 99);
this->btnLog->Name = L"btnLog";
this->btnLog->Size = System::Drawing::Size(67, 63);
this->btnLog->TabIndex = 80;
this->btnLog->Text = L"Log";
this->btnLog->UseVisualStyleBackColor = true;
this->btnLog->Click += gcnew System::EventHandler(this, &ScientificMode::btnLog_Click);
//
// btnCosh
//
this->btnCosh->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 14.25F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(238)));
this->btnCosh->Location = System::Drawing::Point(315, 237);
this->btnCosh->Name = L"btnCosh";
this->btnCosh->Size = System::Drawing::Size(67, 63);
this->btnCosh->TabIndex = 79;
this->btnCosh->Text = L"Cosh";
this->btnCosh->UseVisualStyleBackColor = true;
this->btnCosh->Click += gcnew System::EventHandler(this, &ScientificMode::btnCosh_Click);
//
// btnSin
//
this->btnSin->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 14.25F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(238)));
this->btnSin->Location = System::Drawing::Point(389, 168);
this->btnSin->Name = L"btnSin";
this->btnSin->Size = System::Drawing::Size(67, 63);
this->btnSin->TabIndex = 78;
this->btnSin->Text = L"Sin";
this->btnSin->UseVisualStyleBackColor = true;
```

```
this->btnSin->Click += gcnew System::EventHandler(this, &ScientificMode::btnSin_Click);
//
// btnFactorial
//
this->btnFactorial->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 14.25F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(238)));
this->btnFactorial->Location = System::Drawing::Point(536, 307);
this->btnFactorial->Name = L"btnFactorial";
this->btnFactorial->Size = System::Drawing::Size(67, 63);
this->btnFactorial->TabIndex = 77;
this->btnFactorial->Text = L"x!";
this->btnFactorial->UseVisualStyleBackColor = true;
this->btnFactorial->Click += gcnew System::EventHandler(this, &ScientificMode::btnFactorial_Click);
//
// btnSqrt
//
this->btnSqrt->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 14.25F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(238)));
this->btnSqrt->Location = System::Drawing::Point(462, 168);
this->btnSqrt->Name = L"btnSqrt";
this->btnSqrt->Size = System::Drawing::Size(67, 63);
this->btnSqrt->TabIndex = 76;
this->btnSqrt->Text = L"Sqrt";
this->btnSqrt->UseVisualStyleBackColor = true;
this->btnSqrt->Click += gcnew System::EventHandler(this, &ScientificMode::btnSqrt_Click);
//
// btnSinh
//
this->btnSinh->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 14.25F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(238)));
this->btnSinh->Location = System::Drawing::Point(315, 168);
this->btnSinh->Name = L"btnSinh";
```

```
this->btnSinh->Size = System::Drawing::Size(67, 63);
this->btnSinh->TabIndex = 75;
this->btnSinh->Text = L"Sinh";
this->btnSinh->UseVisualStyleBackColor = true;
this->btnSinh->Click += gcnew System::EventHandler(this, &ScientificMode::btnSinh_Click);
//
// btnQuadrate
//
this->btnQuadrate->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 14.25F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(238)));
this->btnQuadrate->Location = System::Drawing::Point(535, 100);
this->btnQuadrate->Name = L"btnQuadrate";
this->btnQuadrate->Size = System::Drawing::Size(67, 63);
this->btnQuadrate->TabIndex = 74;
this->btnQuadrate->Text = L"x^2";
this->btnQuadrate->UseVisualStyleBackColor = true;
this->btnQuadrate->Click += gcnew System::EventHandler(this, &ScientificMode::btnQuadrate_Click);
//
// btnPI
//
this->btnPI->Font = (gcnew System::Drawing::Font(L"Centaur", 20.25F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(0)));
this->btnPI->Location = System::Drawing::Point(316, 99);
this->btnPI->Name = L"btnPI";
this->btnPI->Size = System::Drawing::Size(67, 63);
this->btnPI->TabIndex = 71;
this->btnPI->Text = L"π";
this->btnPI->UseVisualStyleBackColor = true;
this->btnPI->Click += gcnew System::EventHandler(this, &ScientificMode::btnPI_Click);
//
// ScientificMode
//
this->AutoScaleDimensions = System::Drawing::SizeF(6, 13);
```

```
this->AutoScaleMode = System::Windows::Forms::AutoScaleMode::Font;
this->ClientSize = System::Drawing::Size(666, 454);
this->Controls->Add(this->btnReciprocity);
this->Controls->Add(this->btnCub);
this->Controls->Add(this->btnMod);
this->Controls->Add(this->btnOct);
this->Controls->Add(this->btnPercent);
this->Controls->Add(this->btnPow);
this->Controls->Add(this->btnTan);
this->Controls->Add(this->btnHex);
this->Controls->Add(this->btnLn);
this->Controls->Add(this->btnTanh);
this->Controls->Add(this->btnCos);
this->Controls->Add(this->btnBin);
this->Controls->Add(this->btnLog);
this->Controls->Add(this->btnCosh);
this->Controls->Add(this->btnSin);
this->Controls->Add(this->btnFactorial);
this->Controls->Add(this->btnSqrt);
this->Controls->Add(this->btnSinh);
this->Controls->Add(this->btnQuadrate);
this->Controls->Add(this->btnPI);
this->Name = L"ScientificMode";
this->Text = L"ScientificMode";
this->Load += gcnew System::EventHandler(this, &ScientificMode::scientificMode_Load);
this->ResumeLayout(false);
}
#pragma endregion
private: System::Void scientificMode_Load(System::Object^ sender, System::EventArgs^ e) {
this->Width = 633;
this->Height = 485;
historyList->Width = 593;
display->Width = 593;
}

// Button PI
```

```
private: System::Void btnPI_Click(System::Object^ sender, System::EventArgs^ e) {
lblShow->Text = "PI";
historyList->Items->Add("3.14159265358979323846264338327 95");
display->Text = ("3.14159265358979323846264338327 95");
}

// Button Log
private: System::Void btnLog_Click(System::Object^ sender, System::EventArgs^ e)
{
a = Double::Parse(display->Text);
lblShow->Text = System::Convert::ToString("log10 " + "(" + (display->Text) + ")");
historyList->Items->Add(lblShow->Text);
a = Math::Log10(a);
display->Text = System::Convert::ToString(a);
}

// Button ln
private: System::Void btnLn_Click(System::Object^ sender, System::EventArgs^ e) {
a = System::Math::Log(Convert::ToDouble(display->Text));
lblShow->Text = System::Convert::ToString("ln " + "(" + (display->Text) + ")");
historyList->Items->Add(lblShow->Text);
display->Text = Convert::ToString(a);
}

// Button x^2
private: System::Void btnQuadrate_Click(System::Object^ sender,
System::EventArgs^ e) {
lblShow->Text = display->Text + "^2";
historyList->Items->Add(lblShow->Text);
a = Convert::ToDouble(display->Text) * Convert::ToDouble(display->Text);
display->Text = Convert::ToString(a);
}

// Button Sinh
private: System::Void btnSinh_Click(System::Object^ sender, System::EventArgs^ e)
{
a = Double::Parse(display->Text);
lblShow->Text = System::Convert::ToString("Sinh " + "(" + (display->Text) + ")");
```

```
historyList->Items->Add(lblShow->Text);
a = Math::Sinh(a);
display->Text = System::Convert::ToString(a);
}

// Button Sin
private: System::Void btnSin_Click(System::Object^ sender, System::EventArgs^ e)
{
a = Double::Parse(display->Text);
lblShow->Text = System::Convert::ToString("Sinr " + "(" + (display->Text) + ")");
historyList->Items->Add(lblShow->Text);
a = Math::Sin(a);
display->Text = System::Convert::ToString(a);
}

// Button Sqrt
private: System::Void btnSqrt_Click(System::Object^ sender, System::EventArgs^ e)
{
a = Double::Parse(display->Text);
lblShow->Text = System::Convert::ToString("Sqrt " + "(" + (display->Text) + ")");
historyList->Items->Add(lblShow->Text);
a = Math::Sqrt(a);
display->Text = System::Convert::ToString(a);
}

// Button x^3
private: System::Void btnCub_Click(System::Object^ sender, System::EventArgs^ e)
{
lblShow->Text = display->Text + "^3";
historyList->Items->Add(lblShow->Text);
a = Convert::ToDouble(display->Text) * Convert::ToDouble(display->Text) *
Convert::ToDouble(display->Text);
display->Text = Convert::ToString(a);
}

// Button Cosh
private: System::Void btnCosh_Click(System::Object^ sender, System::EventArgs^
e) {
```

```
a = Double::Parse(display->Text);
lblShow->Text = System::Convert::ToString("Cosh " + "(" + (display->Text) + ")");
historyList->Items->Add(lblShow->Text);
a = Math::Cosh(a);
display->Text = System::Convert::ToString(a);
}

// Button Cos
private: System::Void btnCos_Click(System::Object^ sender, System::EventArgs^ e)
{
a = Double::Parse(display->Text);
lblShow->Text = System::Convert::ToString("Cosr " + "(" + (display->Text) + ")");
historyList->Items->Add(lblShow->Text);
a = Math::Cos(a);
display->Text = System::Convert::ToString(a);
}

// Button Bin
private: System::Void btnBin_Click(System::Object^ sender, System::EventArgs^ e)
{
lblShow->Text = System::Convert::ToString("Bin " + "(" + (display->Text) + ")");
historyList->Items->Add(lblShow->Text);
int a = (int)(Double::Parse(display->Text));
display->Text = System::Convert::ToString(a, 2);
}

// Button Tanh
private: System::Void btnTanh_Click(System::Object^ sender, System::EventArgs^ e) {
a = Double::Parse(display->Text);
lblShow->Text = System::Convert::ToString("Tanh " + "(" + (display->Text) + ")");
historyList->Items->Add(lblShow->Text);
a = Math::Tanh(a);
display->Text = System::Convert::ToString(a);
}

// Button Tan
```

```
private: System::Void btnTan_Click(System::Object^ sender, System::EventArgs^ e)
{
a = Double::Parse(display->Text);
lblShow->Text = System::Convert::ToString("Tanr " + "(" + (display->Text) + ")");
historyList->Items->Add(lblShow->Text);
a = Math::Tan(a);
display->Text = System::Convert::ToString(a);
}

// Button Hex
private: System::Void btnHex_Click(System::Object^ sender, System::EventArgs^ e)
{
lblShow->Text = System::Convert::ToString("Hex " + "(" + (display->Text) + ")");
historyList->Items->Add(lblShow->Text);
int a = (int)(Double::Parse(display->Text));
display->Text = System::Convert::ToString(a, 16);
}

// Button x!
private: System::Void btnFactorial_Click(System::Object^ sender,
System::EventArgs^ e) {
double fact = Double::Parse(display->Text);
lblShow->Text = System::Convert::ToString(fact) + "!";
historyList->Items->Add(System::Convert::ToString(fact) + "!");
for (int i = fact - 1; i > 0; i--) {
fact = fact * double(i);
}
display->Text = System::Convert::ToString(fact);
}

// Button %
private: System::Void btnPercent_Click(System::Object^ sender, System::EventArgs^
e) {
lblShow->Text = (display->Text) + "%";
historyList->Items->Add(lblShow->Text);
a = Convert::ToDouble(display->Text) / Convert::ToDouble(100);
display->Text = Convert::ToString(a);
}
```

```
// Button Oct
private: System::Void btnOct_Click(System::Object^ sender, System::EventArgs^ e)
{
lblShow->Text = System::Convert::ToString("Oct " + "(" + (display->Text) + ")");
historyList->Items->Add(lblShow->Text);
int a = (int)(Double::Parse(display->Text));
display->Text = System::Convert::ToString(a, 8);
}

// Button 1/x
private: System::Void btnReciprocity_Click(System::Object^ sender, System::EventArgs^ e) {
lblShow->Text = "1/" + (display->Text);
historyList->Items->Add(lblShow->Text);
a = Convert::ToDouble(1.0 / Convert::ToDouble(display->Text));
display->Text = Convert::ToString(a);
}
};
}
```

2.7 Korišćenje aplikacije

Nakon poktretanja aplikacije korisnik treba da izabere jedan od četiri ponuđena režima, a to su:

1. standardni režim - omogućava korišćenje osnovnih operacija prilikom zadavanja izraza;
2. naučni režim - omogućava korišćenje osnovnih i naprednih operacija prilikom zadavanja izraza;
3. opisni režim – prikazuje neke osnovne informacije kalkulatora;
4. izlazni režim – omogućava prekidanje rada aplikacije kalkulatora.

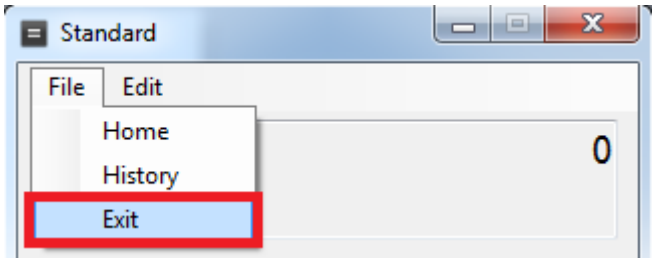
Nakon biranja jednog od sledeća dva režima: standardni ili naučni režim, korisnik je u mogućnosti da zadaje pomoću komponente računara(miša) ulaze tj. izraze za koje želi da mu aplikacija odradi izračunavanje.

Kada korisnik završi sa zadavanjem izraza, on može da pritisne znak jednakosti, nakon čega mu aplikacija prikazuje rezultat.

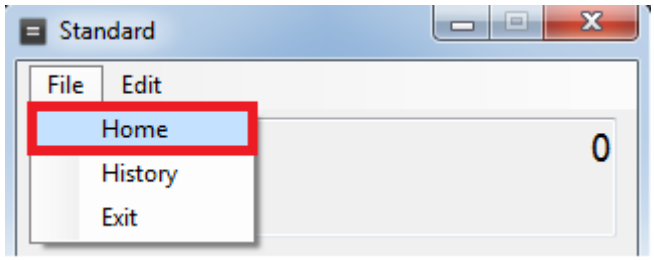
Pored zadavanja izraza, korisnik je u mogućnosti da uredjuje izraz.

Kada korisnik završi sa svojim potrebama, on može da zatvori aplikaciju na neki od sledećih načina:

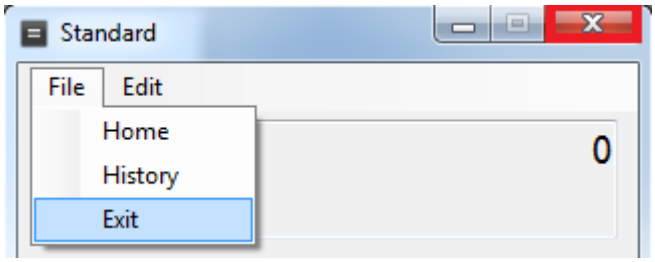
- klikom na padajući meni koji se naziva "File", biranje opcije iz liste koja se naziva "Exit" (slika 1) kojom direktno izlazi iz nje;
- klikom na padajući meni koji se naziva "File", biranje opcije iz liste koja se naziva "Home" (slika 2) kojom se prikazuje Home režima, zatim biranje režima koji se naziva "Exit", kojim se direktno izlazi iz nje;
- klikom na dugme aplikacije X (slika 3) kojim se direktno izlazi iz nje.



Slika 1 Prvi način izlaženja iz aplikacije



Slika 2 Drugi način izlaženja iz aplikacije



Slika 3 Treći način izlaženja iz aplikacije

3 UML dijagrami

The Unified Modeling Language ili skraćeno UML je standardni grafički jezik za modelovanje objektno-orjentisanog softvera.

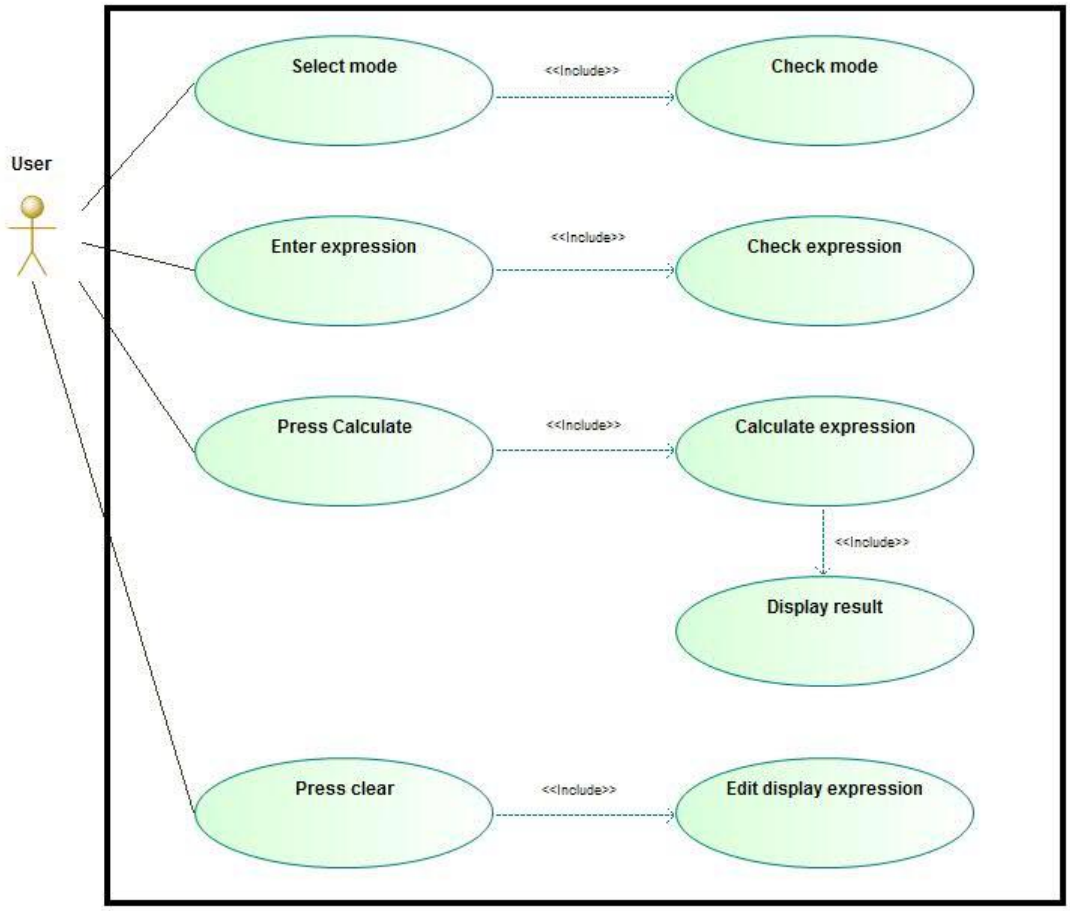
Zbog toga što je UML bogat jezik koji obuhvata velik broj dijagrama biće prikazani samo dijagrami koji se najčešće koriste u praksi, a to su:

- dijagram slučajeva korišćenja;
- dijagram klasa;
- dijagram sekvenci;
- dijagram aktivnosti;
- dijagram stanja.

3.1 Dijagram slučajeva korišćenja

Dijagram slučajeva korišćenja (engl. use case diagram) prikaz je interakcije korisnika sa sistemom koji pokazuje odnos između korisnika i različitih slučajeva korišćenja u kojima je korisnik uključen.

Slučajevi korišćenja predstavljeni su krugovima ili elipsama, a korisnici(akteri) predstavljeni su čovečuljcima.



Slika 4 Dijagram slučajeva korišćenja

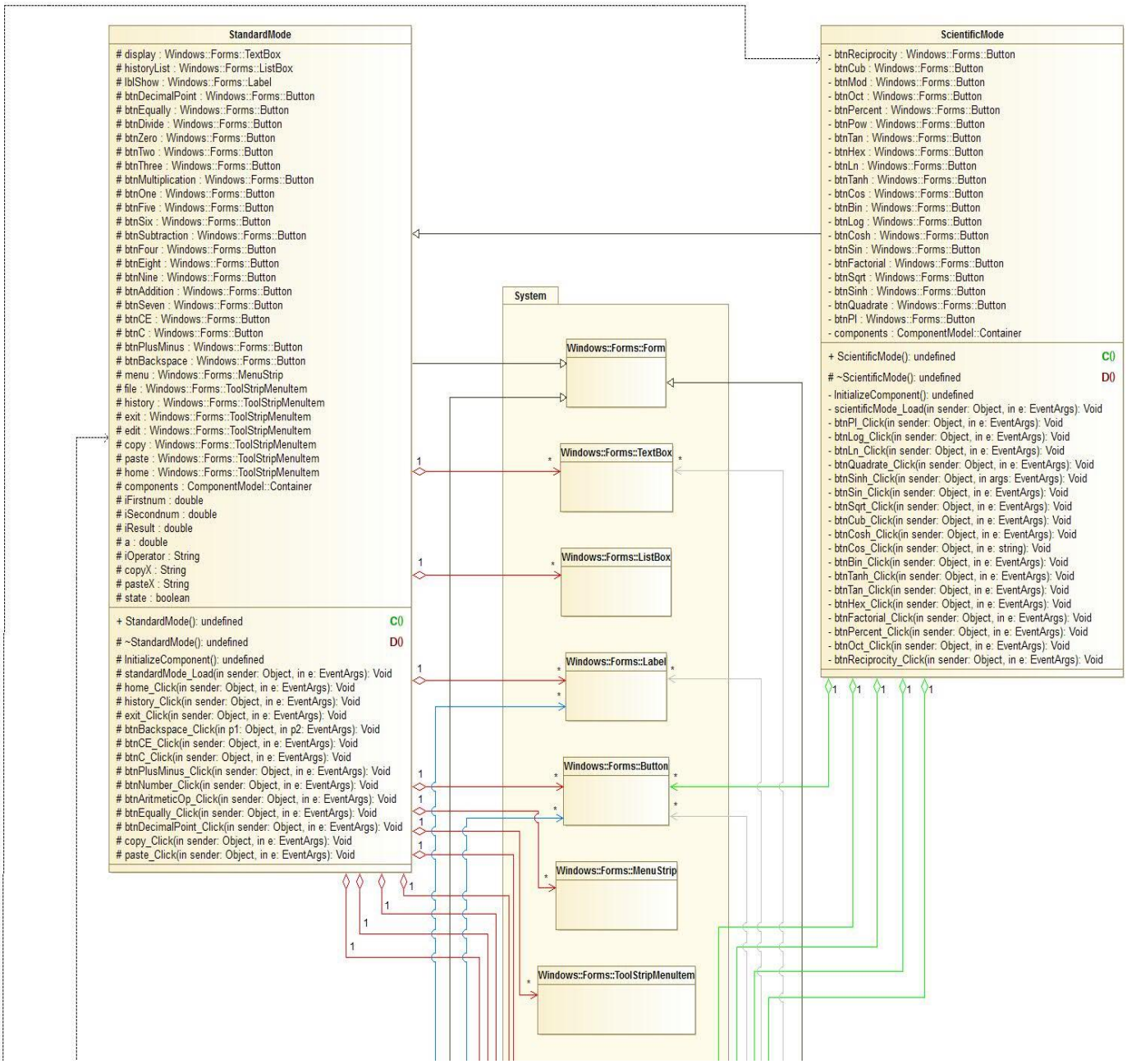
3.2 Dijagram klasa

Dijagram klasa (engl. class diagram) je vrsta strukturnog dijagrama u softverskom inžinjeringu, koji opisuje strukturu sastava objašnjavajući klase unutar sastava, njihove atribute i odnose.

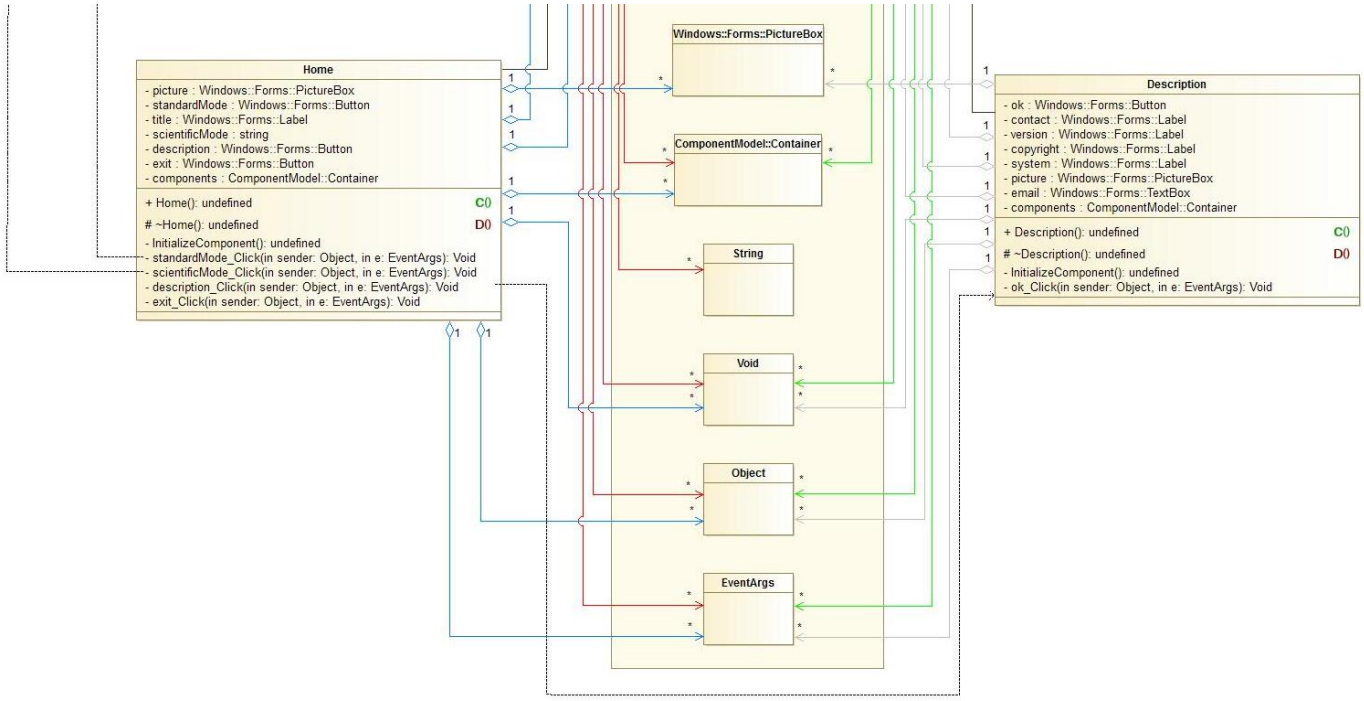
Elementi dijagrama klasa su:

- stvari: klasa, interfejsi, tipovi, izuzeci, šabloni, saradnje, paketi
- relacije: zavisnosti, generalizacije, asocijacije, realizacije

Simbol klase je pravougaonik podeljen horizontalnim linijama u odeljke(naziv klase, atributi, operacije, odgovornosti)



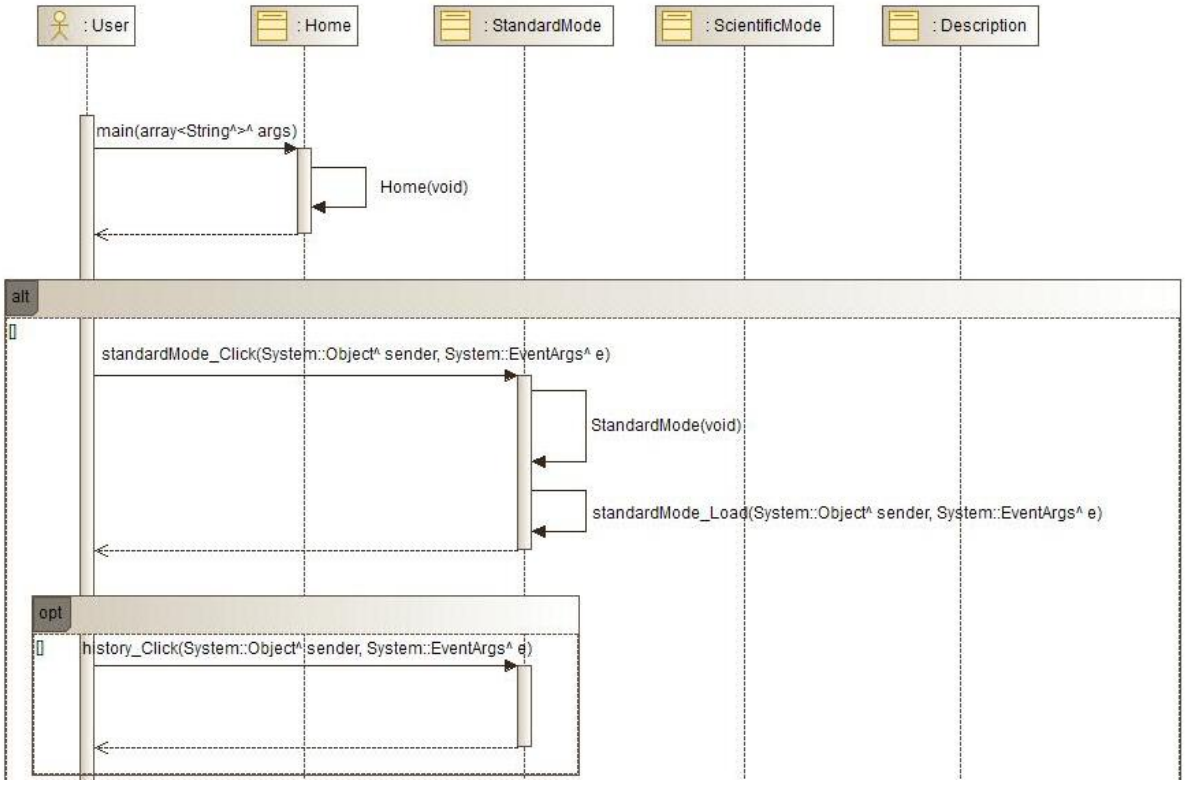
Slika 5 Dijagram klasa, prvi deo



Slika 6 Dijagram klasa, drugi deo

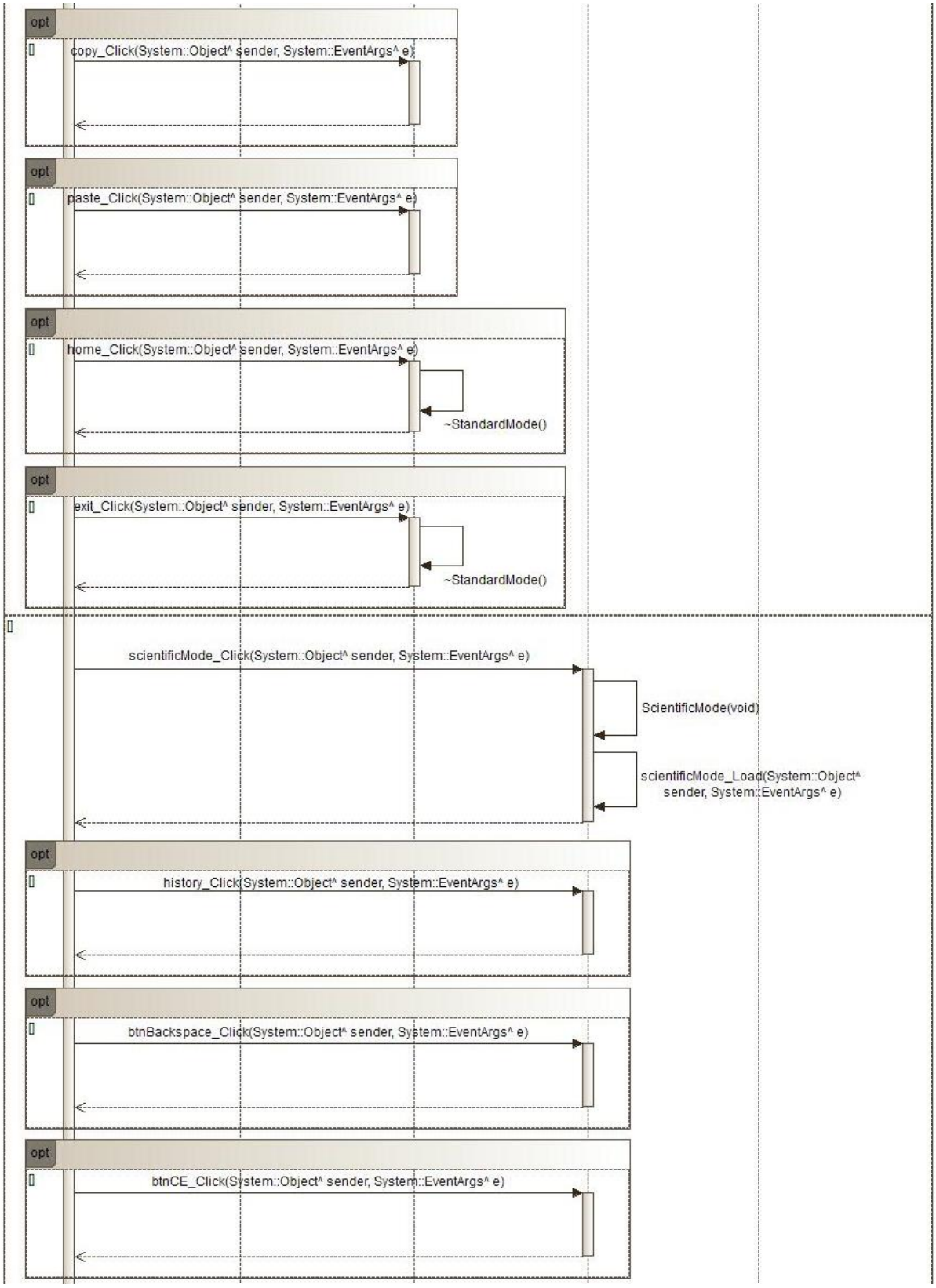
3.3 Dijagram sekvenci

Dijagram sekvenci (engl. sequence diagram) prikazuje komunikaciju između skupa objekata, koja se ostvaruje porukama koje objekti međusobno razmenjuju u cilju ostvarivanja očekivanog ponašanja. Dijagram sekvenci može da sadrži aktere, objekte i poruke.



Slika 7 Dijagram sekvenci, prvi deo

Slika 8 Dijagram sekvenci, drugi deo

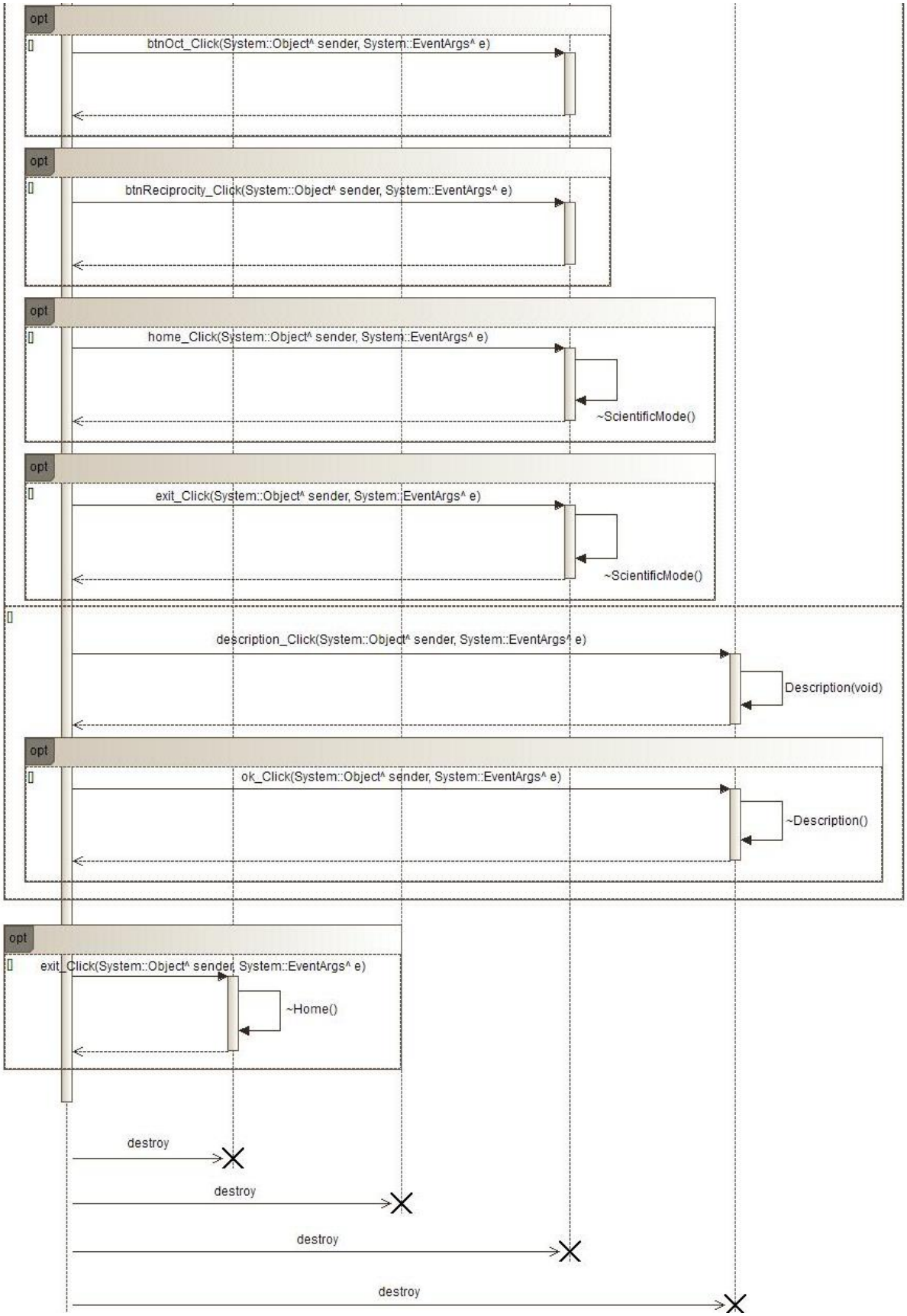


Slika 9 Dijagram sekvenci, treći deo

Slika 10 Dijagram sekvenci, četvrti deo

Slika 11 Dijagram sekvenci, peti deo

Slika 12 Dijagram sekvenci, šesti deo



Slika 13 Dijagram sekvenci, sedmi deo

3.4 Dijagram aktivnosti

Dijagrami aktivnosti (engl. activity diagram) su namenjeni modeliranju dinamičkih aspekata (ponašanja) sistema.

Dijagram aktivnosti prikazuje:

- tok aktivnosti koju izvršavaju objekti;
- eventualno i tok objekata između koraka aktivnosti.

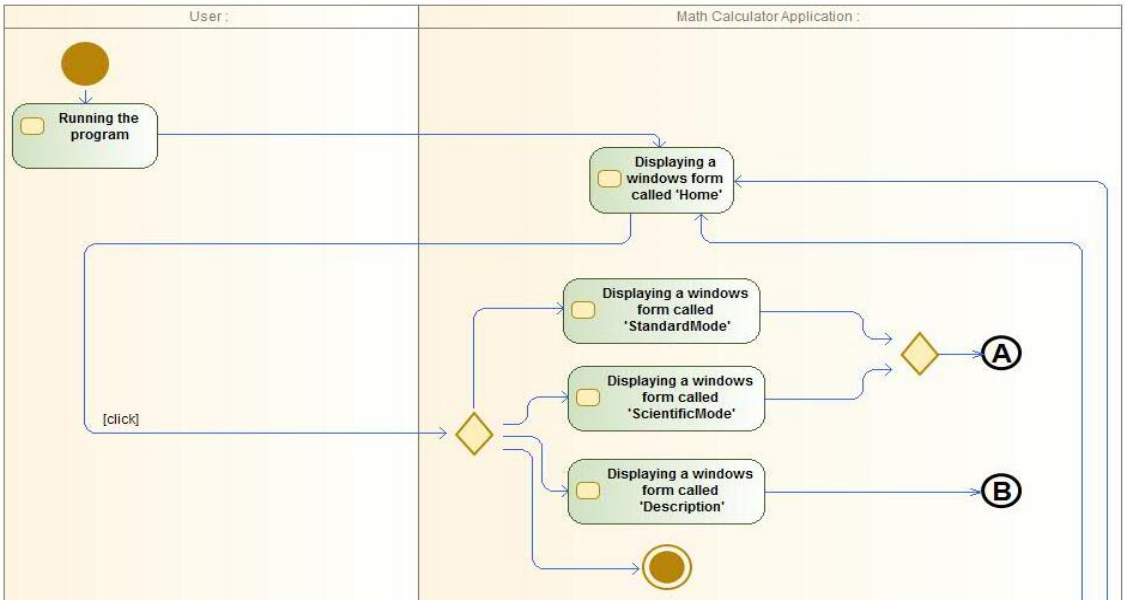
Aktivnost je specifikacija parametrizovanog ponašanja koje se izražava kroz tok izvršenja preko sekvenciranja i konkurisanja podaktivnosti. Dijagram aktivnosti je graf koji sadrži čvorove i grane.

Grane:

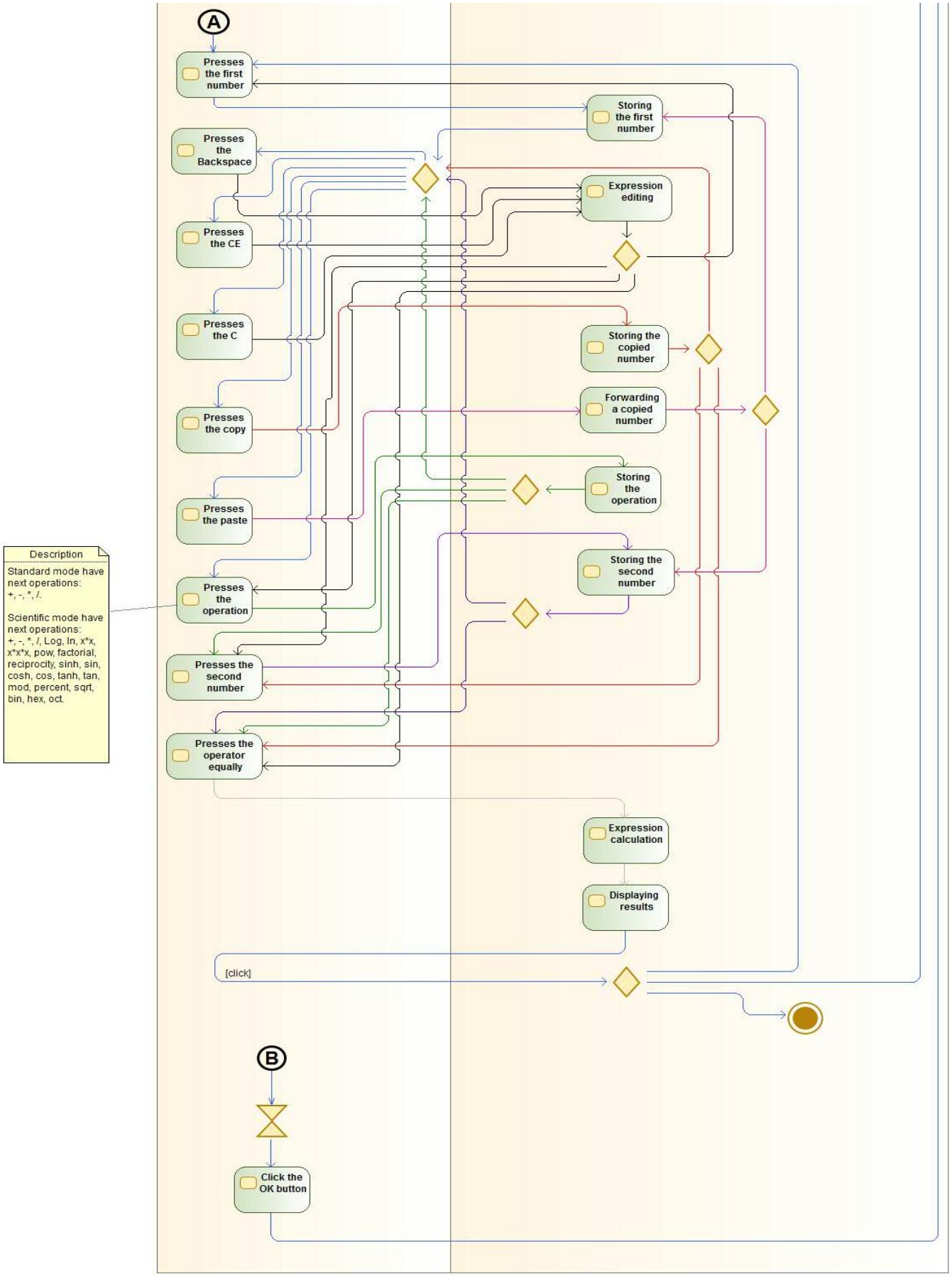
- prelazi (tranzicije) između akcija;
- tok objektač.

Čvorovi:

- akcije i aktivnosti;
- objekti;
- slanje signala (send signal) ;
- prihvatanja događaja (accept event) ;
- prihvatanja vremenskog događaja (accept time event) ;
- kontrolni čvorovi:
  - sekvencijalna grananja i spajanja u toku kontrole (decision i merge) ;
  - konkurentna grananja i spajanja u toku kontrole (fork i join) ;
- pseudočvorovi: početni, završni i kraj toka ;
- konektori.



Slika 14 Dijagram aktivnosti, prvi deo



Slika 15 Dijagram aktivnosti, drugi deo

3.5 Dijagram stanja

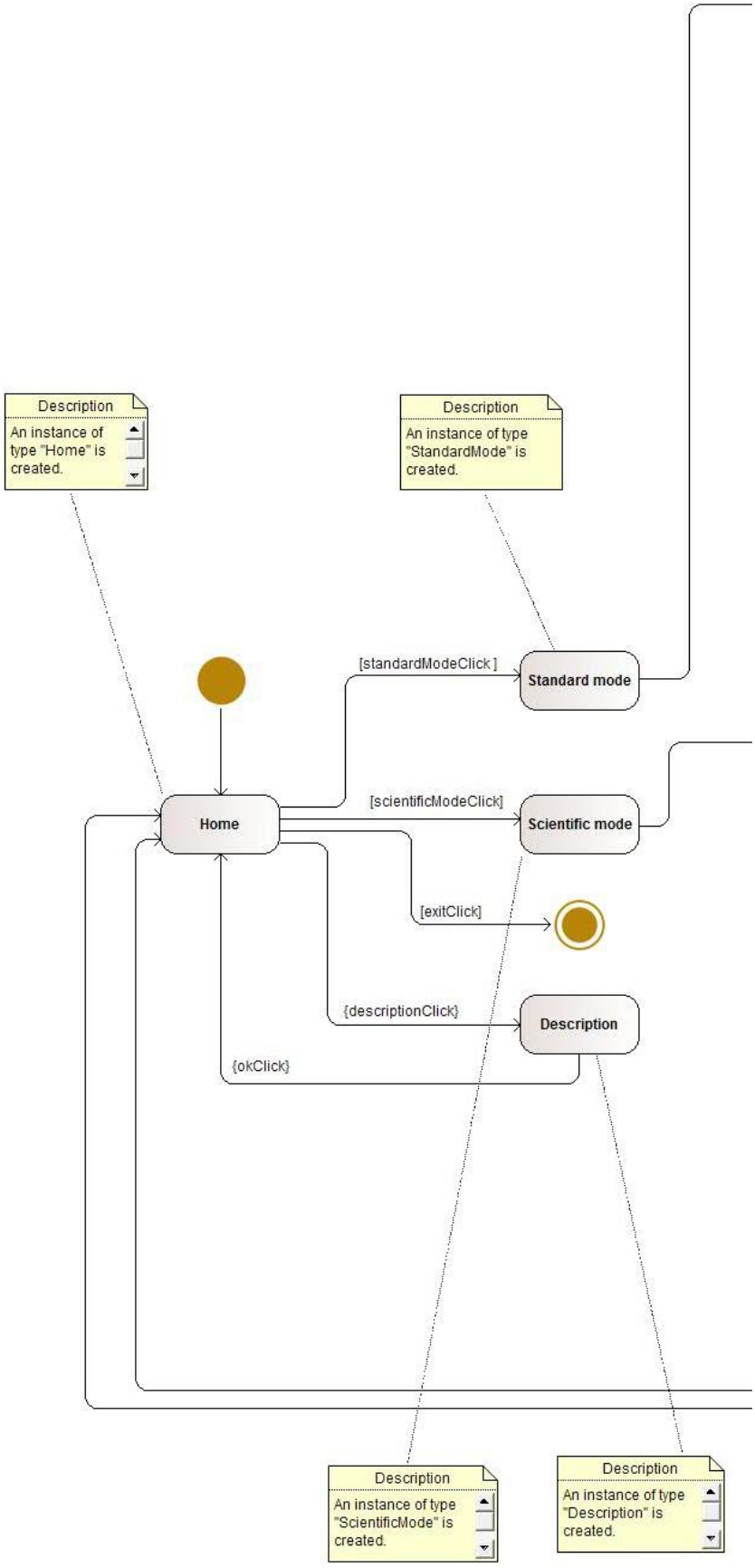
Dijagrami stanja (engl. state machine diagram) se koriste za opisivanje ponašanja sistema.

On može da opiše moguća stanja objekta kako se događaji pojavljuju.

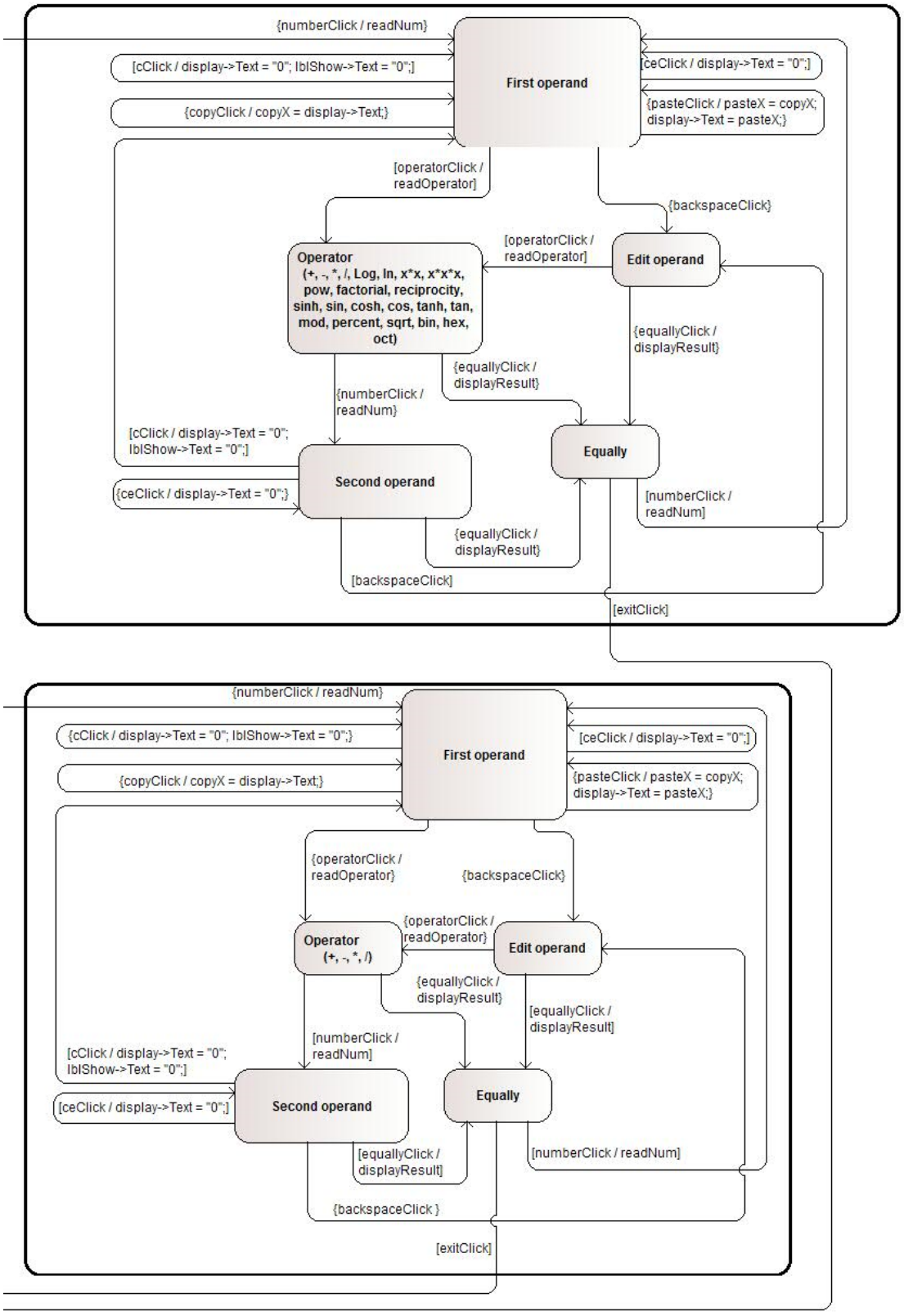
Svaki dijagram obično predstavlja objekte jedne klase i prati različita stanja tih objekata kroz sistem.

Dijagram stanja se može upotrebiti da grafički predstavi automate konačnih stanja.

Stanje se označava pravougaonikom sa zaobljenim ivicama.



Slika 16 Dijagram stanja, prvi deo



Slika 17 Dijagram stanja, drugi deo

4 Literatura

https://www.youtube.com/watch?v=rlW_k02TFpQ, 04.07.2020(10:23)

https://www.codeproject.com/Tips/1209869/How-to-Create-a-Windows-Form-Project-in-Visual-Stu, 04.07.2020 (10:41)

softverski inženjering, treća godina,računarska tehnika i softversko inženjerstvo, moodle portal: http://moodle.fink.rs, 05.10.2020 (21:55)