Федеральное государственное автономное

образовательное учреждение

высшего образования

«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт нефти и газа

Кафедра проектирования и эксплуатации газонефтепроводов

**ОТЧЕТ О ПРАКТИЧЕСКОЙ РАБОТЕ №5**

по прикладному программированию

«Графический интерфейс пользователя»

Преподаватель А. С. Сатышев

подпись, дата

Студент НБ18-07Б, 081831211 П. А. Кирсанов

подпись, дата

Красноярск 2020

**СОДЕРЖАНИЕ**

Введение 3

1 Основная часть 4

2 Результат работы программы 8

Заключение 10

**ВВЕДЕНИЕ**

В состав Matlab входит среда GUIDE для создания приложений с графическим интерфейсом пользователя.

Цель данной работы – ознакомиться со средой GUIDE для создания приложений с графическим интерфейсом пользователя, а также создать пользовательский графический интерфейс при помощи GUIDE.

В ходе выполнения данной практической работы буду использовать следующие команды:

- get(…) – запись свойства;

- set(…) ¬– чтение свойства;

- switch – используется для выбора того или иного варианта выполнения кода;

- case – вычисляет выражение и выбирает для выполнения одну из нескольких групп операторов.

Основные элементы «Guide»:

- property inspector – инспектор свойств;

- callback – функция обратного вызова;

- pushbutton – обычная кнопка;

- edit Text - поле ввода текста;

- popup Menu – выпадающий список;

- check Box – флажок (галочка);

- static Text – область ввода текста;

- Axes – оси графика.

Основные свойства графических объектов:

- tag – имя объекта;

- string – строка, для каждой из элементов имеет своё назначение;

- value – значение;

- background Color – цвет фона объекта;

- enabled/disable – состояние объекта (доступен/не доступен);

- Visible – видимость объекта.

**1 Основная часть**

function varargout = video3(varargin)

gui\_Singleton = 1;

gui\_State = struct('gui\_Name', mfilename, ...

 'gui\_Singleton', gui\_Singleton, ...

 'gui\_OpeningFcn', @video3\_OpeningFcn, ...

 'gui\_OutputFcn', @video3\_OutputFcn, ...

 'gui\_LayoutFcn', [] , ...

 'gui\_Callback', []);

if nargin && ischar(varargin{1})

 gui\_State.gui\_Callback = str2func(varargin{1});

end

if nargout

 [varargout{1:nargout}] = gui\_mainfcn(gui\_State, varargin{:});

else

 gui\_mainfcn(gui\_State, varargin{:});

end

%----------- Процедура срабатывающая при запуске программы ----------------

function video3\_OpeningFcn(hObject, ~, handles, varargin)

% Choose default command line output for video3

handles.output = hObject;

% Описание структуры хранящей данные о линии

% line.name - Имя линии, строка

% line.color - Цвет линии, строка (в нотации Matlab (r, g, b и т.д.))

% line.width - Толщина линии, число

% line.id - Идентификатор линии полученный от функции plot

% Массив ячеек хранящий структуры с описанием построенных линий

handles.lines = {};

% Update handles structure

guidata(hObject, handles);

%----------- Конец процедуры video3\_OpeningFcn ----------------------------

%----------- Процедура срабатывающая при изменении имени линии ------------

function lineName\_Callback(hObject, eventdata, handles)

%----------- Конец процедуры lineName\_Callback ----------------------------

%----------- Процедура срабатывающая при нажатии кнопки -------------------

function plotLine\_Callback(hObject, eventdata, handles)

k = -5 + (5+5)\*rand; % Получаем случайное число на промежутке (-5, 5)

b = -5 + (10+7)\*rand;

x = 0.1:0.1:10; % Область определения, можете земенить на свою

y = k\*sin(x/2)+b; % Формула линии, можете заменить на свою

axes(handles.charts);

hold on

Imyalinii = get(handles.lineName, 'string');

Shirinalinii = get(handles.lineWidth, 'Value');

Colorlinii = get(handles.lineColor, 'Value');

switch Colorlinii

 case 1

 Line1 = plot(x,y, 'b', 'DisplayName', Imyalinii);

 if Shirinalinii == 1

 set(Line1 ,'LineWidth', 2);

 elseif Shirinalinii == 2

 set(Line1 ,'LineWidth', 5);

 end

 case 2

 Line2 = plot(x,y, 'y', 'DisplayName', Imyalinii);

 if Shirinalinii == 1

 set(Line2 ,'LineWidth', 2);

 elseif Shirinalinii == 2

 set(Line2 ,'LineWidth', 5);

 end

 case 3

 Line3 = plot(x,y, 'c', 'DisplayName', Imyalinii);

 if Shirinalinii == 1

 set( Line3,'LineWidth', 2);

 elseif Shirinalinii == 2

 set( Line3,'LineWidth', 5);

 end

 case 4

 Line4= plot(x,y, 'm', 'DisplayName', Imyalinii);

 if Shirinalinii == 1

 set( Line4,'LineWidth', 2);

 elseif Shirinalinii == 2

 set( Line4,'LineWidth', 5);

 end

end

deli = get(handles.clearCharts,'Value');

if deli == 1

 cla

 else deli == 2

 hold off

end

Mn = get(handles.lineName, 'String');

str\_part = Mn;

old\_str = get(handles.listLines, 'String');

new\_str = strvcat(old\_str, str\_part);

set(handles.listLines, 'String', new\_str);

guidata(hObject,handles);

%----------- Конец процедуры plotLine\_Callback ----------------------------

%----------- Процедура срабатывающая при выборе толщины линии --------------

function lineWidth\_Callback(hObject, eventdata, handles)

%----------- Конец процедуры lineWidth\_Callback ---------------------------

%----------- Процедура срабатывающая при выборе цвета линии ---------------

function lineColor\_Callback(hObject, eventdata, handles)

%----------- Конец процедуры lineColor\_Callback ---------------------------

%----------- Процедуры срабатывающая при выбора "галочки" очистки графиков-

function clearCharts\_Callback(hObject, eventdata, handles)

%----------- Конец процедуры clearCharts\_Callback -------------------------

%----------- Процедура управляющая отображением легенды -------------------

function showLegend\_Callback(hObject, eventdata, handles)

legenda= get(hObject,'Value');

if legenda== 0

 legend off

else legenda==1

 legend show

end

guidata(hObject,handles);

%----------- Конец процедуры showLegend\_Callback --------------------------

%----------- Процедура срабатывающая при выборе элемента списка линий -----

function listLines\_Callback(hObject, eventdata, handles)

%----------- Конец процедуры listLines ------------------------------------

%----------- Процедура срабатывающая при клике в контекстном меню "Удалить"

function deleteLine\_Callback(hObject, eventdata, handles)

%----------- Конец процедуры deleteLine -----------------------------------

%----------- Процедура срабатывающая при открытии

% контекстного меню списка линий -------------------------------

function listLinesContext\_Callback(hObject, eventdata, handles)

%----------- Процедуры создания некоторых элементов интерфейса ------------

function lineColor\_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))

 set(hObject,'BackgroundColor','white');

end

function lineWidth\_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))

 set(hObject,'BackgroundColor','white');

end

function lineName\_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))

 set(hObject,'BackgroundColor','white');

end

function listLinesContext\_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))

 set(hObject,'BackgroundColor','white');

end

function listLines\_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))

 set(hObject,'BackgroundColor','white');

end

%----------- Конец процедур создания элементов

%----------- Процедура срабатывающая при закрытии программы ---------------

function varargout = video3\_OutputFcn(hObject, eventdata, handles)

varargout{1} = handles.output;

%----------- Конец выполнения программы

**2 Результаты работы программы**

Результат работы программы проиллюстрирован на рисунках:



Рисунок 1 – Результат



Рисунок 2 – Результат после нажатия «очищать оси»

**ЗАКЛЮЧЕНИЕ**

Эта работа далась мне тяжелее всего, я сразу понял, что нам требуется сделать, но не понимал каким образом. Но по итогу было интересно. Несколько раз я пересматривал всё, что связано с пятой работой и о чудо! Я наконец понял, как это воплотить в жизнь.

Хотелось бы поблагодарить вас, за то что не требуете от нас достать звезду с неба или покорить Эверест. За курс программирования я приобрёл массу знаний для работы в матлабе. Думаю, что когда-нибудь это ещё пригодиться.