

控制系统数字仿真第二次上机作业要求

一、函数编写练习

编写名为 **ZhiShu** 的函数完成如下功能：

1. 判断输入参数是否为**质数**，如果输入参数为质数，则在工作窗口输出一行字“此数为质数”，否则在工作窗口输出一行字“此数不是质数，可以被 XX 整除”，XX 为该数的**因数**。

2. 如果输入参数为偶数，编写并调用名为 **EvenNum** **内嵌函数**，在工作窗口输出一行字“此数为偶数，其平方根为 XXX”，XXX 为该数的平方根；

3. 用 **try catch end** 实现防输错功能，当用户输入的数据不是数值时，在命令窗口中输出“输入错误，请输入正整数!!”

二、流程控制语句练习

编写脚本文件，调用你所编写的 **ZhiShu** 函数，找出 0 到 100 内的全部质数，并将结果存储在名为 **PrimeNum** 的数组内。

二、画图练习

1. Plot 命令的练习

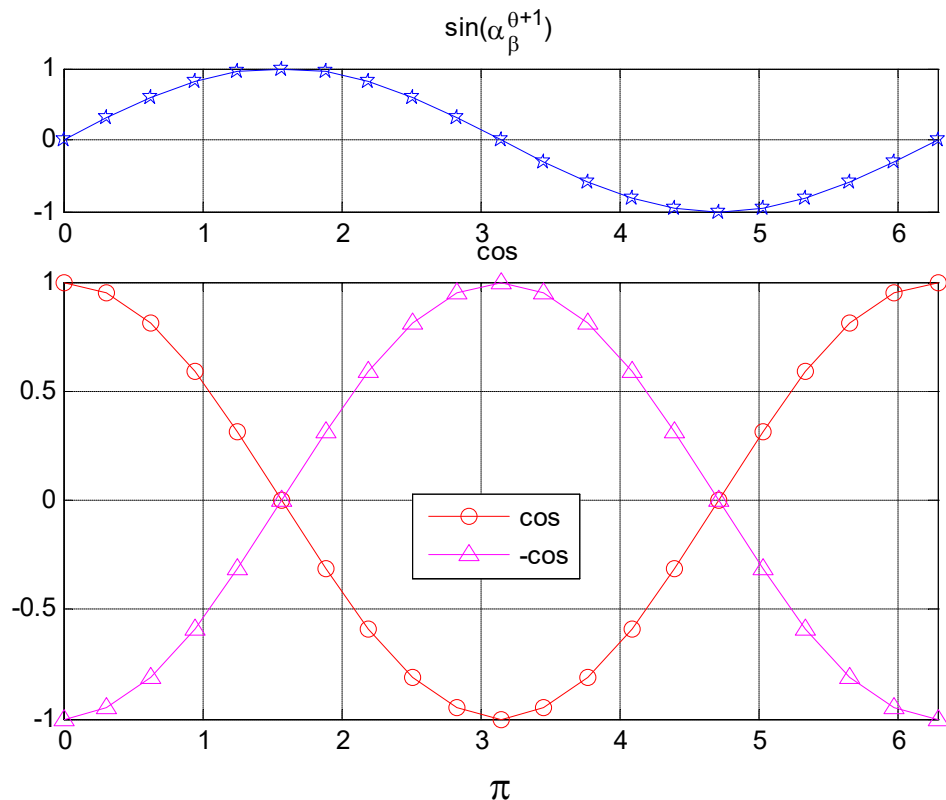


图 1

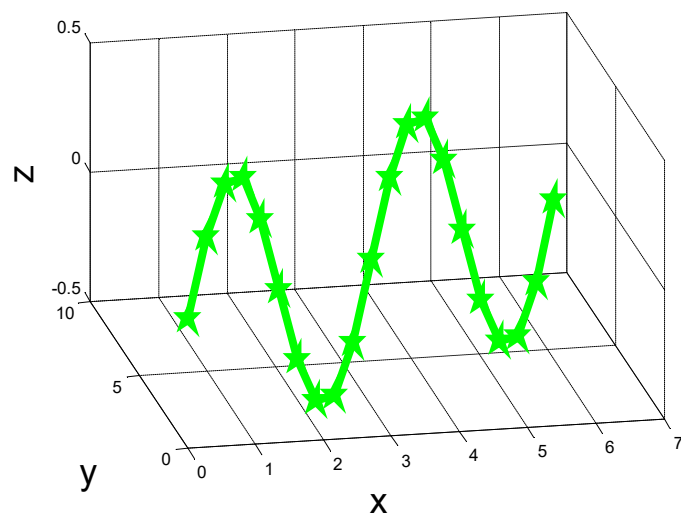
用 **help plot** 查看 **plot** 命令的说明，画出上图，要求：将窗口分成三行，第一行子图中画正弦曲线，第二、三行子图中画 **cos** 和 **-cos**,横坐标范围 $[0, 2\pi]$ ，线型、标注和注释如图所示。

提示：标注希腊字母 **xlabel('alpha')**,在工作空间中输入 **help plot**, 点击 [doc plot](#),在弹出的帮助文件中看 [String](#) 的链接，可以看到所有希腊字母的标注方式；

画图，若不指定画图窗口的句柄，就在当前的窗口画图。

当前窗口，是按照创建次序逐步更新的，如果不用鼠标选择，则最后一个窗口是当前窗口；如果用鼠标选择，则选中的窗口是当前窗口。

2. 三维曲线绘图练习



用 `plot3` 命令画出图 2 所示的三维曲线， x, y 的取值范围为 $[0, 2\pi]$ ， Z 是 $\sin(x)$ 和 $\cos(y)$ 的乘积，线型用绿实线加五星，线宽 5。

3. 三维曲面绘图

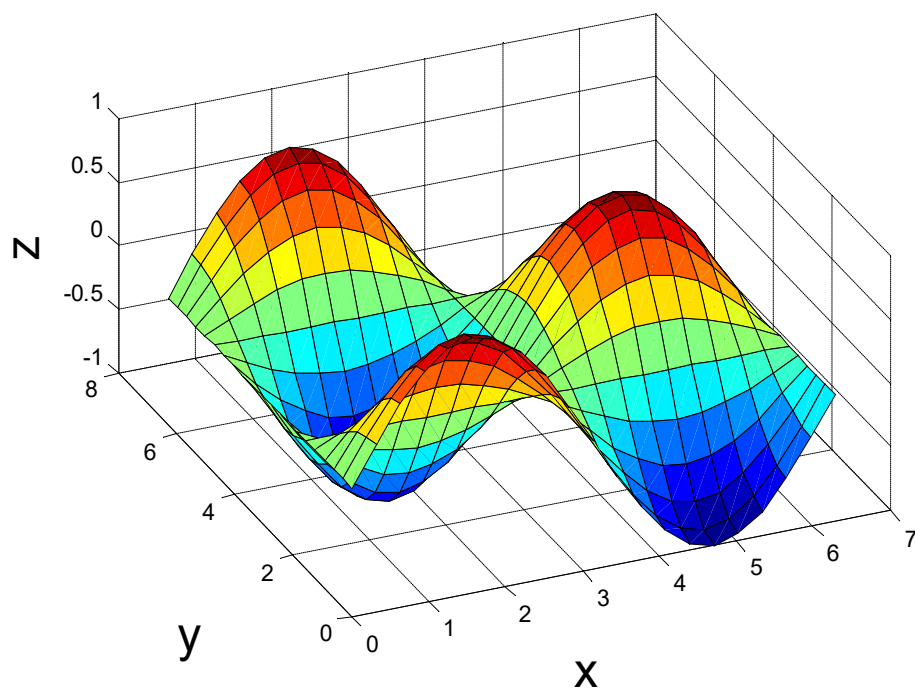


图 3

用 `surf` 命令画出图 3 所示的三维曲面， x, y 的取值范围为 $[0, 2\pi]$ ， Z 是 $\sin(x)$ 和 $\cos(y)$ 的乘积。

4. 填充绘图练习

用 `help` 查看 `area` 和 `fill` 命令的帮助文件，编写代码，画出图 4 和图 5，图中横坐标范围为 $[-2\pi, 2\pi]$ ，纵坐标为横坐标的正弦值。

体验 `area` 和 `fill` 的区别

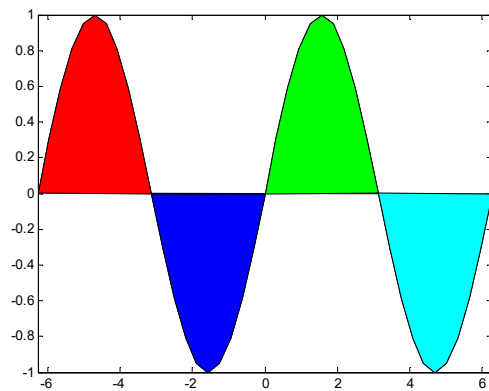


图 4

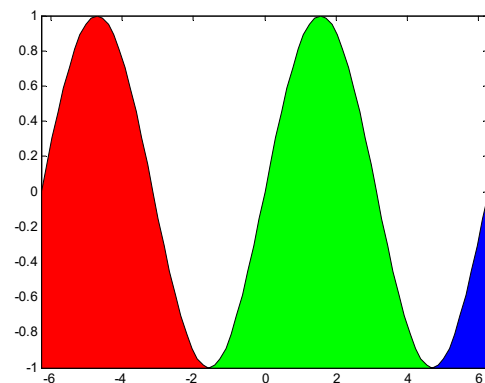


图 5

三、GUI 的练习

利用 GUIDE 做简单的界面，并设置回调函数

1. File->New-GUI, 打开 GUI 创建页面，如图 6 所示

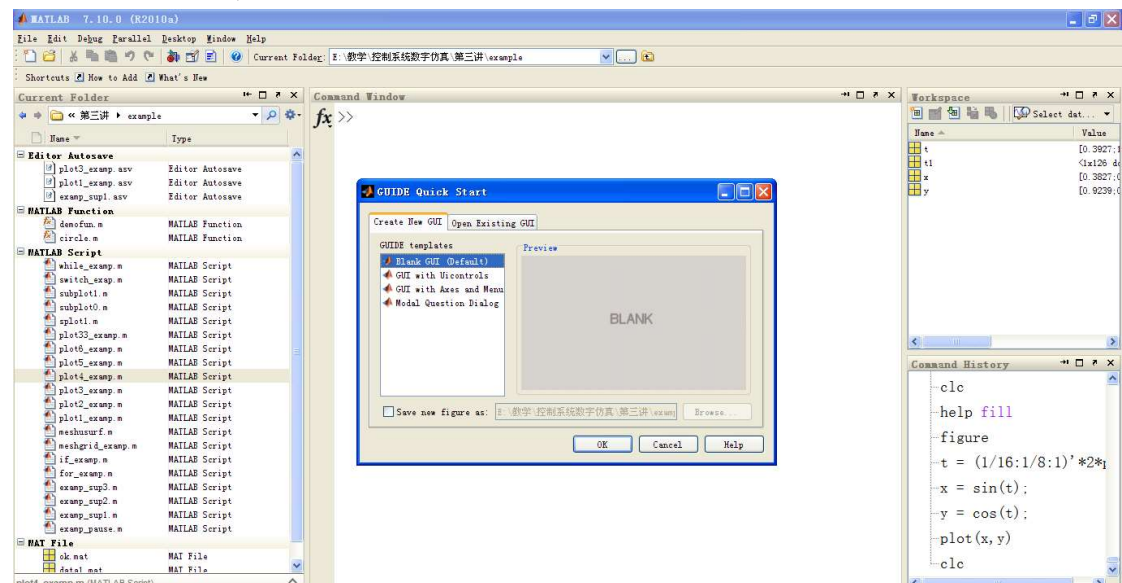


图 6 打开 GUIDE

2. 选择 “Blank GUI (Default)” 创建一个 GUI 画布

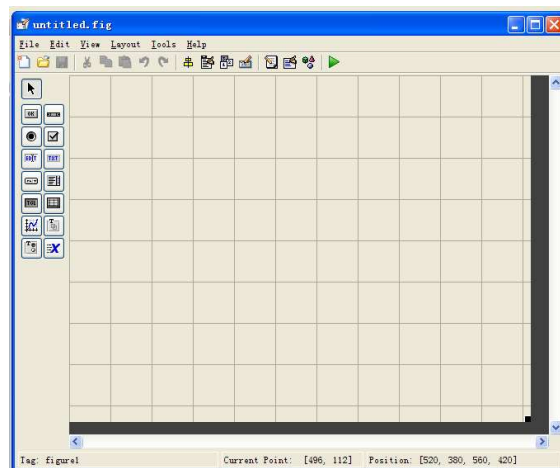


图 7GUIDE 画布

3. 在 GUI 窗体中拖放控件

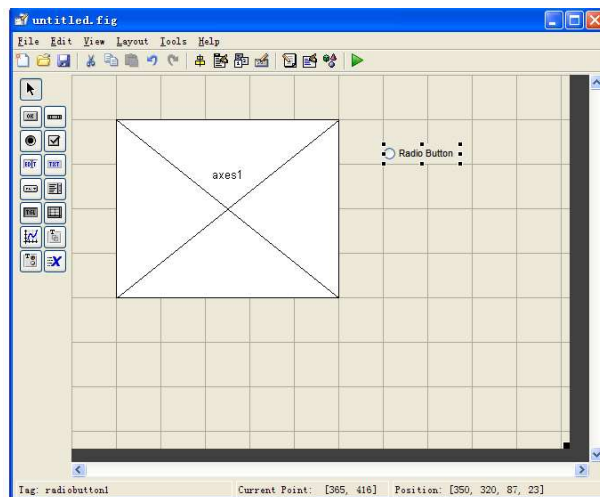


图 8 拖放控件

4. 点击保存图标，将该 GUI 命名为 GUI_temp2，如图 9 所示

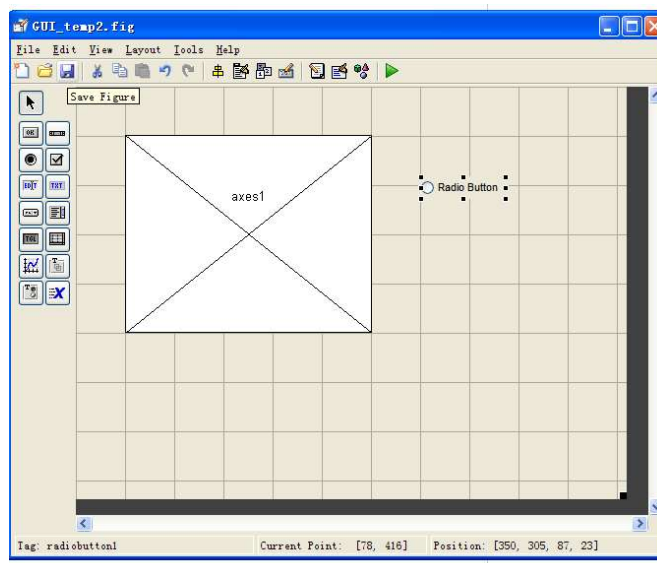


图 9

5. 双击“Radio Button”，弹出该控件的属性页，如图 9 所示

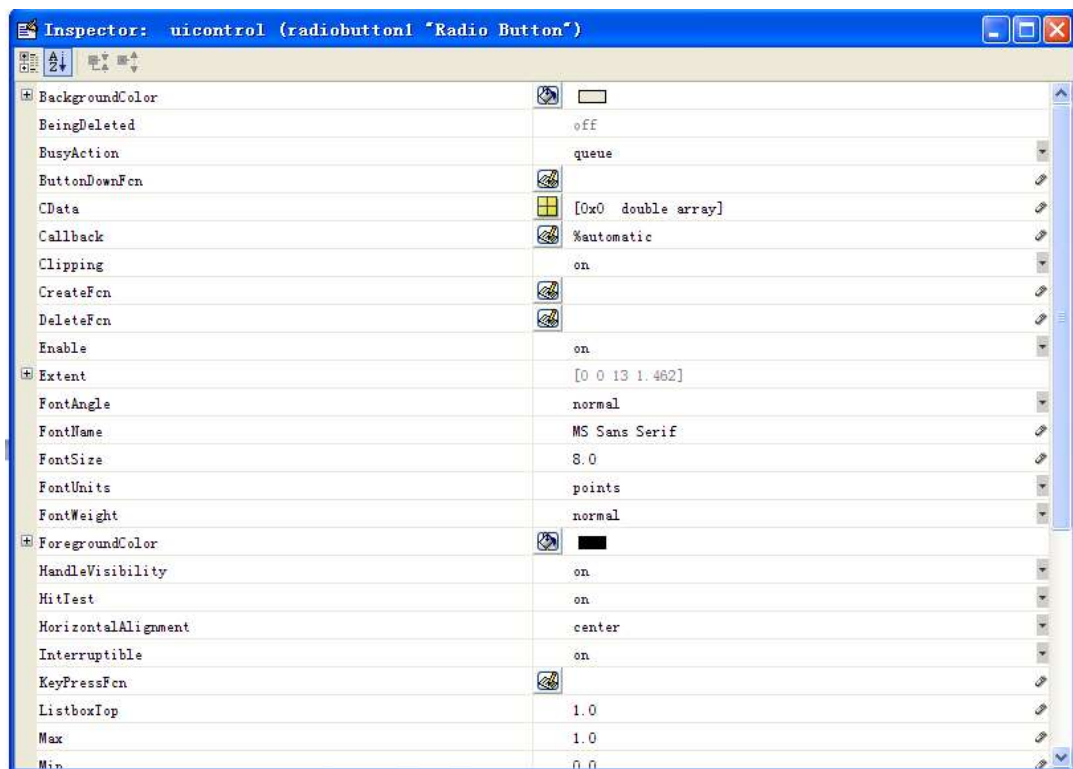



图 10 Radio Button 属性页

6. 双击“Callback”属性的  图标，弹出该 GUI 界面的模板文档，如图 11

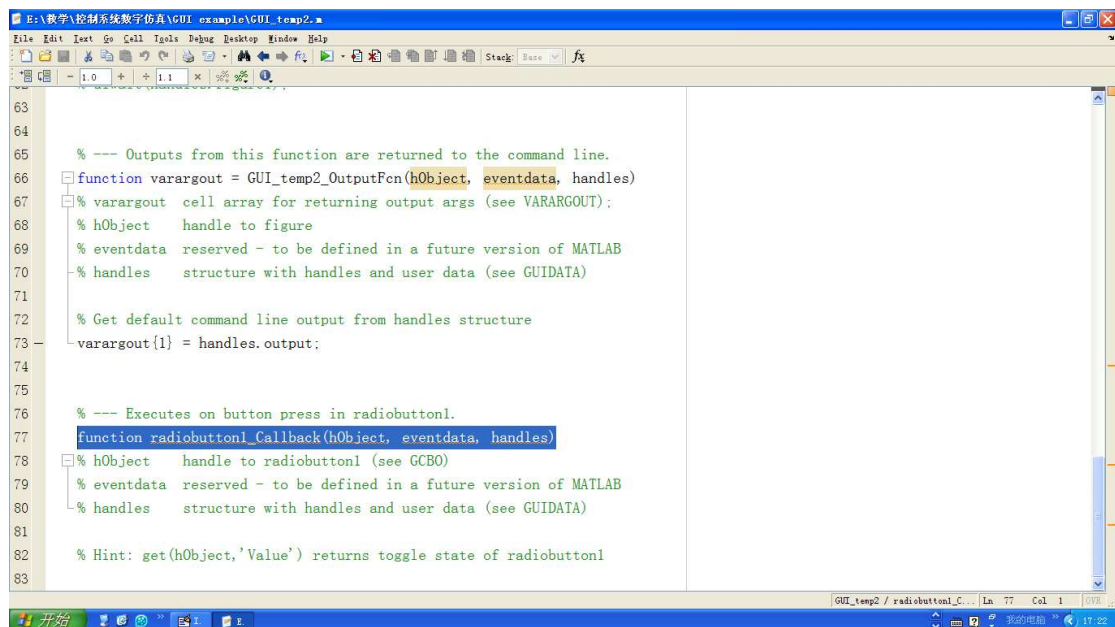


图 11 GUI 界面的模板文档

7. 在回调函数“function radiobutton1_Callback(hObject, eventdata, handles)”中写入如下代码
x=0:0.1*pi:2*pi;
y=sin(x);
plot(x,y,'b-p')

8. 运行该 GUI 界面，点击 Radio Button，画出下图

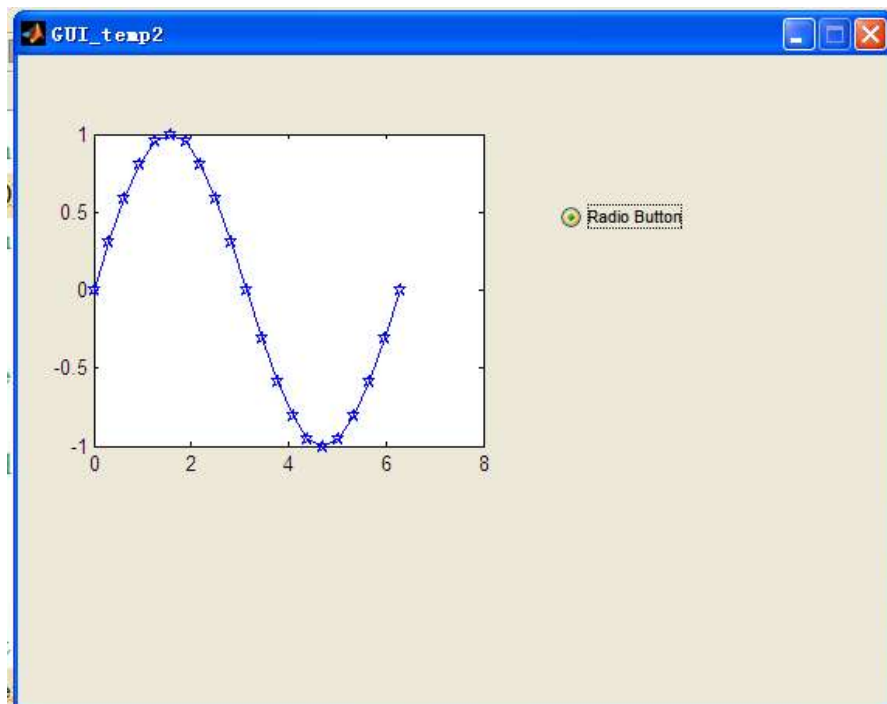


图 12

9. 若在画布中拖放两个坐标轴，两个按钮，如图 13

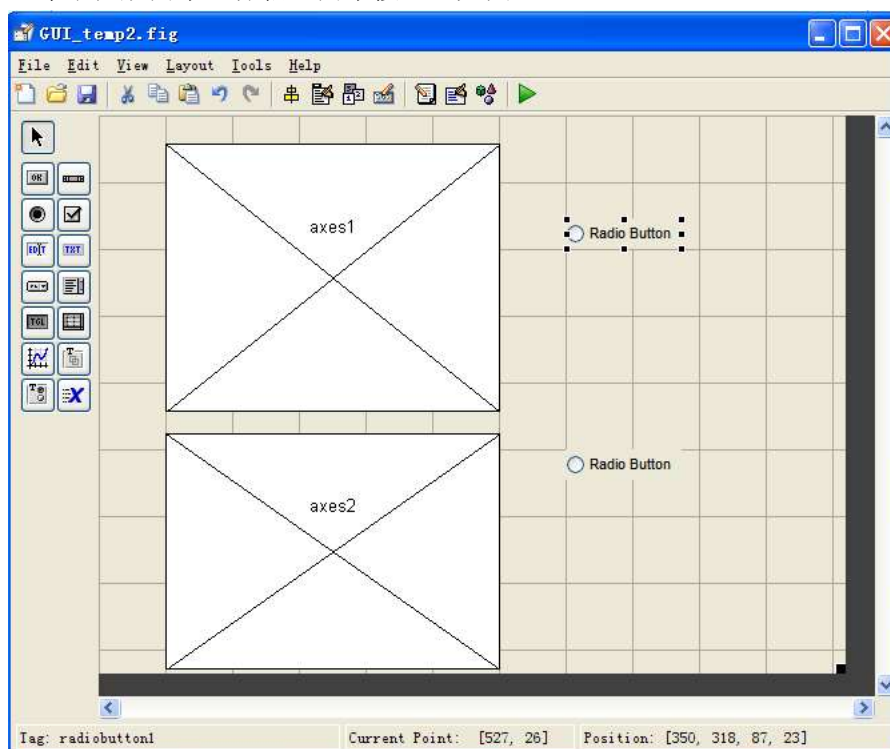



图 13

分别双击两个按钮，在弹出的属性页中选择“String”属性，将两个按钮的文本名分别改为“正弦”和“余弦”

Position	[69.8 11.077 17.4 1.769]
SelectionHighlight	on
SliderStep	[0.01 0.1]
String	余弦
Style	radiobutton
Tag	radiobutton2
TooltipString	

10. 分别双击两个按钮的“Callback”属性的图标，在正弦按钮弹出的回调函数

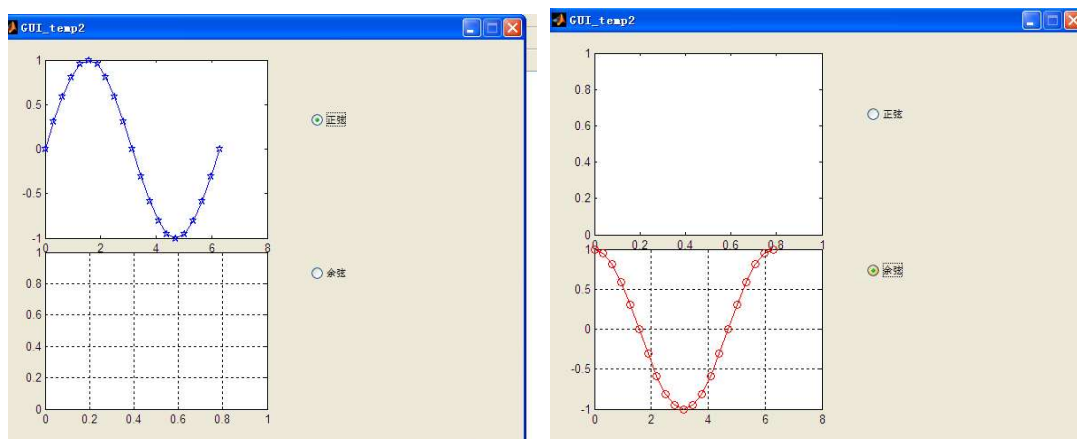
“function radiobutton1_Callback(hObject, eventdata, handles)”中写入如下代码

```
x=0:0.1*pi:2*pi;
y=sin(x);
plot(handles.axes1,x,y,'b-p');
cla(handles.axes2);
set(handles.radiobutton1,'Value',1);
set(handles.radiobutton2,'Value',0);
grid on
```

在余弦按钮弹出的回调函数“function radiobutton2_Callback(hObject, eventdata, handles)”中写入如下代码

```
x=0:0.1*pi:2*pi;
y=cos(x);
plot(handles.axes2,x,y,'r-o');
cla(handles.axes1);
set(handles.radiobutton2,'Value',1);
set(handles.radiobutton1,'Value',0);
grid on
```

运行结果，实现按钮的互斥功能。如下图所示。



通过上述实例，理解 GUI 界面设计过程和利用句柄进行属性设置和回调函数的写法。

★★★★★GUI 设计小知识

Handle 是一个对象(图形窗口、坐标轴、控件、函数)的识别码(Identifier)，在每次程序运

行时，句柄值是和对象唯一对应的。

`gca` 获取当前坐标轴的句柄；

`gcf` 获取当前窗口的句柄；

利用句柄值可以对该句柄指定的对象属性进行操作

`Set(句柄值, 属性, 属性值);`

当获取到句柄值后，利用句柄对对象的属性进行设置时，要注意到属性的要求。

`set(句柄值, 属性, 属性值, ...)`

`get(句柄值, 属性, 属性值)`

1. GUI 回调函数 Callback

Callback 是在点击控件时触发的动作。

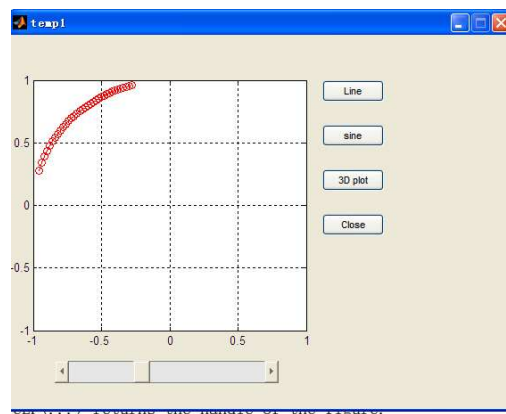
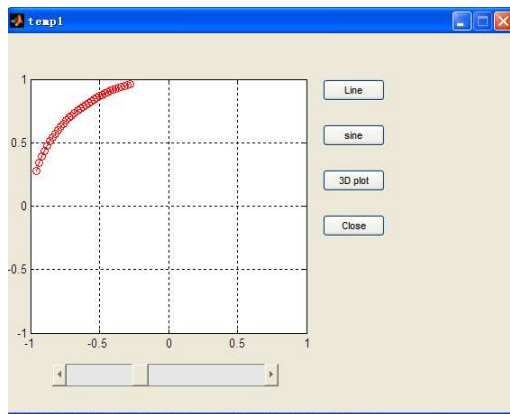
CreatFcn 是控件创建时触发的动作

Pause 是暂停

M 文件格式的界面作实时的交互操作，可能会出错!!! 这是由于 m 文件效率太低造成的。

2. 可以在 CreatFcn 中用 `gca`(获取当前坐标轴的句柄)、`gcbo` (获取当前控件的句柄)获取句柄，再根据句柄进行操作。
3. 在 GUI 界面的控件属性页回调函数的对话框中直接写代码，相当于写的是脚本文件。
4. 也可以在 GUI 生成的 m 文件回调函数中写代码，这些代码是在函数中执行的，不是脚本。所以这些代码变量的生存周期是和回调函数是一致的。句柄用 handles 传递。

四、编写 GUI 界面程序，用滑动条实现如下功能



当点击滑动条右侧时，在坐标轴上画出红圈，当点击滑动条左侧时，红圈逐渐消失，