

Graphical User Interfaces I



The central station is the bubble gum planet with 280 pounds of clay and three head phone lines that come out and orbit the mass. The clay is not only on the top of the table but under it and under the chairs. It is inspired by the gum under tables rebelliously put there by those imprisoned in boring pedantics. It is a monument to the struggle of repressed physicality. Its form is never finalized as people freely sculpt it.

Outline

- Announcements
 - HW III due Wed. 5PM
 - win valuable prizes!
- Why GUI's?
- GUI components
- Layout with GUIDE
- Callbacks
- Example: converttemp.m

Why GUI's?

- Matlab's real power lies in its programmability, so why would you want to make a GUI?
 - Interactive demo of your ideas/data/algorithms
 - Your Grandma/advisor needs to run your code
 - Impress your friends
 - Kind of fun

GUI Components

- Matlab GUI's can have all of the same UI features you see in Windows/Mac/X-Windows programs
 - radio buttons--choose between discrete values
 - text boxes--enter text
 - buttons--click, click, click
 - menus
 - sliders
 - dialog boxes
 - load/save windows

GUI Components

- There are three GUI-objects:
 - Uimenu -- like "file" or "edit" menus at top of figure
 - Uicontextmenu -- menus that appear when clicked
 - Uicontrol--everything else
- Uicontrols & Uicontextmenus are children of figures
- Uimenu can be children of other menus

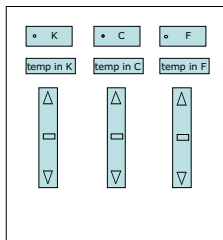
Uicontrols

- Uicontrols are the most important components
- These are created with `ui=uicontrol(mom, 'property', 'value',...)`
 - once you have the handle (ui), you can change values with `set`
 - the type of uicontrol is specified with the style field
 - `pushbutton`, `togglebutton`, `radiobutton`, `checkbox`, `edit`, `text`, `slider`, `frame`, `listbox`, `popupmenu`

Building GUI's

- A GUI is a function (or series of functions). Somewhere in the function it should create a figure and place calls to uicontrols
- Matlab has a special GUI called "guide" that lets you layout your GUI in WYSIWYG format
 - Allows you to position controls and set their properties interactively
 - Saves a function that will create your basic GUI

Example converttemp



- We want to create a simple GUI to convert between Kelvin, Celsius, and Fahrenheit temperatures
 - Users will specify temperatures by typing or moving sliders

Example converttemp

- We will create the buttons and sliders with GUIDE and then save it as converttemp.m
- We will then clean up the function and have it return handles to the objects.

Callbacks

- Right now, our GUI is just a bunch of buttons & sliders. We can type and click, but our clicking is meaningless.
- Uicontrols have a field called callback
 - This field contains a call to a function (as a text string) that gets executed when the control is activated (clicked, slid, typed into)
 - Callbacks give the GUI life

Callback Programming

- Callback functions will need to manipulate the properties of the objects in the window, in order to produce graphical results
- we will need to get handles to these objects
 - gcbo -- gets handle to the callback object (the one that was clicked)
 - gcbf -- gets handle to the figure containing the callback object
 - g=findobj(h, 'property', 'value')
 - returns handles to objects under (children of) object h with property=value

Callback Programming

- We don't want to write a callback function for every object
- Usually, you have a single callback for all controls, controls of a certain type, or groups of controls
- The functions must then figure out the right thing to do
 - Matlab recommends "switchyard" programming:

```
function cbfunc(id)
    switch id
    case 1:
        <do something>
    case 2:
        <do something>
    end
```

Callback Programming

- Calls to callback functions are usually set when the GUI is started (they don't have to be though,...)
- This means that we can't pass data to the callbacks
- Data for GUI's is "hidden" in the objects

Callback Programming

- Places to launder your data
 - Tag -- a name or comment
 - can search for tags with findobj
 - Value--value of an object in its current state
 - (0,1 for radiobuttons, value between MIN and MAX for sliders)
 - String--label of object, or the what the user typed in an edit box
 - UserData--anything you want!
 - ALL OBJECTS HAVE USERDATA !

Example: converttemp.m

- We've set the positions, lets set some data:
 - Set sliders' min & max values
 - Set sliders' values
 - Set edits' strings
 - Set Celsius button to on (value=1) and others to off (0)
 - Set Kelvin and Fahrenheit sliders' visibility to off
 - Set figures userdata to handle matrix
- Let's figure out the callbacks!
