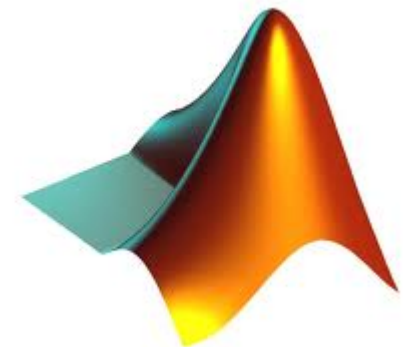


GUI, DU1

Cvičenia z Pokročilého spracovania obrazu

IPT Metódy hľadania hrán

- Sobel
- Canny (potlačenie šumu, 4 filtre, prahovanie pomocou hysterézy)
- Roberts
- Prewitt
- LoG (Laplacián Gaussiánu) = Zero crossing (Marrov–Hildrethovej algoritmus)



IPT Metódy hľadania hrán

```
edge (I) ;
```

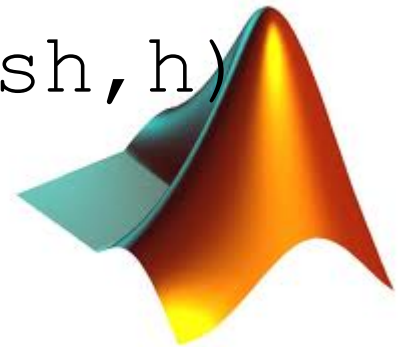
```
edge (I, 'sobel')
```

```
edge (I, 'log', threshhold)
```

```
edge (I, 'canny', threshhold, sigma)
```

```
edge (I, 'zerocross')
```

```
BW = edge (I, 'zerocross', thresh, h)
```



GUI

```
>> guide
```

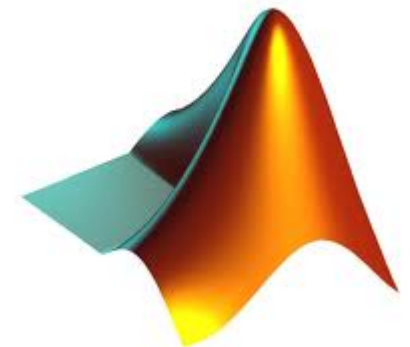
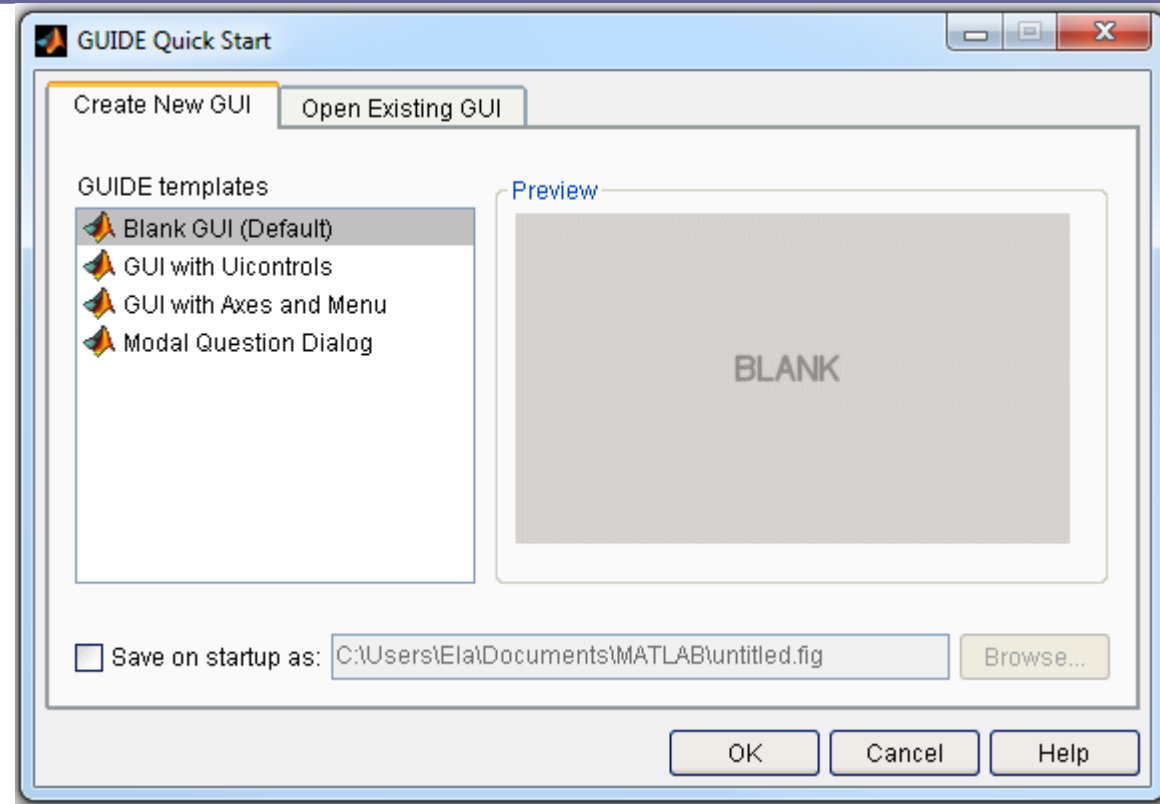
- **Blank GUI**

- **Vytvorí dva súbory:**

- `meno.fig`

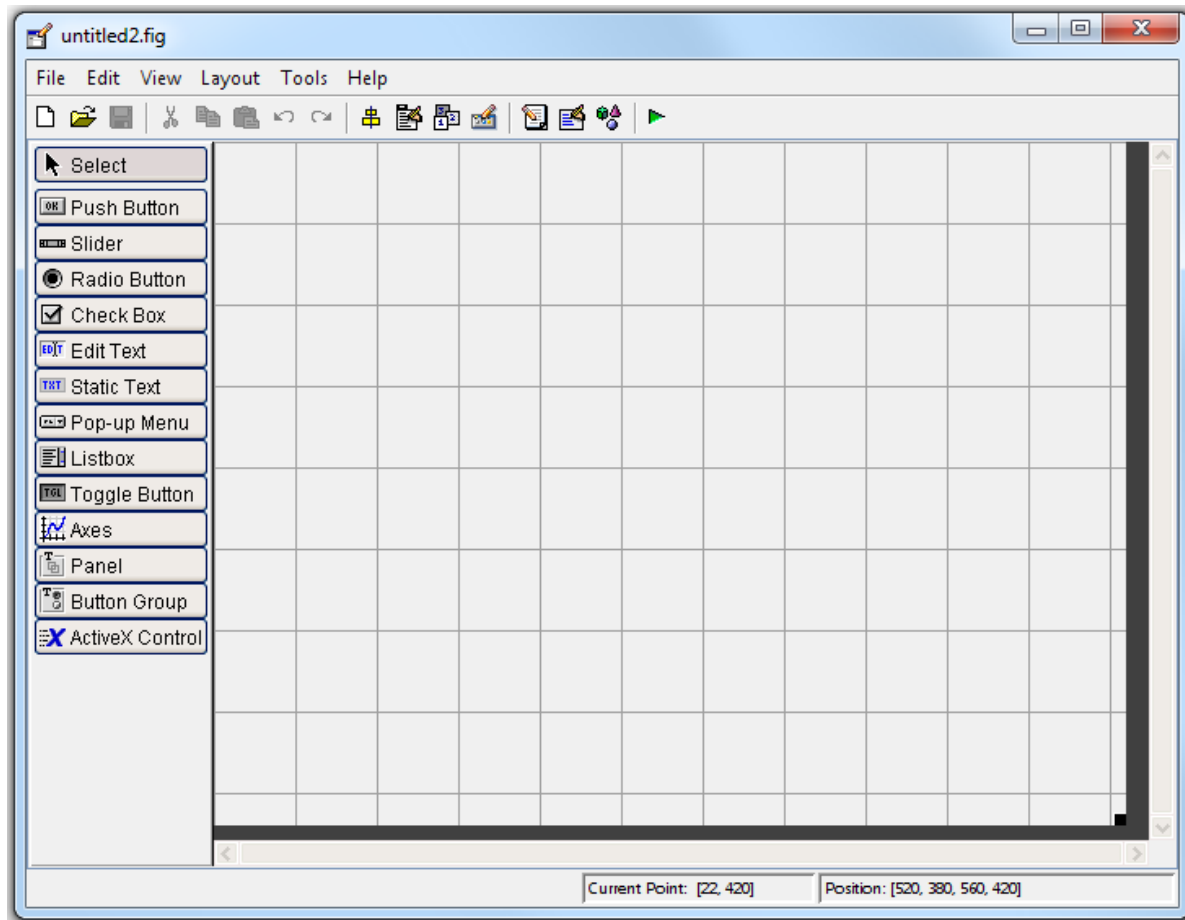
- `meno.m`

- **Nemeniť meno už vytvoreného GUI**

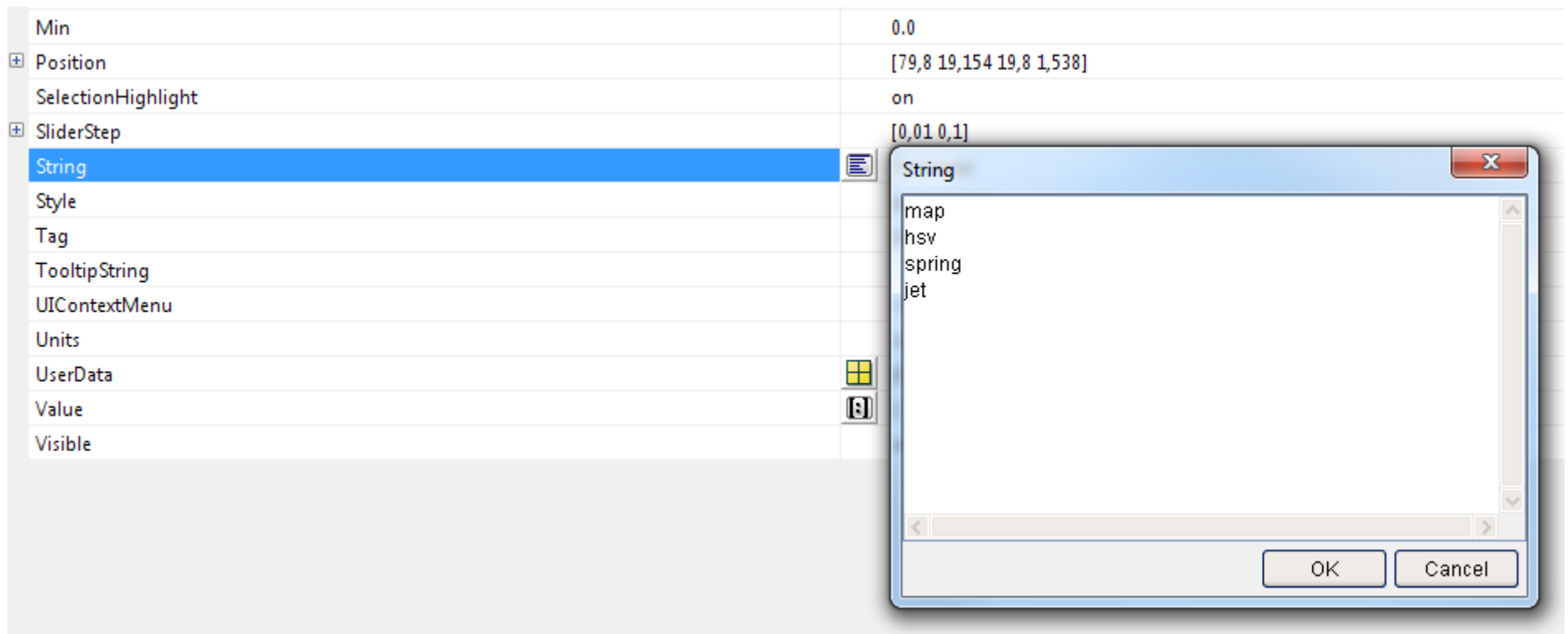


GUI

- GUI objekty:
 - Button, radio button, check box, slider
 - Edit text, Static text
 - Axes
 - Pop-up menu
 - List box
 - Panel
 - Button group...



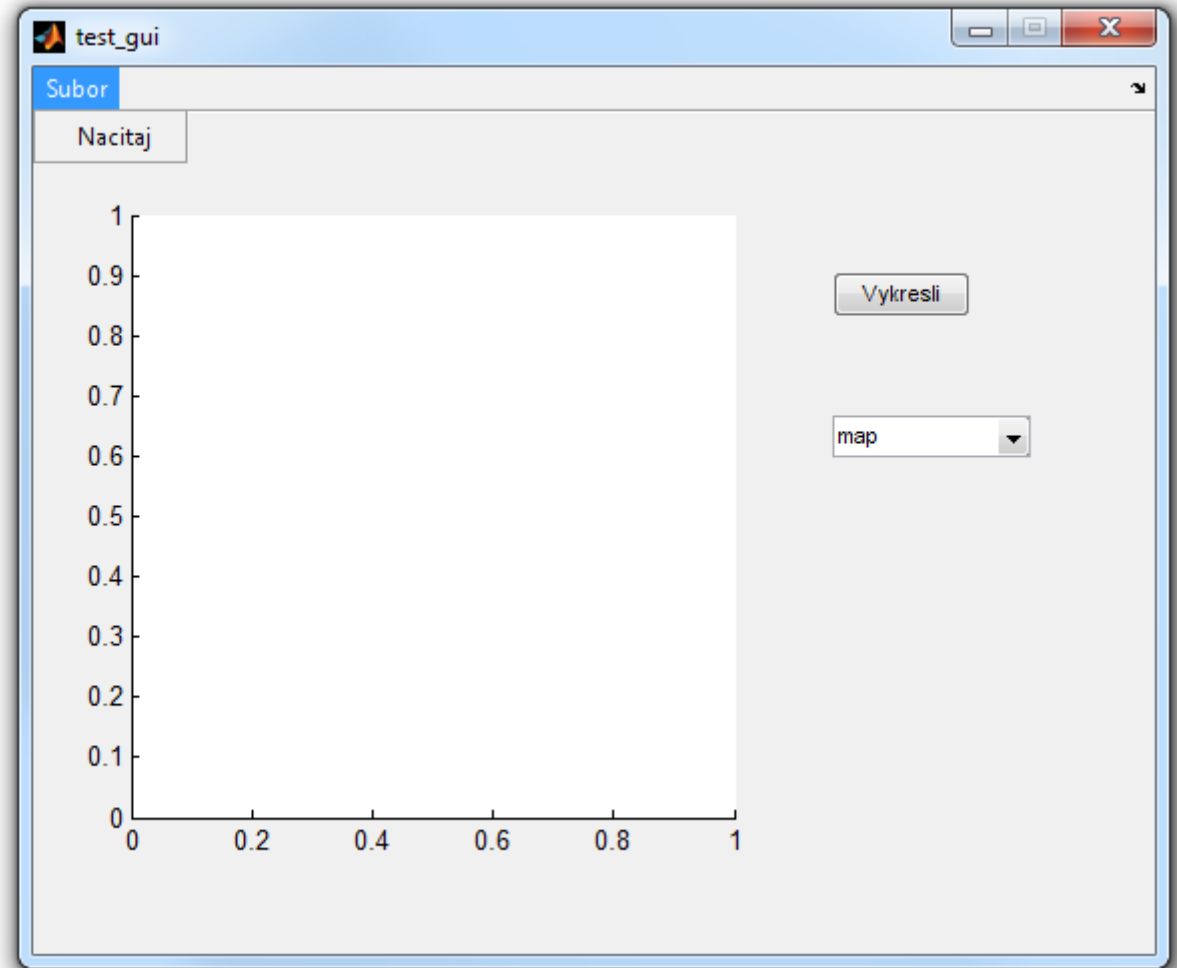
- Property Inspector
 - Color, text, name, position, opacity



GUI

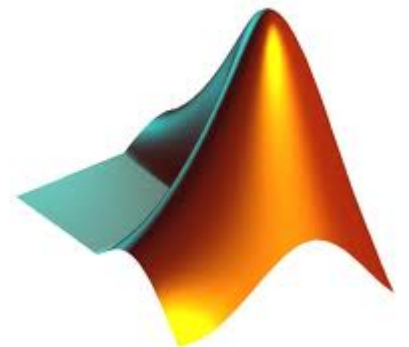
- Save as
- Spusti
- Nic sa nedeje!
- Treba dopísat' kód

```
>> test_gui  
>>
```



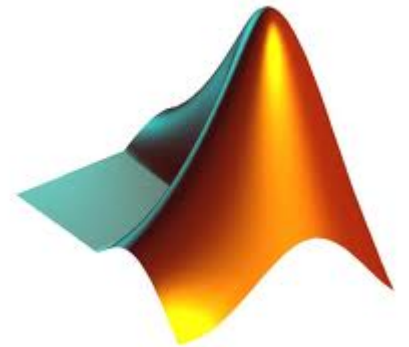
GUI

- Callbacks = funkcie, ktoré sa vykonajú po aktivácii objektu
- ak chceme využívať v jednom callbacku premennú ktorú sme vytvorili v inom, použijeme funkcie `get` a `set`



GUI - Callbacks

- Callback
- ButtonDownFcn
- KeyPressFcn
- CreateFcn
- ...



GUI

- **Handles = štruktúra uchovávajúca data**

```
set(handles.text2, 'Visible', 'on');
```

```
g = get(handles.radiobutton1, 'Value');
```

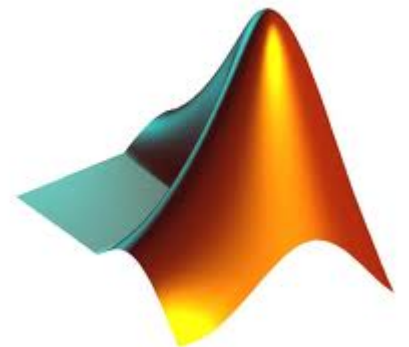
```
set(object, 'property', value)
```

```
get(object, 'property')
```

- **Globálne data:**

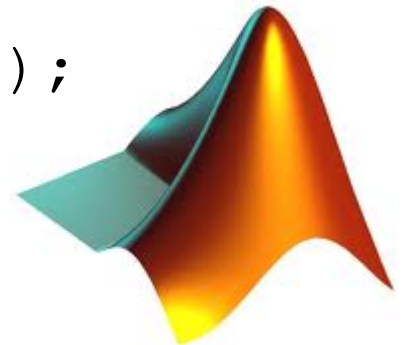
```
handles.moje_data = hodnota;
```

```
guidata(hObject, handles)
```



GUI

- Objekty majú okrem štandardných parametrov tzv. 'UserData',
 - môžeme vložiť ľubovoľné dáta (obrázok, číslo)
- V jednom callbacku načítame
 - `RGB = imread('1.jpg');`
 - `set(handles.pushbutton1, 'UserData', RGB);`
- V druhom zavoláme
 - `I=get(handles.pushbutton1, 'UserData');`
 - `imshow(I);`

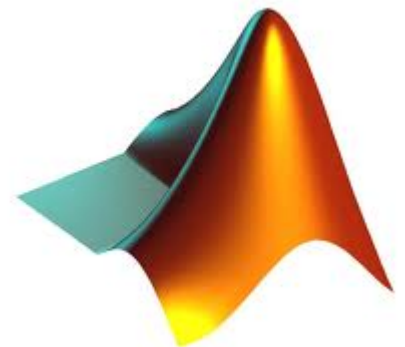


- Načítanie obrázka

```
[FileName, PathName]=uigetfile('* .jpg',  
'Vyber .jpg');
```

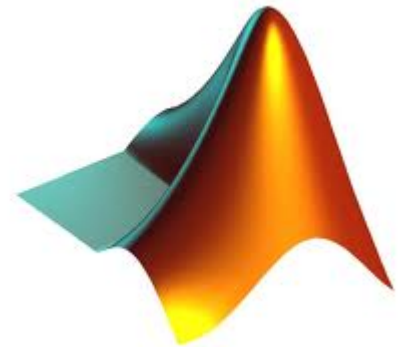
```
I = imread(fullfile(PathName,  
FileName));
```

```
figure; imshow(I);
```



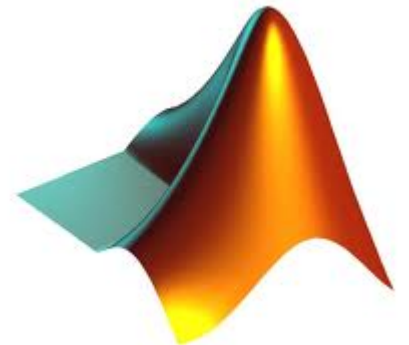
Veľká úloha 1

- Do 28.10.2015 23:59
- *priezvisko_GUI.fig*, *priezvisko_GUI.m*
- Email: DUvidenie@gmail.com
- **!!! Subjekt správy: PSO DU1**



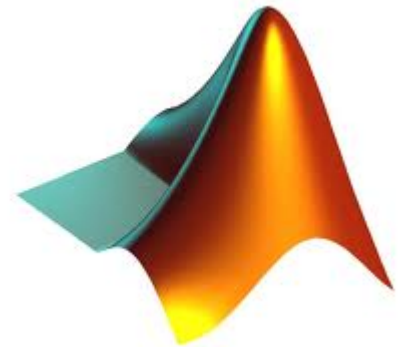
Musí spĺňať

- 2 vykresľovacie plochy (**axes**)
 - 1. Originál (farebný alebo šedoúrovňový)
 - 2. Výsledok aplikovanej metódy
 - Možnosť zvoliť si, ktorý obrázok budeme následne upravovať (šedoúrovňový, alebo upravený (napr šum))
- Načítanie obrázka (jpg, png, bmp) z adresárovej štruktúry pomocou dialógového okna (**menu**)
- Konverzia na šedoúrovňový obrázok (**menu alebo tlačítko**)
- Možnosť pridania šumu (**button group alebo popup menu**)



Musí spíňať

- 3 metódy na hľadanie hrán (menu, popup menu alebo button group)
 - bez použitia funkcie edge
 - možnosť výberu prahu (edit text alebo slider)
 - výsledok bude binárny obrázok
- 2 metódy vyhladenia obrázka (odstránenie šumu) (menu, popup menu alebo radiobutton)
 - zadanie veľkosti filtra (edit text, popup menu alebo radiobutton)
 - Gaussian, mean, median
- po každej úprave zobrazit' výsledok



Typy

- Vytvoriť prehľadné GUI
- Vytvoriť pomocné funkcie na detekciu hrán a vyhladenie obrázka, ktoré majú na vstupe obrázok, zvolenú metódu a nastavené parametre (veľkosť filtra, prah)

