Programming Language: MatLab

1st Semester 2015

Chong-Wai

W13, 6th Dec

Content

- Panel
- Grouping
- The method
- The code

Preparing the panel

Preparing a panel as below:

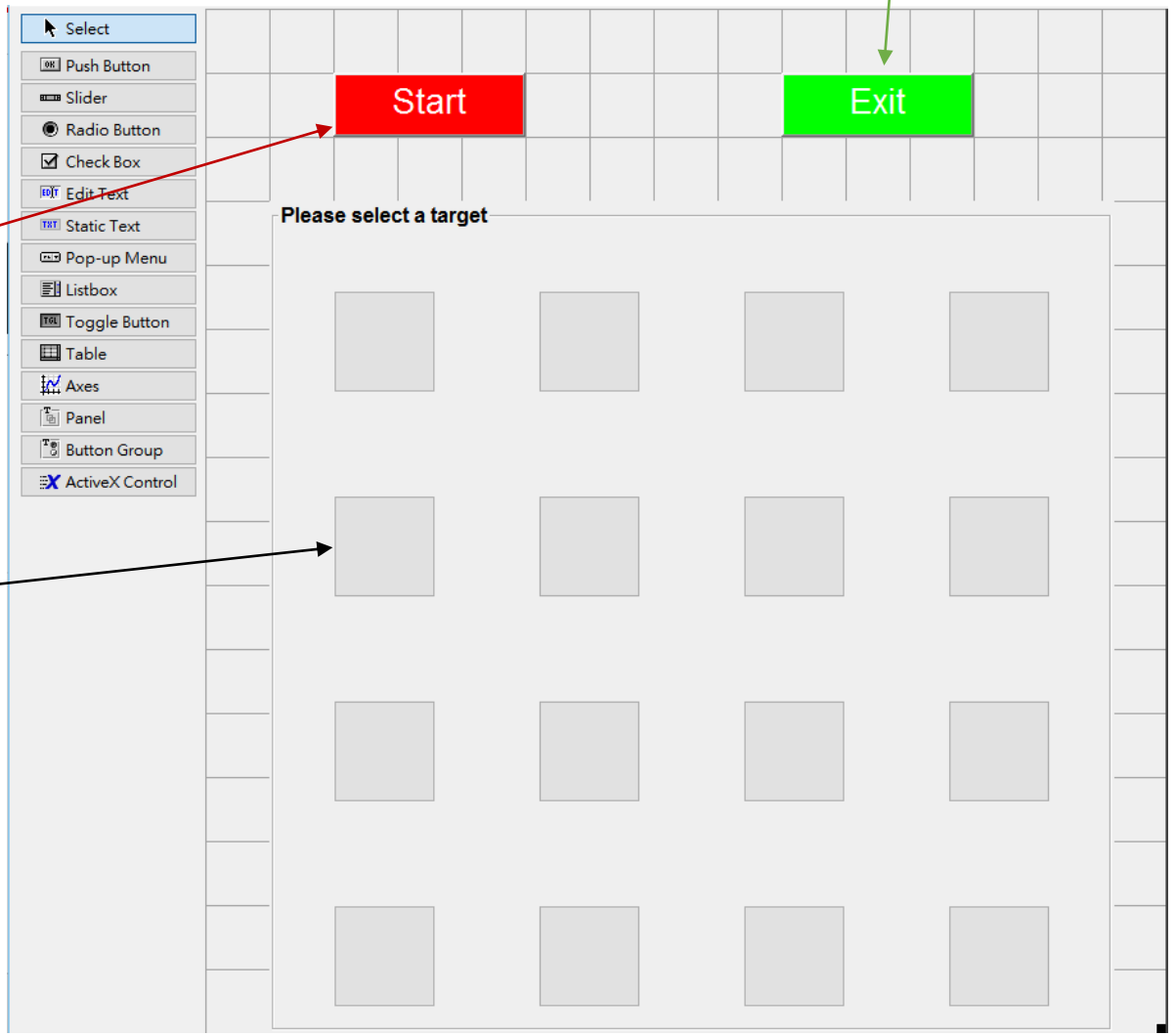
a) 16+2 Pushbutton

b) 1 Button Group

Pushbutton:
Exit

Pushbutton:
Start

Pushbutton:
B1~B16



Preparing the panel

Loading the image to the push button

Any preloaded action should be embed in “OpeningFcn”

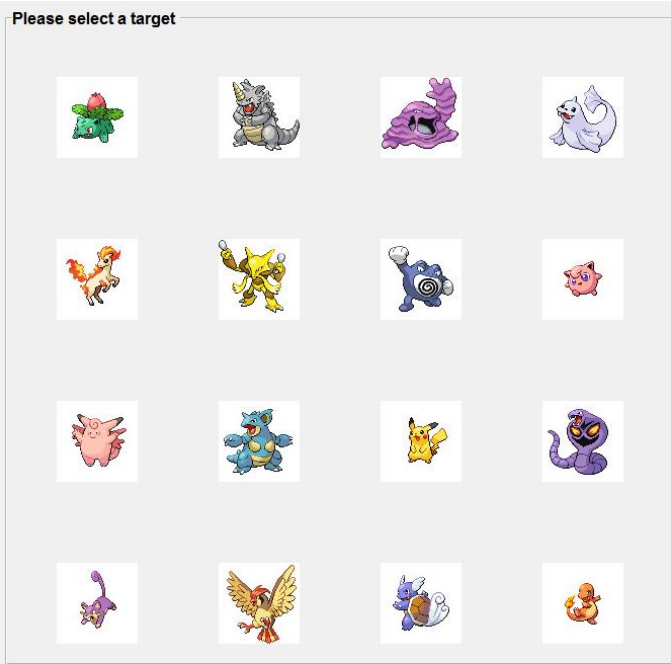
```
function Proj1_OpeningFcn(hObject, eventdata, handles, varargin)

handles.output = hObject;
guidata(hObject, handles);
for k = 1:16
    [ico,map]=imread(sprintf('im%d-1.jpg',k));      % reading the image to variable ico
    set(handles.(sprintf('B%d',k)),'cdata',ico);    % revealing the image on the push button
end
```

Reading the image to variable ico
(refer to lecture note in W10_3)

Setting the image on the pushbutton through handles->’Cdata’

sprintf(): convert integer (number) to string

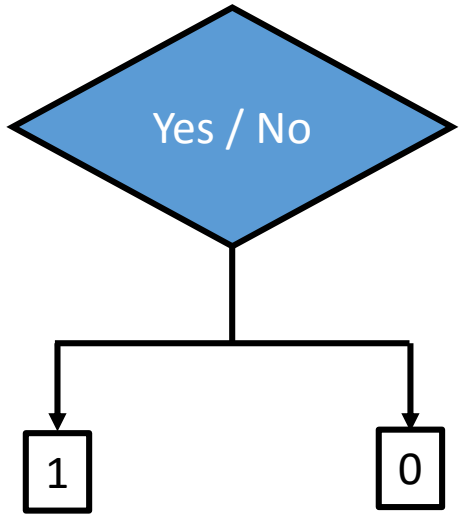


Grouping

Dec	Bin			
	2^3	2^2	2^1	2^0
0/16	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
12	1	1	0	0
13	1	1	0	1
14	1	1	1	0
15	1	1	1	1

The third group

Binary coding



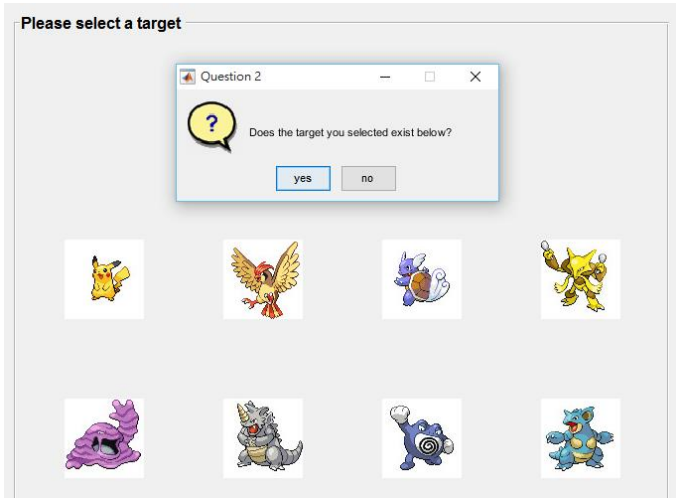
Repeated 4 times

Possible config. = 2^4=16

Method

- Classifying the buttons to 4 groups (each with 8)
- Showing the corresponding buttons group by group
- Using question dialog box to pro

```
Var = questdlg('qstring','title','str1','str2',default)
```



- Recording the corresponding response
- Showing the result (hiding the rest)
- Adding mystery by randomizing the sequence

The code

The actions of pushbutton “Start”

```
function Start_Callback(hObject, eventdata, handles)

bits =    [[1,3,5,7,9,11,13,15];
           [2,3,6,7,10,11,14,15];
           [4,5,6,7,12,13,14,15];
           [8,9,10,11,12,13,14,15];];
% Sorting the target to four groups
```

Defining the group members according to p5

```
bits =

     1     3     5     7     9    11    13    15
     2     3     6     7    10    11    14    15
     4     5     6     7    12    13    14    15
     8     9    10    11    12    13    14    15
```

```
Xpos = [50;210;370;530;50;210;370;530]

Ypos = [340;340;340;340;180;180;180;180]
```

Defining the locations for the push buttons

```
Tar = 0;
```

The variable to record the response from dialog boxes

The code

The actions of pushbutton “Start”

```
for L = 1:4
    for k = 1:16
        set(handles.(sprintf('B%d',k)),'visible','Off');
        % hiding all the buttons
    end

    ordering = randperm(8);

    for j = 1:8
        % visualizing and relocating the corresponding button
        X = Xpos(ordering(j));
        Y = Ypos(ordering(j));
        set(handles.(sprintf('B%d',bits(L,j))),
        'visible','On','position',[X Y 80 80]);
    end

    resp = questdlg('Does the target you selected exist below?'
        ,(sprintf('Question %d',L))','yes','no ','yes')
        %,'position',[100 100 300 100]);


    waitfor(resp);
    if resp == 'yes'
        Tar = Tar + bits(L,1)
    end
end
```

The loop for 4 questions

Hiding the unnecessary buttons (visible = off)

Generating a random the ordering

Showing the buttons of the current group (visible = on) & Relocating their position according to (random) ordering

Creating a message box and record the response to ‘resp’

Storing the result to ‘Tar’ for differ “L” (loop)

The code

The actions of pushbutton “Start”

```
for k = 1:16
    set(handles.(sprintf('B%d',k)),'visible','Off');
    % hiding all the buttons
end
```

Hiding all the buttons, before showing the result

```
    set(handles.uibuttongroup1,
    'ForegroundColor','r',
    'FontSize',24,
    'Title', 'The target you selected is:');
```

Modifying the text and color of the button group

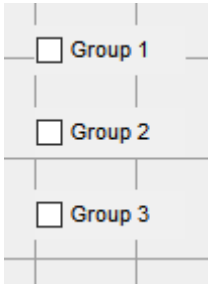
```
if Tar == 0
    set(handles.B16,'visible','On','position',[290,340,80,80]);
    % Showing the result
else
    set(handles.(sprintf('B%d',Tar)),'visible','On',
    'position',[290,340,80,80]);     % Showing the result
end
```

Justifying the result and showing it correspondingly

Homework

Try to construct a similar GUI program, for some updates

1) Background and button images
2) Using “checkbox” to replace “message box”



3) You can prepare a program with N elements for N > 4 and not equal to 16

extra) For N > 16