

ELE101

Precept 6

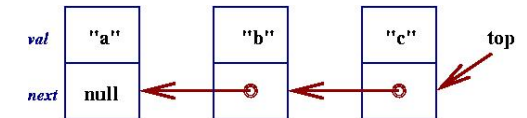
James Donald
March 25-26, 2004

Graphical User Interface (GUI) Programming

Review & outline

- Last time:

- Pointers
- Strings
- Structures
- Linked lists



- Today:

- GUI Programming
 - Event-driven programming
 - Program resources



Setup instructions

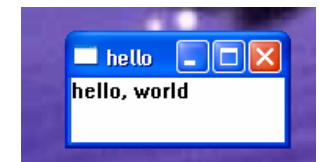
- Download the precept 6 files from Blackboard and unzip **precept6.zip** into a new folder preferably somewhere on your H:\ drive.
- Open up **ele101login.bat**
- Use the 'cd' command to reach the new folder where you unzipped the contents of **precept6.zip** from within the shell.
- Open up the emulator with the following command:
 - bash.202\$ **emulator_precept6.bat &**
- If you got here early and have time to spare, try typing **make** and observe the build process.

What is event-driven programming?

- Example: Let's try running "hello world" for Windows. Type in the following:
 - bash.202\$ **./hellowin**
- A big step forward from our text-based "hello world".
- The full source is in **hellowin.c**, about 80 lines long.
- Key parts:

```
while (GetMessage(&msg, NULL, 0, 0)) {  
    TranslateMessage(&msg);  
    DispatchMessage(&msg);  
}
```

```
/* In some other function ... */  
switch (message) {  
    case WM_PAINT:
```



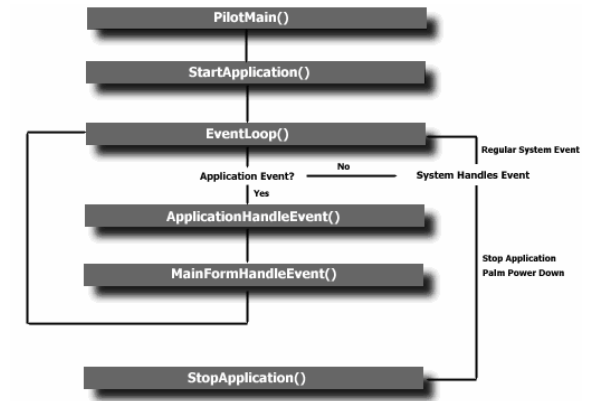
GUI programming on the Palm

- Same concept, but different interfaces as usual.
- Click on the 'hello' icon in your Palm emulator, and let's try this program out.
- Now let's check out the source code in **hello.c**.
 - Contains code to initialize the program.
 - Then enters the event loop mode.
 - Event loop code repeatedly calls AppHandleEvent() or MainFormHandleEvent().
 - After finishing the event loop, it runs some clean-up code then exits.



With so much baggage, how can we manage our GUI programs?

- We'll relocate common Palm GUI functions into another file called **ele101.c**.
- Open up our next example, **lottery.c**.



Dissecting the lottery program

- Click the 'lottery' icon in the emulator and let's see what this program does.
 - Upon the click of a button, it prints data to the screen.
 - The text entered in the input field gets processed.
 - Records the locations of all taps to the screen.
- Where is the code to handle these events?
 - It is all in `MainFormHandleEvent()`



User input and output

- Sensing stylus taps:

```
case penDownEvent:
    /* ... */
```
- Sensing button clicks:

```
case ctlSelectEvent:
    if (evt->data.ctlEnter.controlID ==
        drawbutton)
        /* ... */
```
- Getting input from the text field:

```
field = FrmGetObjectPtr(FrmGetActiveForm(),
    FrmGetObjectIndex(FrmGetActiveForm(),
        numberfield));
str = (Char*) FldGetTextPtr(field);
```
- Printing to the screen:

```
WinDrawChars("hello, world", 12, 50, 60);
```



hello, world

Administrivia

- Homework assignment #4 due Tuesday March 30th.
- Midterm #2 on Thursday April 1st.
- Office hours as usual after precept today.

- Mobile computing in the news:

San Francisco Chronicle

TREK TECH

40 years since the Enterprise's inception, some of its science fiction gadgets are part of everyday life

March 15th, 2004

"When I designed the UI (user interface) for the Palm OS back in '93, my first sketches were influenced by the UI of the Enterprise bridge panels," said Rob Haitani, product design architect for Palm-One Inc., the Milpitas firm that makes the popular handheld personal computers.



Program resources

- Where did we initialize the 'Draw' button and the input field? There is no explicit code to create these controls in **lottery.c**.
 - **ele101.c** calls `FrmInitForm(formId)`; but how does it know what's in the form?
 - Form contents are specified in a resource script: **lottery.rcp**. Application 'resources' include things such as menus and icons.
 - **lottery.rcp** is not written in C. It is a funky scripting language, but we can still use it in conjunction with our C programs.

12 \$

The lottery program's .rcp file

```
#include "resource.h"
```

```
FORM MainForm 1 1 158 158
```

```
BEGIN
```

```
    TITLE "Lottery Numbers"
```

```
    BUTTON "Draw!" ID drawbutton AT (CENTER 40 AUTO AUTO)
```

```
    FIELD ID numberfield AT (CENTER 20 75 AUTO) \
```

```
        UNDERLINED MAXCHARS 15
```

```
    GRAFFITISTATEINDICATOR AT (150 150)
```

```
END
```

```
APPLICATIONICONNAME 1000 "Lottery"
```

```
ICON "lottery.pbitm"
```

```
VERSION "0.0.1"
```

- But for now don't worry too much about the details. We'll provide this file for at least the next couple assignments.
- **resource.h** is the missing link between the resource script and our C program.

In Conclusion

- Graphical user interface (GUI) programs are typically written using an event-driven style.
- For Palm GUI programming, **#include "ele101.h"** to modularize your code; the meat of your program goes in `MainFormHandleEvent()`.
- We've learned new methods for user input and output that could take some getting used to.
- A Palm program's resources are defined by an **.rcp** resource script file.

