Basics of learning efficient designing in Qt

An introduction to designing GUI in Qt

Basic Reading Material

Please use the following basic reading material:

1. http://www.zetcode.com/gui/qt4/utilityclasses/
2. http://qt-project.org/doc/qtcreator-3.0/creator-tutorials.html

Advanced reading Material

The reader with advanced background would find this material useful:

1. https://qt-project.org/doc/qt-5.1/qtquick/tutorials-samegame-samegame1.html

Introduction : What will we learn here ?

Designing in Qt.

The Problem Statement

Problem 1 (Compulsory - 100 marks, difficulty level: *)

Design a user interface(GUI) in Qt. The user will be asked to enter a number in numeric (say, 55) and will be provided a button named “Convert to text”. On pressing this button, the entered number should be displayed in words (fifty five).

Optional bonus problems

These following problems are optional and have higher levels of difficulty. Bonus marks (indicated against each) will be given for attempting these.

(**/10 points) Problem O1 :

Add one more feature to the above GUI, which will enable user to enter a number in text (fifty five). Add one button named “Convert to Number”, on pressing which the entered number in words will be shown in numeric digits (55).

(**/10 points) Problem O2 :

Add one more feature to the above GUI, which will enable user to enter a number in text (fifty five) and display a comma separated value as it is typed in (note that the internal representation will be without commas of course).

In India, accountants often like to put comma inside long numbers (for example: instead of

55020010, we write 5,50,20,010 which is 550.2001 lakhs or 5.502001 Crores or “five crores fifty five lakhs twenty thousand and ten only”).

(/2 points) Problem O3 :**

Add one more feature to the above GUI, which will enable user to enter a number in text (say fifty five) and HEAR the spoken english phrase (i.e. “fifty five”). You don’t have to do this - tell the TA during your viva HOW it can be done

Submission Instructions

Please follow the submission instructions below:

1. Upload on Sakai before 17:15 hours

Material to read for next week’s lab (after semester break)

- gnu binutils (see http://www.gnu.org/software/binutils/)
  - Specially Profiling - the use of gprof (also see http://sourceware.org/binutils/docs/gprof/)

- gasm - the gnu assembler and inline coding https://web.archive.org/web/20120510042325/http://www.ibm.com/developerworks/linux/library/l-gas-nasm/index.html

- compiler directives (see http://www.cprogramming.com/reference/preprocessor/)
  - pragma
  - ifdef and ifndef,
  - elif
  - extern variables; scope of variable
  - limiting compiler passes to only the first pass to produce assembly code instead of machine code

Try these sample / representative questions for next week

Based on the above reading material for next week, please attempt the following sample / representative problems:

1. (Difficulty level *)
   a. use gprof on the code you have developed so far and rewrite the code accordingly.

    b. test the compiler directives to see how they work

2. (Difficulty level **)
    a. swap two 8 bit unsigned numbers using assembly code.
    Using the time function, measure the cpu time it takes to execute this.
    b. Include this code in a C program. Now measure the cpu time it takes to execute this.
    c. Using pure C, swap the two unsigned 8 bit numbers. Again measure the cpu time it takes to execute this.
    d. What is the fastest way you can think of to swap two numbers (both 8 bit and unsigned) ?
3. (Difficulty level ***)