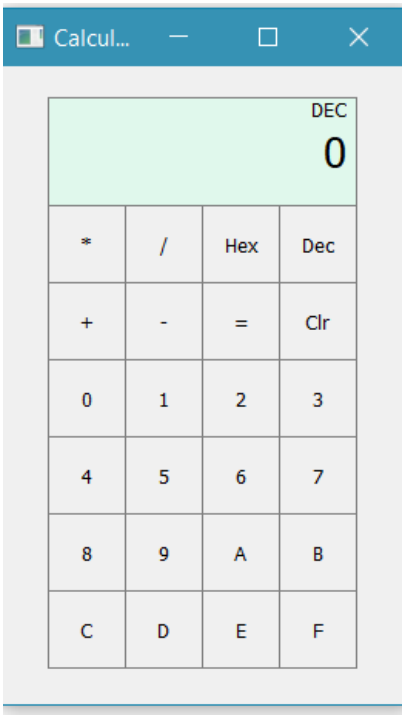
CMPE 230 – SYSTEMS PROGRAMMING

PROJECT 3

In this project, the task is to develop a decimal/hexadecimal calculator using QT Programming.



There are two parts of the project one of them is designing the interface and the other is coding.

How to run the program?

On command line go to the directory where the project is. On a bash command prompt you need to enter:

```
$ qmake -project
$ qmake
$ make
$ ./Calculator
```

Please refer to https://wiki.qt.io/Getting_Started_on_the_Commandline for more details.

Or

Open QT Creator and build the program. Then, click run button.

How to use?

Calculator starts in decimal mode so make sure to change it to hexadecimal mode if you want to use hexadecimal digits. If a hexadecimal digit is clicked in decimal mode, “Invalid” appears on screen and calculator is reset. When mode is changed, the calculator is reset. Mode of the calculator can be seen on the screen.

“Dec” button: To change to decimal mode.

“Hex” button: To change to hexadecimal mode.

Insert a number using the digit buttons and press an operation. The operation that you pressed appears on screen under the number. Operations are as follows, all of them are binary operations:

“+”: Addition. Result is first number + second number.

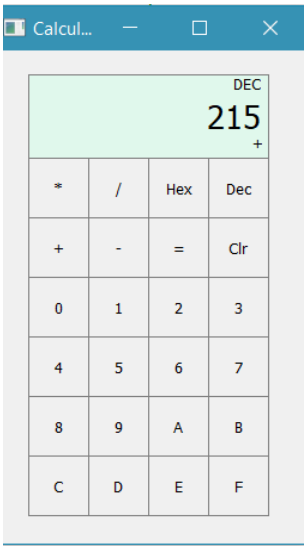
“-”: Subtraction. Result is first number - second number. Negative results may appear.

“*”: Multiplication. Result is first number * second number.

“/”: Integer division. Result is first number / second number. No fractional part appears.

Enter a second number using digit buttons. You can press “=” to see the result or you can continue performing operations by clicking them. Once you click another operation, the previous operation is performed, and result appears on screen. If you click an operation button and clicked another operation button immediately after it, the last operation is considered. For example:

254 is entered and “+” is clicked. After that “*” is clicked and 2 is entered and “=” is clicked. The result is equivalent to 254*2.



2 After a number is entered and "+" button clicked

1 After pressing an operation button, enter a new number and press "=" to see the result

If "=" button is clicked without clicking operation buttons, the calculator is reset.

To reset the calculator, click "Clr" button.

Implementation

In mainwindow.cpp, the interface design is connected to the code. In constructor, buttons are connected to specified slots for some actions.

For example:

connect(ui->pushButton0,SIGNAL(clicked(bool)),this,SLOT(digit_buttons()));

The button with name pushButton0 calls digit_buttons() slot when it is clicked.

There are three slots for different type of buttons.

For digits, digit_buttons() slot is used. Pressed digits are displayed in the screen according to the current condition.

For "+", "-", "*", "/" and "=" operator_buttons() slot is used. If no number is entered before, the current integer number which appears on the screen and the operator are stored. Else, calculate method is called with parameters pressed button's text and the last integer

number which appears on the screen. In calculate method, the last operation is applied to stored number and the number which is given as a parameter. If the pressed button's text is "=", then stored number is assigned to 0 and other variables set to their initial values. Else, this text is stored in last operator.

For "Dec", "Hex" and "Clr", control_buttons() slot is used. If button is "Hex", calculator's mode is changed to hex mode (by changing global Boolean variable isDec) and all values are set to their initial values. "Dec" does the similar things. "If "Clr" is pressed, clear method is called. This method sets the values of variables to their initial values and sets the value on the screen to 0.

There are also hexToDec() and decToHex methods which converts hexadecimal value to decimal or vice versa. Calculations are performed in decimal mode, so when the calculator mode is hex, the values are converted to decimal, the operations are performed, and the result is converted to hex again.

More information can be seen in the comments in the source code.

Conclusion

This Calculator can perform only limited number of operations and is very simple. New features can be added to improve it. Also, operator precedence can be considered. User may write the equation and the calculator may calculate it when "=" button is pressed.