



# **Lecture Qt001**

## **Introduction to Qt 5.x Programming**

Instructor: David J. Coe

CPE 353 – Software Design and Engineering

Department of Electrical and Computer Engineering

# Outline

- What is Qt?
- Why Qt?
- Access to Qt
- Required Skills for Success
- Reminders
- Key Points

# What is Qt?

- Qt is a cross-platform toolkit for developing applications that utilize a graphical user interface
- Cross-platform tools allow you to develop an application once and deploy it across different platforms with few if any modifications
- History
  - Developed by Trolltech
  - Purchased by Nokia for development of mobile apps
  - Now owned by Digia, Inc.

# Why Qt?

- Qt applications are deployable to all major desktop and mobile platforms
  - Desktop Platforms Supported
    - Linux, Mac OS X, Windows
  - Mobile Platforms Supported
    - Android, iOS, Blackberry
- Free access to open-source Qt development tools for Mac OS X, Linux, and Windows environments
- ***Widgets-based Qt is a great example of object-oriented programming in C++***

# Access to Qt - Lab

- Qt 5 is currently installed on the Linux workstations in EB-246 and EB-216
- Use your Charger Card to unlock the laboratory doors via the card reader
- All students enrolled in the course should have an account that allows access to these Linux machines
  - Default username is the same as your LDAP username that you use for Canvas or WiFi access
  - Default password is your First Initial and Last Initial followed by the last six digits of your A-number

# Access to Qt - Home

- You may also download an open source version of the Qt 5 development tools from <https://www.qt.io/download>
  - The toolkit is available as a pre-compiled binary or source code that you will have to compile
  - MySQL support may not be enabled by default with some versions of the binary
- As with any software downloaded from the Internet, be sure to scan for malware prior to installation

# Required Skills for Success

- Qt toolkit makes extensive use of C++ constructs for object-oriented development
  - Pointers and dynamic allocation
  - Void functions and value-returning functions
  - Classes, objects, and methods
  - Inheritance, virtual functions, and polymorphism
  - Operator and function overloading
  - Abstract base classes
  - Container classes (stacks, queues, lists, vectors, etc.)
- Additional skills
  - The ability to use standard development tools such as ***make***
  - The ability to manage your files from the Linux command line
- ***Hint: Review the above concepts***

# Reminders

- As a prerequisite for several 400-level CPE courses, it is critical that you successfully complete CPE 353
- Failure to successfully complete this course with a grade of C or better will delay your graduation an entire year since CPE 353 is only offered during fall semester



# Key Points

- If it has been some time since you looked at either CPE 211 or CPE 212, **you are already behind**
  - *Start reviewing now!!*
- The key to passing this course will be your ability to use the Qt 5 development environment on the Linux laboratory computers to design, implement, and test a program
  - *Be sure to practice using the lab machines when working through the sample problems!!*