Course Administration

- **Instructor**
  - Francisco de la Parra
  - parra@cs.queensu.ca
  (no mail to fadl@queensu.ca)
- **TA's:**
  - Akib Anwar Hridoy (saah@queensu.ca)
  - Lucas Bullen (lb149@queensu.ca)
  - Mackenzie Furlong (mwf2@queensu.ca)
  - Cameron Raymond (15cjkr@queensu.ca)
  - Duncan Stuart (ds172@queensu.ca)
  - Ella Stewart (eps3@queensu.ca)
  - Sean Remedios (sr166@queensu.ca)

---

Course Administration

- **Lectures**
  - Monday → 11:30 am – 12:30 pm
  - Tuesday → 1:30 pm – 2:30 pm
  - Thursday → 12:30 pm - 1:30 pm
- **4 Lab sessions per week (11 weeks)**
- **3 Quizzes**
- **4 Assignments**
- **Exam – 3 hours**

---

Grading Scheme

- Three quizzes (3 x 5% = 15%)
  - About 20 multiple choice questions
  - One or two coding problems
- Four assignments (4 x 10% = 40%)
  - Assignment 1: Console I/O, random numbers, small classes
  - Assignment 2: File I/O, more testing and results involved
  - Assignment 3: Small class hierarchy, interfaces
  - Assignment 4: Graphical user interface (javaFX, Scene Builder)
- Three-hour final exam (45%)
## Resources

- **Course website**

- **Old course website (Winter 2018)**

- **One more website (Winter 2018)**
  - [http://sites.cs.queensu.ca/courses/cisc124/Section_1/Winter_2019/CISC124-Section_2](http://sites.cs.queensu.ca/courses/cisc124/Section_1/Winter_2019/CISC124-Section_2)

## Tools

- **JDK**
  - [jdk.java.net/11/](http://jdk.java.net/11/) (jdk-11.0.1 General Availability Release)
  - [https://docs.oracle.com/en/java/javase/11/](https://docs.oracle.com/en/java/javase/11/) (JDK 11 Documentation)

- **Eclipse**
  - `eclipse-inst-win64.exe` (Windows)

- **Optional**
  - Good text editor (i.e. Notepad++ for Windows users)

## Content Overview and Class Work

- **Content Overview**
  - Develop software systems with OO techniques
  - Solve medium complexity problems with OO techniques
  - Implement solutions with the OO facilities of Java
  - Class design, GUI development

- **Class Work**
  - Problem solving with OO techniques
  - Java OO constructs
  - Design of reusable classes
  - Reuse of Java libraries

## Java Environment

- **Java**
  - Pseudo-Compiled language (Source code to byte code)
  - Interpreted language (Byte code to machine language)
  - Dynamic and distributed (JVM: Java Virtual Machine)
  - Simple (plain syntax, straight constructs: classes, fields, methods)
  - Robust (no pointers, garbage collection, type checking)
  - Secure (no direct memory access, sandbox model, byte code verification)
  - Large repository of reusable libraries