Course Organization

CPSC 233: Introduction to Computer Science for Computer Science Majors II
Winter 2022

Jonathan Hudson, Ph.D.
Instructor
Department of Computer Science
University of Calgary

Sunday, 9 January 2022

Copyright © 2021

UNIVERSITY OF CALGARY

Welcome!

Jonathan Hudson, Ph.D

L01 MoWeFr 10:00 - 10:50 ENA 101

L02 MoWeFr 11:00 - 11:50 EEEL 161

(Zoom, recorded -> in-person after Jan 28th)

Office: ICT 712 (Return to it when classes in-person)

Office hours: 13:00-13:50 PM Tuesdays and Thursdays (Zoom, link in D2L) or by email-scheduled appointments. (using waiting room, not recorded)

jwhudson@ucalgary.ca

https://pages.cpsc.ucalgary.ca/~jwhudson/CPSC233W22/

Tutorials

Start next week!

Also through D2L, possibly recorded, however point is active interaction with TA for material, and assignment help.

Project demos will occur (for grading) in tutorials.

Use link to your tutorial only.

Your enrollment tutorial TA will mark your assignment/project material and they are only responsible for the students enrolled in their tutorial.

UNIVERSITY OF CALGARY

Computer Science Intro 2?

- We move from understanding computer, data, python language.
- To introducing structural design principals of object-oriented programming.
- We will also introduce skills like version control (Git), testing (junit), and graphical layout design for applications (JavaFx).
- Large focus of using a new language of Java, is a demonstration of how explicit typing, and structural use of object-oriented concepts can allow for larger programs to be designed.
- In some ways a miniature introduction to Software Engineering principles.
- You will have a project in partners that will help you see how these principals interact when more than one person is involved in a code base.

UNIVERSITY OF CALGARY

Calendar

From the calendar:

- “Emphasis on object-oriented analysis and design of small-scale computational systems and implementation using an object oriented language.
- Issues of design, modularization, and programming style will be emphasized.”

UNIVERSITY OF CALGARY

Course Goal

Course Outcomes:

- Describe the difference between procedural and object-oriented approaches to program decomposition.
- Apply the principles of object-oriented programming to design and document, using a standard modelling language, solutions to small-scale computational problems.
- Read, trace the execution, and determine the outcome of small software systems developed using objectoriented constructs including classes, objects, encapsulation, inheritance, and interfaces.
- Create and debug small software systems that make effective use of constructs including classes, objects, encapsulation, inheritance, and interfaces.
- Develop a client that makes use of external object-oriented libraries or application programming interfaces.
- Become familiar with a version control system and integrated development environment.

UNIVERSITY OF CALGARY

Course Goal

Course Outcomes:

- Describe the difference between **Python usage (without classes) and OO Java** approaches to program decomposition.
- Apply the principles of object-oriented programming to design and document, using a standard modelling language (**JavaFX**), solutions to small-scale computational problems.
- Read, trace the execution, and determine the outcome of small software systems developed using **Java** object-oriented constructs including classes, objects, encapsulation, inheritance, and interfaces.
- Create and debug (**Junit**) small software systems that make effective use of constructs including classes, objects, encapsulation, inheritance, and interfaces.
- Develop a client that makes use of external object-oriented libraries or application programming interfaces (**JavaFX**).
- Become familiar with a version control system (**Git**) and integrated development environment (**IntelliJ, Eclipse, Netbeans, etc.).**

UNIVERSITY OF CALGARY

Lectures

We will learn fundamentals of **object-oriented programming** using **Java**

We will cover (all the same things for 231 Python to start):

- Variables
- Arithmetic operations
- Conditions and Loops
- Functions
- Strings, Lists, ,Tuples, Sets, Dictionaries
- Files, Exceptions, Command Line Arguments

UNIVERSITY OF CALGARY

Lectures

We will learn fundamentals of **object-oriented programming** using **Java**

We will cover (new things):

- **Version control via Git**
- **Unit testing via JUnit**
- **Classes and Objects (but here we'll spend much more time)**
  - **Encapsulation**
  - **Inheritance**
  - **Interfaces**
  - **Larger Design**
- **GUI application via JavaFX**

UNIVERSITY OF CALGARY

Top Hat

- Download the top hat app on your smartphone got to TopHat on laptop.

- Create an account if you don't have one.
- Search for "Calgary" and select University of Calgary
- **Join Code: 280082**
- **https://app-ca.tophat.com/e/280082**
- **No marking or attendance records**
- **Might not make very much use of it unless I want to have some ideas I want to revisit in class**
- **Great for a first class get to know who is in class.**

Out of lecture?

There is no attendance taken at tutorials but they are highly recommended

- TAs will use classes to cover coding material in hands-on environment
- Material will be covered and there will also be assignment work/help
- **Projects will be demo'd to TA for a grade here!!!**

UNIVERSITY OF CALGARY

Grading

Component	Weighting %
Participation (5 of 6)	10%
Quizzes (5 of 6)	10%
Assignments (3)	10%, 10%, 10%
Project	50%

- Each piece of work (reports, assignments, quizzes, midterm exam(s) or final examination) submitted by the student will be assigned a grade. The student's grade for each component listed above will be combined with the indicated weights to produce an overall percentage for the course, which will be used to determine the course letter grade.
- The conversion between a percentage grade and letter grade is as follows.

	A+	A	A-	B+	B	B-	C+	C	C-	D+	D
Minimum % Required	95 %	90 %	85 %	80%	75%	70 %	65 %	60%	55%	50 %	45 %

UNIVERSITY OF CALGARY

Assignments

- Three individual assignment (30%) consists of programming questions
- Each assignment is due at 11:59 pm on the Friday due date.

Assignments	Due at 23:59
Assignment 1	Feb 11
Assignment 2	Mar 11
Assignment 3	Apr 1

UNIVERSITY OF CALGARY

Project

- Three demos (in tutorial time)
- Project final code is due due at 11:59 pm on the last day of lectures April 12th.

Project	Breakdown
Demos	10%, 10%, 10%
Reflection	5%
Final Code	15%

UNIVERSITY OF CALGARY

Course Policies

- When you email include your first name, and last name.
- Please use “CPSC233W22” as the prefix in the subject line

- There are no late submissions for participation, quizzes, or assignments. Submit early and double check after submitting. You can submit multiple times on D2L with no issue, so excuses will not be accepted.

UNIVERSITY OF CALGARY

Zoom Norms

- Respect others:
  - Keep your zoom muted unless asking a question. (please indicate in chat you have a question and I'll make time to let you ask)
  - Video is not necessary. However, for office hours and even smaller tutorials it is recommended.
  - You can ask questions via chat at any time. Ability to answer will be time and class pace dependent.
  - Arrive on time.
  - Refrain from using the chat for topics not related to the current material.
  - Use directed chat if you chat with someone you know. (Be aware that the directed chat is not private!)
  - Avoid any activity that might disturb your classmates.

UNIVERSITY OF CALGARY

Academic Dishonesty

- *“A single offence of cheating, plagiarism, or other academic misconduct, on term work, tests, or final examinations, etc., may lead to disciplinary probation or a student's suspension or expulsion from the faculty by the Dean, if it is determined that the offence warrants such action. ”*

- We have tools that let me quickly see if assignments appear to be highly similar and techniques like changing names, comments, and other details will not trick them.

- Please refer to the University Calendar for more details.

- **This course is fundamental and is essential for CS studies.**

UNIVERSITY OF CALGARY

Academic Dishonesty

- *All the work you submit must be your own.*
- *When you take algorithms or segments of code from somewhere else you must cite where you obtained them from.*
- *You need to understand all of the code in your work because the midterm and final are evaluating your understanding, not if you were able to make it work*

You will have a project partner. You will each have part of project where you will complete something by yourself to demonstrate your knowledge, you can of course discuss with your partner how you are going to do so, but you are still required to do that part yourself and will need to be able to demonstrate your knowledge of its working during your project demonstration during tutorial time.

For assignments/quizzes/participation/project reflection, these will be individual work submission. You must complete the work yourself and not copy from other students in any way.

UNIVERSITY OF CALGARY

Be Computer Science ‘Lazy’

Java API is your friend. I will reference if often and it is a common skill when you start learning a second language to be able to use API to understand a new language.

You can discuss understanding with other students, the TA, the instructor, etc. when learning a new topic. (assignments are still individual work, project is still limited to your pair of students and parts will be individual work as well).

UNIVERSITY OF CALGARY

Getting Help

- Do your part: Attend the lectures and tutorials
- Act early!
- First try it yourself→
  - Study the material carefully
  - Break the problem down
  - Try to narrow down the question
  - Search on google for your answer
- Still unclear?
- Ask your TA
- Come to my office ☺

UNIVERSITY OF CALGARY

Crisis line!

- If you think:
  - You suck at programming!
  - You suck at python!
  - You are not sure about this course!
  - You are OK with only a passing mark!!!
  - You tried but you didn’t understand!
- Come to my office → I’ll prove to you that you are wrong!
- Come early before things piled up!

UNIVERSITY OF CALGARY

Object-Oriented Programming

- Procedural programming was how you used Python up until we reached Classes/Objects
- Procedural programming looks like a bunch of steps (maybe hidden in functional) design but it reads a lot like
  - 1. do this
  - 2. do that
  - 3. now this
  - 4. go to function for next steps

  Etc.

UNIVERSITY OF CALGARY

Object-Oriented Programming

- Object-oriented design works on idea that we consider certain groups of data/concepts (like in CPSC 231 F21 with Stars in A3) as related
- We then encapsulate this concept in an object.
- We can also design these objects so that connect to each other. With inheritance, like Pet did with having a Dog/Cat/Fish subvariants in CPSC 231 F21. Or with relationships like a Lecture having a list of Students that are taking it.
- This design idea helps us limit what we need to think about (or keep in our head) when we code about on ‘concept’ this helps us design easier to understand, change, upkeep, and test code. This compartmentalization helps us scale code design to 10s, 100s, 1000s of people working with or on same code.

UNIVERSITY OF CALGARY

Why Java?

- Java is a widely used high-level programming language for general-purpose programming

- Design philosophy emphasizes **objects** and **explicit types**
  - No concern of whitespace indentation
  - Using {} codes blocks
- Explicit syntax (not always efficient)
  - Concepts take more lines of code
  - But its much clearer to us and compiler what we want the code to do

UNIVERSITY OF CALGARY

To do list

- An installation video for Java setup on your own computer will be posted this week in D2L
- Install Java (likely JDK 17) on your laptop
  - Tutorials will briefly introduce this but not spend all of next week's tutorial on setup (it will also introduce basic Java programming)
  - Recommend IntelliJ as an environment. (Eclipse might be used on UofC tutorial computers, can use Netbeans or other as well)
  - We will be using Scenebuilder 2 as an add in at end of course, so you may want to investigate for your different IDE if this will work for you (it does in IntelliJ and Eclipse)

UNIVERSITY OF CALGARY

Access to CPSC

- Java is your primary work environment for this course.
- Your code should run on Java JDK 16 which is the lab installation (we will teach Java 11 focused syntax in most cases) Newest JDK 17 will all you to run Java 16
- You can access the CPSC lab remotely. (no need to do this)
  - SSH (Secure Shell) allows you to establish a remote connection with the CPSC lab.
  - https://ucalgary.service-now.com/it?id=kb_article&sys_id=29aedd1bdb3e63c0d1b63ccb7c961963
- Please do not use any non-Linux-based CPSC server for this course.

UNIVERSITY OF CALGARY

Onward to … basics of Java!

Jonathan Hudson
jwhudson@ucalgary.ca
https://pages.cpsc.ucalgary.ca/~jwhudson/

UNIVERSITY OF CALGARY