CSc 133 Lecture Notes

1 – Course Introduction

Computer Science Department
California State University, Sacramento

Overview

- Classroom conduct
- Prerequisites
- Course topics
- Texts and references
- Grading: exams and programs
- Computers
- Communication
- Workload
- Ethics

Contacting Your Instructor

Dr. Pinar Muyan-Ozcelik

Office Hours: Wednesday & Friday 8:45 – 10:15 am
(held via Zoom,
see Canvas for the Zoom link)

Email: pmuyan@csus.edu

Webpage: https://ecs.csus.edu/~pmuyan

Class Modality

- CSC 133 is delivered using flipped-classroom methods this semester.
- Most weeks, we will meet at the scheduled class start times only once a week (mostly on Thursdays). Some weeks (like the first week of instruction) we will meet both on Tuesday and Thursday. **All class meetings will be in-person.**
- On the days we meet, a part of the (or the whole) lecture will be conducted in class and/or questions about the programming assignments and course topics will be discussed.

Class Modality (cont.)

- Our first meetings will be held on the lecture days of the first week of instruction. The other days that we will meet will be announced on Canvas. All meetings will begin at the scheduled class start time.

- However, the majority of the lectures will be completely pre-recorded and posted to Canvas (hence, on the days of those lectures, either we will not meet or if we will meet, questions about programming assignments or course topics will be discussed).

- **You are not allowed to make audio or video recordings of the class meetings and office hours.**

Recordings

- The majority of CSC 133 lectures will be completely pre-recorded. For the days that we will conduct a part of the lecture in a class meeting, the remaining part of the lecture will be pre-recorded.
- Recordings will be posted to the "Home" section of Canvas (under the "Announcements, Class Meetings, and Links to Lectures/Office Hours" module).
- Each recording will be available to be streamed starting the day of the lecture (on either Tuesday or Thursday) at 9:00 am.
  - make sure that you watch the lectures in a timely manner (e.g., on the day of the lecture), which is very important for your learning.

- **You are not allowed to save a copy of the recordings. Also, you are not allowed to share the links of recordings and office hours with anyone else.**

Classroom Etiquette

This course requires concentration and focus!

Please refrain from:

browsing, facebooking, social networking,
texting, instant messaging, tweeting, blogging,
gaming, during class…

Prerequisites

- CSc 130 (Algorithms and Data Structures)
- CSc 131 (Intro. to Software Engineering)

… which implies:

- o CSc 15 (Programming Methodology I)
- o CSc 20 (Programming Methodology II)
- o CSc 28 (Discrete Structures)
- o Math 29 (Pre-calculus Math)

Prerequisites By Topic

Programming Experience (review "Java Basics" in Appendices)

- 3 semesters in Java, C++, or similar OOP.
- Object-based principles: class/object definitions, method invocation, public vs. private fields, etc.
- Algorithms/data structures: lists, stacks, trees, hashtables, recursion

Software Engineering Topics

- Life Cycle: requirements, design, implementation, testing
- UML: Class, use-case, sequence diagrams

Math Topics (review "Vector/Matrix Algebra" in Appendices)

- Polynomial equations, trigonometric functions, matrix operations
- Cartesian coordinates, vectors, coordinate transformations

Repeat Policy

- A course cannot be repeated for a *third* – or subsequent – time (i.e., taken for a ***fourth* – or subsequent – time**) without a petition and our **College does not approve petitions**.
  - If you are taking CSC133 for the third time, be aware that you will not be able to take CSC133 for the fourth time at CSUS and you will need to take a similar course elsewhere and transfer it to CSUS.

What is this course about ?

Two main topics:

Advanced features of “O-O” paradigm

Introduction to Computer Graphics

Also covers:

Mobile App Development

First topic: **Object-Oriented Paradigm**

We will focus on how to write programs correctly!

- Language implementation:
  - Abstraction
  - Encapsulation
  - Inheritance
  - Polymorphism

- Tools supporting OOA/OOD/OOP:
  - formalisms such as *UML*
  - ***design patterns (underlying theme of CSC 133!)***

Second topic: **Computer Graphics**

- Devices and color models
- User interface (“GUI”) mechanisms
- Event-driven programming
- Basic line and polygon drawing
- Basic animation
- Object, World, Display coordinate systems
- Geometric transformations

Additional topic: **Mobile App Development**

- Introduction to Mobile App Development and CN1 (Codename One: Java-based, cross-platform mobile app development environment)

- Application of OOP and CG concepts to CN1:
  - CN1 code snippets will be provided in lectures
  - Assignments are required to be solved using CN1

Texts and References

- Required Texts:
  - CSc 133 Lecture Notes, Fall 2022, available at the "Home" section of Canvas
  - Codename One Developer Guide:

    CN1 Developer Guide - Revision 3.6 (pdf is available on Canvas)
  - Codename One JavaDocs of APIs: https://www.codenameone.com/javadoc/index.html

Texts and References (cont.)

- Recommended (**not Required**) Texts:
  - Object-Oriented Design & Patterns, 2nd Ed.,
    Cay Horstmann, John Wiley & Sons,
    ISBN 0-471-74487-5
  - Schaum's Outlines: Computer Graphics, 2nd Ed.,
    Xiang and Plastock, McGraw-Hill,
    ISBN 0-07-135781-5
- Supplemental material:
  - Basic Debugging With Eclipse: https://www.youtube.com/watch?v=PJWtO5wrptg

Grading

- Weighted Curve based on:

Programming Assignments (4)	45%
Midterm Exam	25%
Final Exam	30%

- Additional Criteria
  - Passing completion of **both**:
    - Programming assignments (combined)
    - Exams (Midterm + Final combined)

Grading (cont.)

Programming Assignments

- Required to be solved using CN1, submitted via Canvas
- Important tips will be given in class!
- There will be four (4) programming assignments
- They will be ***cumulative****!* Don't try to skip one!
- Late assignments are accepted **up until 10 days** past due date
- Late penalty: 5% per day, weekend days and holidays are counted
- Submissions can be updated **only** prior to the due date**:**
  - The version submitted right before the due date will be graded
  - If no such version exists, the version submitted right after the due date will be graded (as late assignment)
- *Individual* work
- Must keep a *backup* (soft) copy

Grading (cont.)

Exams

- Midterm exam date is noted on the syllabus
- Final exam date is noted on the outline
  - *scheduled by University*
- Study guides will be provided
  - *but only the course notes are complete!*
- Make-up exam is given only under extreme circumstances:
  - *generally requires prior arrangements*

Computers

- Work on any school machine (see the "Resources" module of the "Home" section of Canvas for the list of labs that you can use for CSC 133) or your machine which has CN1
- To install CN1:
  - Install Java SE JDK 11
  - Install Eclipse 2022-03 for Java Developers
  - Install CN1 as a plugin to the Eclipse
- Later in the semester:
  - Install JavaFX

(installation instructions will be discussed in class)

Communication

- Canvas:
  - Pre-recorded lectures and office hours (links are posted to the "Announcements, Class Meetings, and Links to Lectures/Office Hours" module of the "Home" section of Canvas)
  - assignments
  - announcements and messages
    - Check the "Home" section (under the "Announcements, Class Meetings, and Links to Lectures/Office Hours" module) and the "Discussions" section (under "Communication Forum" link) of Canvas
    - Subscribe to the Communication Forum (see the syllabus for details)
    - Announcements will also be made in class meetings
  - feedback and grades
- Check your SacLink email and Canvas daily

Workload

- "Freshman Counseling":
  - 1 unit = 1 hr/wk in class + 2-3 hrs/wk outside,
    on average, University-wide
  - 3 units = 9-12 hrs/wk,
    on average, University-wide
  - 12 units = 36-48 hrs/wk,
    on average, University-wide
- Not all classes are "average"!
- This is a programming-intensive course

Ethics

- Submitting work *constitutes an agreement* that *the work is solely your own*
- Students who violate the University policy on academic honesty are:
  - **Automatically Failed**
  - **Referred to the Dean of Students**
- Detailed Ethics policies given in syllabus and posted on Canvas

Ethics (cont.)

- You are allowed and encouraged to discuss assignments with other students in the class. Getting verbal advice/help from people who've already taken the course is also fine.
- Any reference to assignments from previous terms or web postings is unacceptable
- Any copying of non-trivial code is unacceptable
  - Non-trivial = more than a line or so
  - Includes reading someone else's code and then going off to write your own.

Questions?