Carleton University
Department of Systems and Computer Engineering
SYSC 2004 - Object-Oriented Software Development - Fall 2018
Course Outline

Instructor: C. Schramm
Email: cheryl.schramm@carleton.ca
Office: ME4444

1. Undergraduate Calendar Course Description

Designing and implementing small-scale programs as communities of collaborating objects, using a dynamically-typed or statically-typed programming language. Fundamental concepts: classes, objects, encapsulation, information hiding, inheritance, polymorphism. Iterative, incremental development and test-driven development.

Precludes additional credit for SYSC 1101, COMP 1006 and COMP 1406.
Prerequisite(s): SYSC 2006 or permission of the department..
Lectures three hours a week, laboratory two hours a week.

2. Course Objectives

- To learn the fundamental concepts of object-oriented programming (classes, objects, encapsulation, information hiding, inheritance, polymorphism).
- To gain experience applying these concepts by implementing small-scale programs as communities of interacting (collaborating) objects.
- To learn and apply some lightweight, modern techniques commonly used during object-oriented software development (iterative, incremental development; test-driven development).
- To develop the ability to build software experiments as an aid to learning.

3. Prerequisite

SYSC 2006 is the prerequisite for SYSC 2004. Students who have not received credit for either prerequisite course ***must withdraw from SYSC 2004*** by the last date for registration in Fall term courses; otherwise, they will be deregistered before the end of term.

4. Course Resources

Course material will be provided through CULearn. Lecture material, lab information, labs and assignments will all be posted on CULearn. Major announcements and errata will also be posted through CUlearn. You are expected to remain current with all information posted there.

5. Textbook

The **mandatory** course textbook is **Java Early Objects Fall 2018 at zyBooks.com.** Participation exercises will be assigned through this textbook so you will be unable to complete this course without a subscription and account to zyBooks.com

1. Sign in or create an account at learn.zybooks.com
2. Enter zyBook code **CARLETONSYSC2004SchrammFall2018**
3. Subscribe

A subscription is **$77**. Students may begin subscribing on Aug 21, 2018 and the cutoff to subscribe is Dec 07, 2018. Subscriptions will last until Jan 04, 2019.

5. Software

The Java programming environment to be used in the lab (and in assignments) is NetBeans. It is available for a free download at https://netbeans.org/downloads/.

All Java programming environments (Eclipse, NetBeans, BlueJ, IntelliJ and so on) run on top of the Java Development Kit (JDK). The JDK must be FIRST be separately downloaded and installed.

- Use Java 8 Standard Edition (SE) Development Kit
- http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html

6. Evaluation and Grading Scheme

If the student does not write both midterms and the final exam, the student earns an F in the course.

For those students who reach a passing grade on at least one of the two midterms or final exam, the marks will be calculated as follows.

Participation Exercises	10%	(In Zybooks)
Assignments:	20%	
Mid-term tests	30%	
Final exam:	40%	

For those students who do not reach a passing grade on at least one of the two midterms or final exam, the marks will be calculated as follows.

Mid-term tests	50%
Final exam:	50%

In other words, the marks for assignments and participation exercises will only be counted towards the final grade if the student can demonstrate passing at least one of the course's examinations by their own individual effort.

7. Participation Exercises

Each week, a selection of Participation and Challenge exercises from the Zybooks e-textbook will be assigned. The selection and deadline will be posted on CULearn. These exercises range from simple multiple choice questions to short programming snippets meant to test your understanding of a small focussed topic. The purpose of these exercises will be to prepare you for the next week's lectures, and to improve your basic understanding of the topics under study at that time.

8. Lab Periods

Attendance at the scheduled laboratory periods is ***mandatory***, and attendance will be taken. During the labs you will work on short programming exercises that are intended to help you understand particular concepts that have been introduced in the lectures.

There are no marks for the labs. Doing the labs is the best way to learn the course material and prepare for the exams. You are an adult learner, and responsible for your own decisions on how you pursue your education.

9. Assignments

There will be four programming assignments. Assignments are to be completed as homework. While labs are short exercises meant to introduce you to a particular programming construct, assignments are larger and more complex, requiring you to build up your experience and connect different topics together to solve a problem. Assignments are more open-ended and are subject more to interpretation – you have to understand the problem first and then figure out the suitable programming constructs to solve that problem. Assignments are meant to introduce you to the fun of building something that works!

Assignments will be posted approximately every two weeks, and you are expected to work on it over the course of at least one week. Assignments are not meant to be able to be completed in one evening. Assignments are valuable experiences in handling nuances yourself, in debugging, and in coping with larger amounts of code as well as increased code complexity.

Assignments are to be completed individually. Please take full advantage of your instructor, your TAs and your peer students to discuss ideas and approaches; however, it is imperative that you write your own code to ensure true learning. The marks for the assignments will not be included in the final grade unless you earn passing grades on the course's examinations.

Students are responsible for ***backing up their lab work*** before they leave the lab; for example, we recommend that you copy your files to a ***USB flash drive*** and to a cloud-based file hosting service; e.g., ***Google Drive***, ***Dropbox***, etc.

10. Exams

There will be ***two closed-book midterm tests.*** The dates will be announced in class and posted on cuLearn. Computers (and calculators) will **not** be used during the midterm test.

A closed-book final exam will be held during the University's ***December*** examination period. The *Academic Regulations of the University* permit instructors to specify requirements that must be satisfied for students to be eligible to write the final examination or, where circumstances warrant, the deferred final examination.

All students are eligible to write the final examination, regardless of the marks they received during the term.

The final examination is for evaluation purposes only and ***will not be returned to students***. You will be able to make arrangements with your instructor to see your marked final examination after the final grades have been made available. Your exam will not be remarked during this meeting and solutions to the exam questions will not be provided.

11. Attendance

Students are expected to attend all lectures and lab periods. The University requires students to have a conflict-free timetable. For more information, see the current Undergraduate Calendar, *Academic Regulations of the University,* Section 1.2, *Course Selection and Registration* and Section 1.5, *Deregistration*.

Requests to accommodate a missed midterm exam, lab periods, etc., because of conflicts with jobs or vacation plans will not be considered.

12. Missed Term Work

Students are advised to read the Calendar regulations under Section 2.6 of the Undergraduate calendar as it applies to missed term work. If you miss a midterm due to illness, you must provide suitable documentation within 3 working days (Required: date, expected day of recovery and the extent to which you were incapacitated during the time of the test). In this case, you will be assigned a date for the makeup midterm. If the makeup midterm is missed, then a failed grade is given and appeals must be made formally through the Registrar's office (with decisions coming after the term).

Requests for accommodation because of poor performance on the midterm test will not be considered.

13. Email

As required by University regulation, students are required to read their Carleton email daily. The instructor will be using this email account to communicate important and timely messages. Not reading these emails is not a valid excuse for missing a deadline or a requirement.

Emails to the instructor must be professional in nature and in writing. All emails must be from the student's Carleton account. The subject must contain the course name. If the email concerns the lab, the subject must also contain the lab section. If the email concerns grades, the body must contain the student's name **and student number**.

Example Subject: SYSC 2004 – Midterm

Example Subject : SYSC 2004 – Section A1, Lab 3

The instructor attempts to answer all emails in a timely and helpful manner, within two working days. ***Replies will not be sent to emails concerning matters already covered in class, or posted on the course website or discussion groups or news posts.***

14. Academic Integrity

Students should be aware of their obligations with regards to academic integrity. Please review the information about academic integrity provided at the Office of Student Affairs Web site:
http://carleton.ca/studentaffairs/academic-integrity
This site also contains a link to the complete Academic Integrity Policy that was approved by the University's Senate.

15. Intellectual Property

Classroom teaching and learning activities, including lectures, labs, discussions, presentations, etc., by both instructors and students, are copy protected and remain the intellectual property of their respective author(s). All course materials, including course outlines, lecture and lab materials, tests and exams, and other materials, are also protected by copyright and remain the intellectual property of their respective author(s).

Students registered in this course may take notes and make copies of course materials for their own educational use only. Students are not permitted to reproduce or distribute lecture notes and other course materials publicly for commercial or non-commercial purposes without express written consent from the copyright holder(s).

16. Academic Accommodations

Requests for Academic Accommodation
You may need special arrangements to meet your academic obligations during the term. For an accommodation request, the processes are as follows:

Pregnancy obligation
Please contact your instructor with any requests for academic accommodation during the first two weeks of class, or as soon as possible after the need for accommodation is known to exist. For more details, visit the Equity Services website: **carleton.ca/equity/wp-content/uploads/Student-Guide-to-Academic-Accommodation.pdf**

Religious obligation
Please contact your instructor with any requests for academic accommodation during the first two weeks

of class, or as soon as possible after the need for accommodation is known to exist. For more details, visit the Equity Services website: **carleton.ca/equity/wp-content/uploads/Student-Guide-to-Academic-Accommodation.pdf**

Academic Accommodations for Students with Disabilities
If you have a documented disability requiring academic accommodations in this course, please contact the Paul Menton Centre for Students with Disabilities (PMC) at 613-520-6608 or pmc@carleton.ca for a formal evaluation or contact your PMC coordinator to send your instructor your Letter of Accommodation at the beginning of the term. You must also contact the PMC no later than two weeks before the first in-class scheduled test or exam requiring accommodation (if applicable). After requesting accommodation from PMC, meet with your instructor as soon as possible to ensure accommodation arrangements are made. **carleton.ca/pmc**

Survivors of Sexual Violence
As a community, Carleton University is committed to maintaining a positive learning, working and living environment where sexual violence will not be tolerated, and is survivors are supported through academic accommodations as per Carleton's Sexual Violence Policy. For more information about the services available at the university and to obtain information about sexual violence and/or support, visit: **carleton.ca/sexual-violence-support**

Accommodation for Student Activities
Carleton University recognizes the substantial benefits, both to the individual student and for the university, that result from a student participating in activities beyond the classroom experience. Reasonable accommodation must be provided to students who compete or perform at the national or international level. Please contact your instructor with any requests for academic accommodation during the first two weeks of class, or as soon as possible after the need for accommodation is known to exist. **https://carleton.ca/senate/wp-content/uploads/Accommodation-for-Student-Activities-1.pdf**

For more information on academic accommodation, please contact the departmental administrator or visit: **students.carleton.ca/course-outline**

17. Health and Safety:

Every student should have a copy of our Health and Safety Manual. An electronic version of the manual can be found at http://www.sce.carleton.ca/courses/health-and-safety.pdf

18. Tentative week-by-week Schedule

Week 1: Review of Classes, Objects & Arrays.
Introduction to the Zybooks and to the Netbeans Integrated Development Environment

Week 2: Java class definitions: fields (instance variables), constructors, instance methods. Accessor and mutator methods. UML representation of classes and objects.

Week 3: Classes as types. The `new` operator. References to objects. Interacting objects. Grouping objects: array objects. Generic collections: class `ArrayList` and `HashMap`.

Week 4: Using classes from Java's library (class `String`, wrapper classes), packages, reading and writing documentation (javadoc) comments. Unit testing (JUnit) and debugging. Class variables

and methods (the `static` reserved word).

Week 5: Inheritance

Week 6: Polymorphism and Abstract Classes

Week 8: Inheritance, continued: method overriding, dynamic method lookup, method polymorphism, the `super` call. Overriding methods inherited from `Object` (`equals`, `toString`). Handling errors with exceptions, the exception class hierarchy.

Week 8 : Inheritance, continued: abstract classes, interfaces. Comparison of inheritance and composition as techniques for object-oriented design.

Week 9: Introduction to design patterns. The observer pattern and event handling.

Week 10 onwards: Graphical User Interfaces