

CSCI 2132 — Software Development Course Syllabus

Instructor Information

Instructor:	Dr. Vlado Keselj	Office:	432 (G.CS bldg.)
E-mail:	vlado@dnlp.ca	Office Hours:	Open-door policy or appointment by e-mail
Lectures Time:	MWF 12:35-13:25	Location:	Chemistry 125
Labs Time:	Lab Section B01, Thursdays 08:35-09:55 in Goldberg CS 143 (TLab 2) Lab Section B02, Thursdays 08:35-09:55 in Goldberg CS 133 (TLab 1) Lab Section B03, Thursdays 08:35-09:55 in LSC-Common-Area 220 Lab Section B04, Thursdays 10:05-11:25 in Goldberg CS 143 (TLab 2) Lab Section B05, Thursdays 10:05-11:25 in Goldberg CS 133 (TLab 1) Lab Section B06, Thursdays 11:35-12:55 in Goldberg CS 143 (TLab 2)		
Course Homepage:	http://www.cs.dal.ca/~vlado/csci2132		
Course Mail List:	csci2132@lists.dnlp.ca		

Important Dates

Please check the course calendar on the web site for more details.

1. Term starts: Tue Sep 4, 2018
2. Last day to add classes: Tue Sep 18, 2018
3. Midterm Exam I (tentative): Thu Sep 27, 2018
4. Last day to drop class without “W”: Mon Oct 1, 2018
5. No class, Thanksgiving: Mon Oct 8, 2018
6. Last day to drop class with “W”: Tue Oct 30, 2018
7. Midterm Exam II (tentative): Thu Nov 8, 2018
8. No class, in lieu of Remembrance Day: Mon Nov 12, 2018
9. Fall Study Break (no classes): Nov 12–16, 2018
10. Term ends: Tue Dec 4, 2017 (Monday classes held)
11. Final Exam: TBA, it will be a 3h exam in the period of Dec 6 to 16, 2018

Course Description

This course introduces programming and software development techniques in a procedural language. In particular, the C programming language and the UNIX-style operating systems will be used to teach program design paradigms, source code management, software testing, debugging, scripting, and other techniques useful for software development.

Learning Outcomes

- Use the command line of the UNIX-style environment.
- Describe the various testing methodologies and their purpose.
- Describe the life cycle of a software project.
- Explain the role and function of build tools.
- Execute the program development cycle starting with a problem specification
- Use pointers in C and manage memory.
- Design a program to solve a problem of moderate complexity given a problem specification.

- Design regular expressions and use tools such as grep and sed to manipulate text streams.
- Explain the C build cycle (preprocess, compile, assemble, link).
- Select an appropriate testing methodology given a problem specification.
- Use the UNIX command-line environment to perform all parts of the software development cycle.
- Write short (50 line) shell scripts to solve a simple problem such as running regression tests.
- Use a standard build tool, such as 'make', to create makefiles, given the source dependencies.
- Write command pipe-lines comprising multiple tools in the UNIX environment to solve simple problems.
- Describe the various kinds of software errors that can occur and their causes.
- Implement a test suite for a given module, program, function, or library.
- Implement moderately complex programs in C.
- Implement simple data structures (linked lists, queues, stacks) in C.
- Explain the purpose of revision control systems.
- Explain the role of software testing and testing methods.
- Identify methodologies for dealing with various software errors.
- Select appropriate methodologies given a piece of software and list of exhibited symptoms.
- Identify and rectify software errors within a piece of software of moderate complexity.

Class Format and Course Communication

- Content will be delivered using a combination of lectures and hands-on labs
- Students must ask the instructor permission before recording class lectures.
- Course announcements will be posted to the course mail list, which comprises the instructor's, TAs', and students' Dal emails. It is the student's responsibility to check their Dal e-mail on a daily basis. To access your Dal e-mail account please see: <https://www.dal.ca/dept/its/o365/services/email.html> or you can contact the CS help desk. If you prefer to use other email address, please forward your Dal email, or request that the instructor adds your email to the course email list.

Evaluation Criteria

1. Assignments (30%)
 - Tentatively 7-10 assignments, best $n - 1$ used for grading if $n > 6$
 - **Late assignments will not be accepted.**
 - Assignments will be submitted electronically; exceptions possible
 - Assignments may include one or two practicums.
2. Two Midterm Exams (20%)
3. Final Exam (50%)
 - Scheduled by the university.
 - Will cover all material in the course.
 - Midterms may be ignored if better mark is obtained by ignoring them and counting 70% for the Final Exam

Midterm and Final Exam Requirements

- Photo ID is required.
- Closed book with allowed "cheat sheet" of 2 hand-written or printed pages.
- No dictionaries, notes, calculators, cell phones, PDAs, talking slide rulers, or other electronic aids allowed.

Required Texts and Resources

- *C Programming: A Modern Approach*, by K. N. King, W. W. Norton & Company, 2008.
- *UNIX for Programmers and Users*, by Graham Glass and King Ables, Prentice Hall, 2003.

Additional Recommended Reading

- *Unix and Linux System Administration Handbook*, by Evi Nemeth, Garth Snyder, Trent R. Hein, Ben Whaley, edition 4th Edition, Pearson Education, 2010.
- *The C Programming Language*, by Brian W. Kernighan and Dennis M. Ritchie, edition 2, Prentice Hall Software Series, 1988.

Prerequisites

CSCI 1101 or suitable prior programming experience

Tentative List of Topics

1. Course Introduction
2. Fundamentals of UNIX Operating System
 - History and Basic commands and utilities
 - Files and Directories
 - Editors, Shells, Regular expressions
3. C Programming Language and Software Development
 - Introduction to C, input and output
 - Operators, expressions, and statements
 - Software development life cycle
 - Arrays and Functions
4. Program Organization and Dynamic Memory Allocation
 - Writing large programs, make
 - Pointers and dynamic memory allocation
 - Linked lists and dynamic arrays
5. Shell Scripting and Control Version Systems
 - Shell Scripting and File Manipulation
 - Control Version Systems

Responsible Computing Policy

Usage of all computing resources in the Faculty of Computer Science must be within the Dalhousie Acceptable Use Policies (<http://its.dal.ca/policies/>) and the Faculty of Computer Science Responsible Computing Policy. (https://www.cs.dal.ca/downloads/fcs_policy_local.pdf)

Culture of Respect

Every person has a right to respect and safety. We believe inclusiveness is fundamental to education and learning. Misogyny and other disrespectful behaviour in our classrooms, on our campus, on social media, and in our community is unacceptable. As a community, we must stand for equality and hold ourselves to a higher standard.

What we all need to do ¹:

1. **Be Ready to Act:** This starts with promising yourself to speak up to help prevent it from happening again. Whatever it takes, summon your courage to address the issue. Try to approach the issue with open-ended questions like “Why did you say that?” or “How did you develop that belief?”
2. **Identify the Behaviour:** Use reflective listening and avoid labeling, name-calling, or assigning blame to the person. Focus the conversation on the behaviour, not on the person. For example, “The comment you just made sounded racist, is that what you intended?” is a better approach than “You’re a racist if you make comments like that.”
3. **Appeal to Principles:** This can work well if the person is known to you, like a friend, sibling, or co-worker. For example, “I have always thought of you as a fair-minded person, so it shocks me when I hear you say something like that.”

¹Source: Speak Up! ©2005 Southern Poverty Law Center. First Printing. This publication was produced by Teaching Tolerance, a project of the Southern Poverty Law Center. Full “Speak Up” document found at: <http://www.dal.ca/dept/dalrespect.html> Revised by Susan Holmes from a document provided April 2015 by Lyndsay Anderson, Manager, Student Dispute Resolution, Dalhousie University 902.494.4140 lyndsay.anderson@dal.ca www.dal.ca/think.

4. **Set Limits:** You cannot control another persons actions, but you can control what happens in your space. Do not be afraid to ask someone “Please do not tell racist jokes in my presence anymore” or state “This classroom is not a place where I allow homophobia to occur.” After you have set that expectation, make sure you consistently maintain it.
5. **Find or be an Ally:** Seek out like-minded people that support your views, and help support others in their challenges. Leading by example can be a powerful way to inspire others to do the same.
6. **Be Vigilant:** Change can happen slowly, but do not let this deter you. Stay prepared, keep speaking up, and do not let yourself be silenced.

University Statements

This course is governed by the academic rules and regulations set forth in the University Calendar and the Senate.

<https://academiccalendar.dal.ca/Catalog/ViewCatalog.aspx?pageid=viewcatalog&catalogid=69&chapterid=3457&loaduserredits=False>

Academic Integrity

At Dalhousie University, we are guided in all of our work by the values of academic integrity: honesty, trust, fairness, responsibility and respect (The Center for Academic Integrity, Duke University, 1999). As a student, you are required to demonstrate these values in all of the work you do. The University provides policies and procedures that every member of the university community is required to follow to ensure academic integrity.

http://www.dal.ca/dept/university_secretariat/academic-integrity.html

Accessibility

The Advising and Access Services Centre is Dalhousie’s centre of expertise for student accessibility and accommodation. The advising team works with students who request accommodation as a result of: a disability, religious obligation, or any barrier related to any other characteristic protected under Human Rights legislation (NS, NB, PEI, NFLD).

http://www.dal.ca/campus_life/student_services/academic-support/accessibility.html

Student Code of Conduct

Everyone at Dalhousie is expected to treat others with dignity and respect. The Code of Student Conduct allows Dalhousie to take disciplinary action if students dont follow this community expectation. When appropriate, violations of the code can be resolved in a reasonable and informal mannerperhaps through a restorative justice process. If an informal resolution cant be reached, or would be inappropriate, procedures exist for formal dispute resolution.

https://www.dal.ca/campus_life/safety-respect/student-rights-and-responsibilities/student-life-policies/code-of-student-conduct.html

Diversity and Inclusion – Culture of Respect

Every person at Dalhousie has a right to be respected and safe. We believe inclusiveness is fundamental to education. We stand for equality. Dalhousie is strengthened in our diversity. We are a respectful and inclusive community. We are committed to being a place where everyone feels welcome and supported, which is why our Strategic Direction prioritizes fostering a culture of diversity and inclusiveness (Strategic Priority 5.2).

<http://www.dal.ca/cultureofrespect.html>

Recognition of Mikmaq Territory

Dalhousie University would like to acknowledge that the University is on Traditional Mikmaq Territory. The Elders in Residence program provides students with access to First Nations elders for guidance, counsel and support. Visit the office in the McCain Building (room 3037) or contact the programs at elders@dal.ca or 902-494-6803 (leave a message).

Learning and Support Resources

General Academic Support — Advising http://www.dal.ca/campus_life/student_services/academic-support/advising.html

Fair Dealing Guidelines <https://libraries.dal.ca/services/copyright-office/guidelines/fair-dealing-guidelines.html>

Dalhousie University Library <http://libraries.dal.ca>