

## Cmpt214

### Programming Principles and Practice

Term 1 (Fall), 2017/18

# Syllabus

---

## Index

- [Description](#)
  - [Prerequisites](#)
  - [Scheduling](#)
  - [Website](#)
  - [Instructor Information](#)
  - [Course Objectives](#)
  - [Student Evaluation](#)
  - [Texts and Library Materials](#)
  - [Topic Outline](#)
  - [Lecture Schedule](#)
  - [Instructional Laboratory](#)
  - [Assignments](#)
  - [Laboratory Resources](#)
  - [Policies](#)
  - [Student Supports](#)
- 

## Description

The purpose of this course is to broaden the student's view of software development. Topics include the C programming language, scripting languages, libraries, and tools and techniques for program development and maintenance.

A theme for this course is "quality programming in the small". The idea is to teach the tools and techniques to write a quality component, whether it is a stand-alone program to do some small task, or it is to be a part of a larger project. These are fundamental principles/skills required by professional programmers.

The University course catalogue description of the class is as follows:

A hands-on approach to software development at the individual and small-team level. Application of software tools — including scripting languages, system utilities and libraries — for construction of small software systems. Integrated with and motivated by programming practices, system development, testing and maintenance issues.

---

# Prerequisites

CMPT 145 or 115 or 117

---

## Scheduling

Class day & time: Tuesday and Thursday, 11:30 a.m. to 12:50 p.m.

Class Location: Thorv 105

Class duration: September 7 through December 7, 2017

Midterm Exam: outside of class, from 17:30 to 19:00 (5:30 p.m. to 7:00 p.m.) on Monday, October 30, 2017 (the last day for withdrawals is November 15)

Labs: Section T01, Fridays, 09:00 - 10:20  
Section T03, Tuesdays, 16:00 - 17:20  
Section T05, Thursdays, 13:00 - 14:20  
Section T07, Fridays, 14:30 - 15:50  
Section T09, Mondays, 09:00 - 10:20

Lab Location: All tutorials are held in room S311 in the Spinks Addition of the Thorvaldson Building.

Labs start: Friday, September 8

Help Desk: Thursdays, 14:30 - 16:00, Room S352

---

## Website

The website for this course is on the [moodle server of the Department of Computer Science](https://moodle.cs.usask.ca/course/view.php?id=520). The URL is <https://moodle.cs.usask.ca/course/view.php?id=520>

---

## Instructor Information

Dr. Tony Kusalik, Computer Science

email: [kusalik@cs.usask.ca](mailto:kusalik@cs.usask.ca) or [tony.kusalik@usask.ca](mailto:tony.kusalik@usask.ca) (do not cut-and-paste these addresses as they have hidden, "garbage" characters in them)

office: Thorv S424 (in the Spinks Addition)

telephone: 966-4904

## Office Hours

The professor is typically available immediately before and after the class for questions, advice, etc.

The professor has combined advising time / office hour every Friday between 11:30 and 12:30. At this time, he can be found in his office, Thorv S424. You can set up an appointment during this time, or you can just "drop in". Priority will be given to students who have made an appointment.

The prof is available to meet students at other times. Simply send an email message and set up an appointment!

Finally, remember that, if you need help, e-mail works 24 hours a day — and you'll probably even get a response in short order! Alternatively post something to one of the class forums on [moodle](#).

## Teaching Assistant

Name	Sections	E-mail
Marina Schmidt (for T01, T03, and T07)		<a href="mailto:mts299@mail.usask.ca">mts299@mail.usask.ca</a>
Dylan McInnes (for T05 and T09)		<a href="mailto:dym533@mail.usask.ca">dym533@mail.usask.ca</a>

---

## Course Objectives

By the completion of this course, students will be expected to:

- be able to write modest programs in the C programming language;
- be familiar with many common UNIX/LINUX commands;
- understand fundamental concepts regarding operating systems, version control systems, the software built process via compilation and linking, portable representation of characters, representation and storage of program variables in memory, UNIX processes, and the UNIX file system.

By the completion of this course, students will be expected to be able to:

- skillfully use complex UNIX/LINUX shell commands;
- write non-trivial scripts in UNIX shell (*bash*) and *awk*;
- search files using the *grep* command and regular expressions;
- write bug-free moderately complex C programs;
- effectively use SVN commands to manage the evolution of a non-trivial program;
- decompose a large program into cohesive, separately compiled modules and to manage the compilation and linking of those modules into a single executable using a "makefile";
- use a line-oriented C/C++ debugger to examine the execution of a program and identify bugs in the code;
- use more sophisticated debugging and testing techniques than used in first year;
- build and use an object module library.

Note that the above two lists are not exclusive.

---

## Student Evaluation

### Grading Scheme

participation	2%
lab exercises	8%
assignments	20%
midterm exam	25% (scheduled for October 30, 2017 <i>outside of regular class time</i> )
final exam	45%

## Important Notes

- Because of the characteristics of the classroom, and the general unavailability of large rooms on campus between 11:30 and 12:50 on Tuesdays or Thursdays, it was necessary to schedule the midterm exam outside of class time. The exam will be held at 17:30 (5:30 p.m.) on Monday, October 30. The room for the exam has not been determined yet. It will be announced well in advance of the exam. Students who cannot be available at this time must contact the instructor at least one week in advance of the class to write the exam at an alternate time on October 30. Such students will have to sign a "Declaration to Secrecy" form.
- The participation grade will be based on two types of evidence: 1% for any constructive and/or contributive involvement in (postings to) the on-line discussion forums for the class; and 1% for participation at least once in interactive polling conducted during class.
- There will be 12 lab exercises, the first being made available on September 8. There will also be 4 assignments, the first being made available near the end of September. See sections of this syllabus on "[Instructional Laboratory](#)" and "[Assignments](#)", respectively, for more information.
- If a student does not write the midterm exam, the corresponding portion of the final grade allocation will be re-assigned to the final exam.
- If students know in advance that they will miss a lab, an assignment, or exam due to extenuating circumstances, they should contact the professor well in advance so that alternate arrangements can be made.
- The midterm exam will be 80 minutes in duration and "closed book". The final exam will be 3 hours in duration and "open cheat sheet". The latter means that students may bring to the exam a single 8.5×11-inch quick reference sheet ("cheat sheet") of their **own** compilation. Students needing special aids for writing exams (e.g. DSS students) must make arrangements prior to the exam with the instructor for presence of these aids. Other than the aforementioned items, no calculators or other supplementary resources will be permitted at the exams.

Midterm and final exams from previous offerings of the class by the same professor will be made available to students prior to exam dates. However, answer keys to the exams will not be made available.

- Late assignments and labs will be accepted, though they will be docked a penalty. The penalties will be determined by exponential functions, the specifics of which are given under the Assignments and Labs sections, respectively, of the [moodle pages](#) for the class.
- Once solutions for labs or assignments have been posted, any late lab or assignment will be credited as having been completed but will receive an automatic grade of 0. This will be the case unless special arrangements with the instructor have been made.
- Due to the large enrolment in the class and limited budgets for markers, it may sometimes be necessary to not mark all the questions on submitted labs and assignments. In such circumstances, the determination of which questions will not be marked will be made on a post-hoc basis. That means it will not be announced in advance. However, solution sets will be provided for any question which is not graded.

## Criteria That Must Be Met To Pass

- Laboratory exercises and assignments are required (mandatory) elements of the class. Therefore, in accordance with the University's "[Academic Courses Policy on Course Delivery, Examinations & Assessment of Student Learning](#)" at least three quarters of the lab exercises and assignments must be completed in order to achieve a passing grade. For instance, students who skip the labs or don't bother doing the assignments will not pass the class no matter how well they "ace" the final exam. However, students may be exempted from having to complete a particular lab or assignment due to extenuating

circumstances on a case-by-case basis (e.g. for medical reasons or serious personal matters). Students should contact the instructor to set up special provisions for the lab or assignment as soon as they are able.

For the purposes of the University's "[Academic Courses Policy on Course Delivery, Examinations & Assessment of Student Learning](#)", if more than one assignment is missed and no exemptions have been arranged, a student will be deemed to have not successfully completed a required element of the class (the assignment portion). This, in turn, will mean that the student will not receive a passing grade in the course. That is, unless special provisions have been arranged, no more than one assignment can be missed for a student to pass the class.

For the purposes of the University's "[Academic Courses Policy on Course Delivery, Examinations & Assessment of Student Learning](#)", if more than three labs are missed and no exemptions have been arranged, a student will be deemed to have not successfully completed a required element of the class (the laboratory exercises) and will not receive a passing grade in the course. That is, unless special provisions have been arranged, no more than three labs can be missed for a student to pass the class.

- The final exam is another required element of the class. Failure to write the final exam will result in failure of the course.

## Final Exam Scheduling

The Registrar schedules all final examinations, including deferred and supplemental examinations. Students are advised not to make travel arrangements for the exam period until the official exam schedule has been posted.

---

## Texts and Library Materials

- Required: [Practical Guide To Linux Commands, Editors, And Shell Programming, by Sobell, 2012, ISBN: 978-0-13-308504-4](#). The [Library also has this an earlier edition of the book](#) and the text has a [website](#).
- Recommended: [C++: The Complete Reference, 4th Edition, by Schildt, 2002, ISBN: 978-0-07-222680-5](#). The [Library has a copy of this book](#).  
[The Elements of C++ Style, by Misfeldt, Bumgardner, and Grey, ISBN: 978-0-521-89308-4](#).
- Alternatives: [Unix : the textbook, by Sarwar, Koretsky, and Sarwar, 2005, ISBN 032122731X](#). [Available from the UofS Library](#).  
[The C programming language, by Kernighan & Ritchie, 1988, ISBN 0201539926](#). [Available from the UofS Library](#).  
[The C++ programming language, by Stroustrup, 1991, ISBN 0201539926](#). [Available from the UofS Library](#).  
[C++ and C debugging, testing, and reliability, by Spuler, 1994, ISBN 0133081729](#). [Available from the UofS Library](#).  
[Practice Of Programming, by Kernighan & Pike, 1999, ISBN 978-0-201-61586-9](#). [Available from the UofS Library](#).  
[The pragmatic programmer: From journeyman to master, by Hunt & Thomas, 2000, ISBN 978-0-201-61622-4](#). [Available from the UofS Library](#).  
[C Quick Syntax Reference, by Olsson, 2015, ISBN 9781430265009](#). [Electronic resource available through the UofS Library](#).

Note that the Sobell and Schildt texts have been listed as texts for this class for several years, so used copies may be available. Earlier editions would be more than sufficient.

---

## Topic Outline

The following is an outline of the main topics to be covered in the class

- LINUX/UNIX commands
  - Bourne shell scripting
  - Regular Expressions
  - AWK programming
  - C Programming
  - Testing
  - Debugging
  - Separate compilation and multi-file development
  - Version control systems
  - Profiling and performance optimization
  - Portability
- 

## Lecture Schedule

The following is a tentative class schedule. It lists at what stage in the class a specific topic will be covered. Note that the schedule is approximate and subject to change.

<b>Week</b>	<b>Topic</b>
1 (partial)	Class Introduction
	Basic UNIX Shell and UNIX Commands
	More Sophisticated Use of Unix
2	UNIX File System Fundamentals
	File-Oriented Commands
	Filename Wildcards
	Input/Output Redirection
3	UNIX Process Abstraction
	Process-Oriented Commands
	Preventing Shell Interpretation
	Regular Expressions and the <i>grep</i> command
4	Regular Expressions versus File Name Wildcards
	Extended Regular Expressions
	Basic C Language Constructs
5	C Language Constructs

	Basic I/O in C
6	Data Representations in Memory
	C Language Constructs
7	Intermediate C Language Constructs
	Advanced C Language Constructs
8	Steps in a Program Build
	Storage Allocation in Runtime Memory
	Debugging
	Midterm Exam
9	Debugging
	Multi-file Development
	Makefiles
10	Version Control
	Object Module Libraries
	Compound UNIX Shell Commands
11	<i>awk</i> and <i>awk</i> scripts
	Shell Programming and Shell Scripts
12	Shell Programming and Shell Scripts
	Performance and Profiling
	Performance and Profiling
13	Portability
	Software Distribution

---

## Instructional Laboratory

Laboratory sections and times are as follows:

Section	Day	Time
T01	Fri	09:00-10:20
T03	Tues	16:00-17:20
T05	Thur	13:00-14:20
T07	Fri	14:30-15:50
T09	Mon	09:00-10:20

All tutorials are held in room S311 in the Spinks Addition of the Thorvaldson Building. If there are insufficient seats in S311, Cmp214 may use the workstations in S315. Cmp214 students have priority for access to the latter workstations during Cmp214 lab times. Labs will start on the week of September 8, and there will be 12 lab exercises.

The laboratory session is for practical experience in the application of concepts introduced in the lectures. The lab exercises themselves are designed such that students should be able to complete them in a relatively short amount of time. They are for less work than the assignments in class, for instance.

Lab periods are 80 minutes in length. An initial portion of the lab time may be devoted to providing introductory information. The remainder of the period will be for completion of the laboratory exercise. The lab instructor will be available during this time to provide assistance as necessary.

Students are to attend one laboratory session each week. If the day for a student's tutorial/lab section falls on a statutory holiday, the student should attend a lab on an alternate day.

Each week's lab component is to be completed in the time period specified and a lab submission will be required. Each student's lab solution is to be submitted electronically.

Lab exercises will be made available on Friday of each week. The solution to the exercises will be due the following Thursday night. Thus a lab "week" will run from Friday to Thursday.

## Important Notes

- Laboratory exercises are a required (mandatory) element of the class. Therefore, in accordance with the University's ["Academic Courses Policy on Course Delivery, Examinations & Assessment of Student Learning"](#), lab exercises must be completed and submitted in order to achieve a passing grade. However, students may be exempted from having to complete a particular lab due to extenuating circumstances on a case-by-case basis (e.g. for medical reasons or serious personal matters). Alternatively, the student may be given an extension on the due date. Students should contact the instructor to set up special provisions for the lab as soon as they are able.
- For the purposes of the University's ["Academic Courses Policy on Course Delivery, Examinations & Assessment of Student Learning"](#), if more than three labs are missed and no exemptions have been arranged, a student will be deemed to have not successfully completed a required element of the class (the laboratory exercises) and will not receive a passing grade in the course. That is, unless special provisions have been arranged, no more than three labs can be missed for a student to pass the class.
- Late labs will be accepted, though they will be docked a penalty. The penalties will be determined by an exponential function, the specifics of which are given under the Labs section of the [moodle pages for the class](#).
- Once solutions for labs have been posted, any late lab or assignment will be credited as having been completed but will receive an automatic grade of 0. This will be the case unless special arrangements with the instructor have been made.
- There is no way for Departmental staff to verify date-stamps on files on home or personal computers. Therefore, students who do their work on a personal or home computer should keep a copy of their submission on the departmental servers. This greatly assists in reconciling situations where students make accidental errors in their lab submissions.

---

## Assignments

There will be approximately 4 assignments in the class. They will be harder and take more effort than the exercises in the Laboratory. Students will typically have 2 to 3 weeks to complete each assignment. Assignment solutions will be submitted electronically.

## Important Notes

- Assignments are required (mandatory) element of the class. Therefore, in accordance with the University's ["Academic Courses Policy on Course Delivery, Examinations & Assessment of Student Learning"](#), assignments must be completed in order to achieve a passing grade. However, students may

be exempted from having to complete a particular assignment due to extenuating circumstances on a case-by-case basis (e.g. for medical reasons or serious personal matters). Alternatively, the student may be given an extension on the due date. Students should contact the instructor to set up special provisions for the assignment as soon as they are able.

- For the purposes of the University's "[Academic Courses Policy on Course Delivery, Examinations & Assessment of Student Learning](#)", if more than one assignment is missed and no exemptions have been arranged, a student will be deemed to have not successfully completed a required element of the class (the assignment portion). This, in turn, will mean that the student will not receive a passing grade in the course. That is, unless special provisions have been arranged, no more than one assignment can be missed for a student to pass the class.
  - Late assignment submissions will be accepted, though they will be docked a penalty. The penalties will be determined by an exponential function, the specifics of which are given under the Assignments section of the [moodle pages for the class](#).
  - If an assignment submission consists of multiple files, the last one uploaded establishes the time-stamp for the submission.
  - Once solutions for labs have been posted, any late lab or assignment will be credited as having been completed but will receive an automatic grade of 0. This will be the case unless special arrangements with the instructor have been made.
  - There is no way for Departmental staff to verify date-stamps on files on home or personal computers. Therefore, students who do their work on a personal or home computer should keep a copy of their submission on the departmental servers. This greatly assists in reconciling situations where students make accidental errors in their assignment submissions.
- 

## Laboratory Resources

The main computation resources for this course will be the LINUX operating system as provided by the `tuxworld` (`tuxworld.usask.ca`) set of computational servers and the Mac OS X operating system provided by the workstations in the [Computer Science instructional laboratory](#) in rooms Thorv S311 and S315. The LINUX workstations in rooms Thorv S315 and S360 may also be used. Often, students will be able to use either LINUX or Mac OS X. When this is not the case, and only one of the operating systems is to be used, the assignment or lab exercise will specify this. Students may remotely access `tuxworld` from any of the workstations in Thorv S311, S315, S320, or S360; from a computer elsewhere on campus; or from off-campus. The procedure is described at <http://www.cs.usask.ca/support/index.php#all>, under "Remote Access to LINUX".

---

## Policies

### Late Assignments and Labs

- Policies on late assignments are outlined in the [Student Evaluation](#) section of this document, and re-iterated in the [Assignments](#) section.
- Policies on late labs are outlined in the [Student Evaluation](#) section of this document, and re-iterated in the [Instructional Laboratory](#) section.

## Missed Assignments and Labs

- Policies on missed assignments are outlined in the [Student Evaluation](#) section of this document, and re-iterated in the [Assignments](#) section.
- Policies on missed labs are outlined in the [Student Evaluation](#) section of this document, and re-iterated in the [Instructional Laboratory](#) section.

## Missed Examinations

Policies of the College of Arts & Science apply to this course in the case of missed examinations. Those policies are described at <http://www.usask.ca/programs/colleges-schools/arts-science/#AcademicInformationandPolicies>. Expand the item entitled "Deferred and Supplemental Examinations".

## Incomplete Course Work and Final Grades

The procedures and policies regarding incomplete course work and final grades described in item 8.4 of the University's ["Academic Courses Policy on Course Delivery, Examinations & Assessment of Student Learning"](#) document apply to this class. Section II of the same document describes provisions governing examinations and grading.

## Academic Honesty

The University of Saskatchewan is committed to the highest standards of academic integrity and honesty. Students are expected to be familiar with these standards regarding academic honesty and to uphold the policies of the University in this respect. All students should read and be familiar with the Regulations on Academic Student Misconduct (<http://www.usask.ca/secretariat/student-conduct-appeals/academic-misconduct.php>) as well as the Standard of Student Conduct in Standard of Student Conduct in Non-Academic Matters and Procedures for Resolution of Complaints and Appeals (<http://www.usask.ca/secretariat/student-conduct-appeals/non-academic-misconduct.php>). Academic honesty is also defined and described in the Department of Computer Science Statement on Academic Honesty (<http://www.cs.usask.ca/students/academic-honesty/index.php>). Students should avoid any behaviour that could potentially result in suspicions of cheating, plagiarism, misrepresentation of facts and/or participation in an offence. Academic dishonesty is a serious offence and can result in suspension or expulsion from the University. Policies on student discipline can be found at <http://policies.usask.ca/policies/student-affairs-and-activities/student-discipline.php>.

For more information on what academic integrity means for students see the Student Conduct & Appeals section of the University Secretary Website at: <http://www.usask.ca/secretariat/student-conduct-appeals/index.php>.

In Cmpt 214 all assignments and laboratory exercises are to be completed on an individual basis, unless specified otherwise. For the purposes of this class, the interpretation of what constitutes "an individual basis" is:

- You may work on certain aspects of an assignment or laboratory as a member of a group. However, other aspects of your work must be done independently. Which aspects must be done independently are described below. In any event, it is good practice to clearly identify the other members of the group in your submission, and acknowledge the contributions of those group members (who contributed what).
- You may work within a group to verify that the members of the group have understood the problem. You may even confirm your design and results together. However, each of you must perform the

implementation phase independently. Failure to perform an independent implementation constitutes academic dishonesty.

- You may work within a group on testing strategies and criteria for the selection of test cases. However, each of you must perform the testing phase independently. Failure to perform an independent test constitutes academic dishonesty.
- Each of you must compose your documentation (internal documentation, external documentation, and testing documentation) independently. Failure to compose your own documentation constitutes academic dishonesty.
- You are not allowed to post queries and then get solutions in discussion forums like stackoverflow, slashdot, superuser, etc. Such actions will constitute academic misconduct and will be dealt with as such.

Please understand that students without the practical experience gained when performing their own implementation, testing, and documentation will be at a significant disadvantage in their examinations when compared to those students who did. It will also be a serious detriment to those students in subsequent courses, and in an employment situation.

## **Examinations with Disability Services for Students (DSS)**

Students who have disabilities (learning, medical, physical, or mental health) are encouraged to register with Disability Services for Students (DSS), if they have not already done so. Students who suspect they may have disabilities should contact DSS for advice and referrals. In order to access DSS programs and supports, students must follow DSS policy and procedures. For more information, check

<https://students.usask.ca/health/centres/disability-services-for-students.php>, or contact DSS at 306-966-7273 or [dss@usask.ca](mailto:dss@usask.ca).

Students registered with DSS may request alternative arrangements for midterm and final examinations. Students must arrange such accommodations through DSS by the stated deadlines. The instructor will provide examinations for students who are being accommodated by the deadlines established by DSS.

## **Class Recordings**

Students wishing to record any portion of lectures, either audio or video plus audio, must contact the instructor beforehand and obtain permission to do so. Note that Section 5 of the [University's "Academic Courses Policy on Class Delivery, Examinations, and Assessment of Student Learning" document](#) lays out policies and requirements associated with such recordings.

## **Moodle Discussion Forums**

Students are encouraged to make use of the moodle discussion forums set up for the class. The professor may on occasion move or remove a posting where he deems such action to be in the best interests of the class.

## **Computer, Software, and Network Use**

University policies apply to use of [University-owned computers](#), [electronic mail on University servers and/or a University email address](#), [institutional data](#), and [materials protected by copyright](#) by students in this class. There are also applicable policies on [security of University \(computer\) networks](#). Should violation of any policies occur by students, disciplinary action will follow the [policies and procedures on student discipline](#).

## Other Policies

All students must be properly registered in order to attend lectures and receive credit for this course.

---

## Student Supports

Student Learning Services (SLS) offers assistance to U of S undergrad and graduate students. For information on specific services, please see the SLS web site, <http://library.usask.ca/studentlearning/>.

The office of the Vice-Provost Teaching and Learning focuses on providing developmental and support services and programs to students and the university community. For more information, see the Teaching and Learning web site at <http://teaching.usask.ca>.

---

*kusalik@cs.usask.ca*