



UNIVERSITY OF CALGARY
FACULTY OF SCIENCE
DEPARTMENT OF COMPUTER SCIENCE
COURSE OUTLINE

1. **Course:** CPSC 231: Introduction to Computer Science for Computer Science Majors I

Lecture Sections:

L01, TR 9:30-10:45, SA 104, Pavol Federl., ICT 742, 220-5103, pfederl@ucalgary.ca

L02, TR 15:30-16:45, SA 104, James Tam, ICT 707, 210-9455, tamj@cpsc.ucalgary.ca

L03, TR 14:00-15:15, SA 104, Pavol Federl, ICT 742, 220-5103, pfederl@ucalgary.ca

L01 and L03 Office Hours: TR 12:00-13:00

L02 Office Hours: T 14:00-14:50 R 16:45-17:15

Course Website: L01/L03: <http://pages.cs.ucalgary.ca/~pfederl/CPSC231>

L02: <http://pages.cpsc.ucalgary.ca/~tamj/2016/231F/>

Computer Science Department Office, ICT 602, 220-6015, cpsc@cpsc.ucalgary.ca

2. **Prerequisites:** None.

(<http://www.ucalgary.ca/pubs/calendar/current/computer-science.html#3620>)

3. **Grading:** The University policy on grading and related matters is described in sections F.1 and F.2 of the online University Calendar. In determining the overall grade in the course the following weights will be used:

Assignments	35%
Midterm Exam	25%
(Thursday October 27 th , 2016 at 18:00 in ST 140 and ST 148)	
Final Exam	40%

This course **will** have a Registrar's Scheduled Final Exam.

Special Regulations affecting Final grade: Each course component will be computed using a grade point. However incremental grades may be awarded, on individual components, which go up in increments of 0.1 starting at a grade point of 1.0 e.g., 1.1, 1.1, 1.2, 1.3...4.0. The term grade will be calculated by summing the following: each component grade multiplied by the weight for each component and then reconverted to a final letter grade using the Official University Grade Point equivalents.

If a student is awarded a higher grade point (not a raw score and not a percentage) on the final exam than the midterm, then the final exam grade point will be used on both the midterm and final exam components when calculating the term grade point. (An analogous swap will NOT occur if the midterm score is higher than the final exam).

4. **Missed Components of Term Work:** The regulations of the Faculty of Science pertaining to this matter are found in the Faculty of Science area of the Calendar. Section 3.6. It is the student's responsibility to familiarize themselves with these regulations. See also Section E.6 of the University calendar.
5. **Scheduled Out-of-Class Activities:** REGULARLY SCHEDULED CLASSES HAVE PRECEDENCE OVER ANY OUT-OF-CLASS-TIME ACTIVITY. If you have a clash with this out-of-class activity, please inform your instructor as soon as possible so that alternative arrangements can be made.
6. **Course Materials:**
Starting Out with Python, Tony Gaddis, *Pearson* (Recommended).

Online Course Components:

Lecture slides will be made available on the course webpage. These lecture slides are mandatory for the course.

7. **Examination Policy:** Closed book. No aids of any kind are permitted. Bring at least one pen and one pencil to the exam. Students should also read the Calendar, Section G, on examinations.
8. **Approved Mandatory and Optional Course Supplemental Fees:** None.
9. **Writing across the Curriculum Statement:** In this course, the quality of the student's writing in the weighted components of the course will be a factor in the evaluation of these components. See also Section E.2 of the University Calendar.
10. **Human Studies Statement:** Students will be expected to participate as subjects or participants in projects. See also Section E.5 of the University Calendar.
11. **OTHER IMPORTANT INFORMATION FOR STUDENTS:**
 - a) **Misconduct:** Academic misconduct (cheating, plagiarism, or any other form) is a very serious offense that will be dealt with rigorously in all cases. A single offence may lead to disciplinary probation or suspension or expulsion. The Faculty of Science follows a zero tolerance policy regarding dishonesty. Please read the sections of the University Calendar under Section K, Student Misconduct to inform yourself of definitions, processes and penalties.
 - b) **Assembly Points:** In case of emergency during class time, be sure to FAMILIARIZE YOURSELF with the information on assembly points which can be found in each classroom and building.
 - c) **Student Accommodations:** Students needing an Accommodation because of a Disability or medical condition should contact Student Accessibility Services in accordance with the Procedure for Accommodations for Students with Disabilities available at http://www.ucalgary.ca/policies/files/policies/procedure-for-accommodations-for-students-with-disabilities_0.pdf. Students needing an Accommodation in relation to their coursework or to fulfil requirements for a graduate degree, based on a Protected Ground other than Disability, should communicate this need, preferably in writing, to the Associate Head of Computer Science.
 - d) **Safewalk:** Campus Security will escort individuals day or night (<http://www.ucalgary.ca/security/safewalk/>). Call 403-220-5333 for assistance. Use any campus phone, emergency phone or the yellow phones located at most parking lot pay booths.
 - e) **Freedom of Information and Privacy:** This course is conducted in accordance with the Freedom of Information and Protection of Privacy Act (FOIPP). As one consequence, students should identify themselves on all written work by placing their name on the front page and their ID number on each subsequent page. For more information see also <http://www.ucalgary.ca/secretariat/privacy>
 - f) **Student Union Information:** VP Academic (403) 220-3911 suvpaca@ucalgary.ca SU Faculty Rep (403) 220-3913 science1@su.ucalgary.ca, science2@su.ucalgary.ca and science3@su.ucalgary.ca, Student Ombuds Office: (403) 220-6420 ombuds@ucalgary.ca, <http://ucalgary.ca/provost/students/ombuds>
 - g) **Internet and Electronic Device Information:** You can assume that in all classes that you attend your cell phone should be turned off unless instructed otherwise. All communications with other individuals via laptop computers, cell phones or other devices connectable to the internet in not allowed during class time unless specifically permitted by the instructor. If you violate this policy you may be asked to leave the classroom. Repeated abuse may result in a charge of misconduct.
 - h) **U.S.R.I.:** At the University of Calgary feedback provided by students through the Universal Student ratings of Instruction (USRI) survey provides valuable information to help with evaluating instruction, enhancing learning and teaching, and selecting courses (www.ucalgary.ca/usri). Your responses make a difference – please participate in USRI surveys.

Department Approval _____ Date _____

Associate Dean's Approval for out of regular class-time activity: _____ Date: _____

Associate Dean's Approval for Alternate final examination arrangements: _____ Date: _____

A signed copy of this document is kept on file in the Computer Science main Office ICT 602

CPSC 231 Syllabus

Tentative Topics Covered:

- Writing/translating/running Python programs
- Variables and constants
- Program documentation
- Keyboard input, console output
- Branching and decision making
- Loops and repetition
- Functions and program decomposition
- Text file input and output
- Composites: strings, lists and tuples
- Classes and objects
- Recursive functions
- Graphic programs

Learning Outcomes (from the above topics)

- write/trace a Python program that use core programming concepts such as: variables, keyboard input/console output, branching, loops, functions, simple recursive functions, assignment/Boolean/mathematical operators, text file input/output
- write/trace a Python program that defines a class with variable attributes and methods (including but not limited to constructors), creates an instance and invokes the methods inside and outside the class
- write/trace a Python program that invokes methods of structures such as lists and strings
- design and implement a small application with a graphical user interface implemented using procedural Python code.
- document a Python a program and understand it's purpose and know common documentation requirements
- develop debugging skills to systematically identify and fix syntax and logic errors in procedural code written by self and others.