



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

1. **Course:** CPSC 355: Computing Machinery

Lecture Sections:

L01, MWF 9:00-9:50, Leonard Manzara, ICT 703, 220-3518, lcmanzar@ucalgary.ca

Office Hours: MWF 10:00-11:00

Course Website: D2L

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

2. **Prerequisites:** One of CPSC 219, 233, 235

(<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	30%
Midterm Exam	30%
<i>(In-Class Wednesday March 8th, 2017)</i>	
Final Exam	40%

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

Special Regulations affecting Final grade: Each of the above components will be given a percentage grade. The final grade will be calculated using the weights above and then converted to a final letter grade using the attached table. To achieve an overall grade of C- or better in the course, you must achieve a minimum grade of C- in the final exam and complete all assignments.

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:**

Computer Organization and Design, Patterson and Hennessey, Morgan Kaufmann

The C Programming Language, Brain Kernighan and Dennis Ritchie, Prentice Hall

Online Course Components:

None.

7. **Examination Policy:** Closed book. 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 _____

Faculty Approval for
out of regular class-time activity: _____
Date: _____

Faculty Approval for
Alternate final examination arrangements: _____
Date: _____

A signed copy of this document is on file in the Computer Science Main Office

CPSC 355 Percentage to Letter Grade Conversion Table

A+	95-100
A	90-95
A-	85-90
B+	80-85
B	75-80
B-	70-75
C+	65-70
C	60-65
C-	55-60
D+	50-55
D	45-50
F	0-45

CPSC 355 Syllabus

Introduction
Course Objectives

Computer Architectures and Assembly Language Programming

Basic Computer Architecture

High-Level Architecture

CPU

System Clock

Primary Memory

Bus

Secondary Memory

Peripheral I/O Devices

Basic CPU Architectures

RISC and CISC Architectures

Instruction Cycle

Assembly Language Programming

Assemblers

Macro Preprocessors

ARMv8-A Architecture

Introduction

ARMv8 Registers

A64 Assembly Language

Basic Program Structure

Basic Arithmetic Instructions

Printing to Standard Output

Branch Instructions and Condition Codes

Loops

The if Construct

The if-else Construct

Introduction to the gdb Debugger

Binary Numbers and Integer Representations

Binary Numbers

Unsigned Integers

Signed Integers

Hexadecimal Numbers

Octal Numbers

Integer Classes and Subtypes

Bitwise Operations

Bitwise Logical Instructions

Bitwise Shift Instructions

Bitfield Operations

Sign/Zero Extend Operations

Binary Arithmetic

Modulus Arithmetic

Addition

Subtraction

Signed Number Branching Conditions

Unsigned Arithmetic

Unsigned Number Branching Conditions

Multiplication

Division

Extended Precision Arithmetic

- The Stack
- Memory and Memory Addressing
- Stack Memory
- The Stack and Frame Pointers
- Frames and Frame Records
- Creating and Addressing Stack Variables
- Memory Alignment
- Basic Load and Store Instructions
- Load/Store Addressing Modes
- Stack Variable Offset Macros

- Data Structures
- One-Dimensional Arrays
- Multidimensional Arrays
- Structures
- Nested Structures

- Subroutines
- Introduction
- Open (Inline) Subroutines
- Closed Subroutines
- Subroutine Linkage
- Saving and Restoring Registers
- Arguments to Subroutines
- Pointer Arguments
- Returning Integers
- Returning Structures
- Subroutines with 9 or More Arguments
- Optimizing Leaf Subroutines

- External Data and Text
- Introduction
- External Variables
- The text, data, and .bss Sections
- The ASCII Character Set
- Creating and Addressing String Literals
- External Arrays of Pointers
- Command-Line Arguments

- Separate Compilation
- Relocatable Object Code
- Separate Compilation and Linking

- Input and Output
- Introduction
- System I/O
- Standard I/O
- File I/O

- Floating-Point Numbers
- Introduction
- Fixed-Point Numbers
- Floating-Point Single Format
- Floating-Point Double Format
- Floating-Point NaNs
- Floating-Point Registers
- Basic Floating-Point Instructions

Floating-Point Arguments
Floating-Point Return Values

Machine Instructions (as time permits)

Introduction

R-Type Format

D-Type Format

I-Type Format

Learning Outcomes:

By the end of the course, students will:

1. By the end of this course, students should be able to describe the architecture of a basic computer.
2. By the end of this course, students should be able to explain the architecture of CPUs in common current use.
3. By the end of this course, students should be able to design and implement assembly language programs of moderate complexity.
4. By the end of this course, students should be able to create a program that consists of separately compiled source code modules that mix C code and assembly language code.
5. By the end of this course, students should be able to explain the underlying machine representation of common data types such as signed and unsigned integers, characters, strings, and floating point numbers.
6. By the end of this course, students should be able to take high-level language constructs and translate them into an equivalent assembly language implementation.

Allowable Sources:

No Restrictions on source material.

Cited Sources:

If you used an article, book, function or algorithm that you did not create for this course you must cite it. (This means you may have to cite yourself!) Use APA for citations in a report, paper or in the header documentation of computer code you submit. If citing a website, make sure you include the date you accessed the website. Don't forget to cite code that you used, even if you modified the code.

Level of Collaboration between Students:

You may discuss the assignments with other students in the class but do NOT share any code, do not ask others to provide you with code and do not show code that you have created for assignments to other students.

Disclosure Policy

If you discuss the assignments with others, make sure to cite these discussions.