

OFFICIAL UNDERGRADUATE COURSE OUTLINE FORM

Note: The University reserves the right to amend course outlines as needed without notice.

Course Code and Number: COMP 256		Number of Credits: 3 Course credit policy (105)																	
Course Full Title: Introduction to Machine Architecture																			
Course Short Title (if title exceeds 30 characters): Intro to Machine Architecture																			
Faculty: Faculty of Professional Studies		Department (or program if no department): Computer Information Systems																	
Calendar Description: Students study the fundamental digital circuits in typical microcomputer hardware, and assembly language programming as a tool for understanding the computer architecture and for controlling computer hardware devices.																			
Prerequisites (or NONE):		COMP 125 and one of (COMP 150 or COMP 152).																	
Corequisites (if applicable, or NONE):		None																	
Pre/corequisites (if applicable, or NONE):																			
Equivalent Courses (cannot be taken for additional credit) Former course code/number: Cross-listed with: Equivalent course(s): <i>Note: Equivalent course(s) should be included in the calendar description by way of a note that students with credit for the equivalent course(s) cannot take this course for further credit.</i>		Transfer Credit Transfer credit already exists: <input type="checkbox"/> Yes <input checked="" type="checkbox"/> No Transfer credit requested (OREg to submit to BCCAT): <input type="checkbox"/> Yes <input checked="" type="checkbox"/> No (if yes, fill in transfer credit form) Resubmit revised outline for articulation: <input checked="" type="checkbox"/> Yes <input type="checkbox"/> No To find out how this course transfers, see bctransferguide.ca .																	
Total Hours: 60 Typical structure of instructional hours: <table border="1"> <tr> <td>Lecture hours</td> <td>45</td> </tr> <tr> <td>Seminars/tutorials/workshops</td> <td></td> </tr> <tr> <td>Laboratory hours</td> <td>15</td> </tr> <tr> <td>Field experience hours</td> <td></td> </tr> <tr> <td>Experiential (practicum, internship, etc.)</td> <td></td> </tr> <tr> <td>Online learning activities</td> <td></td> </tr> <tr> <td>Other contact hours:</td> <td></td> </tr> <tr> <td>Total</td> <td>60</td> </tr> </table>		Lecture hours	45	Seminars/tutorials/workshops		Laboratory hours	15	Field experience hours		Experiential (practicum, internship, etc.)		Online learning activities		Other contact hours:		Total	60	Special Topics Will the course be offered with different topics? <input type="checkbox"/> Yes <input checked="" type="checkbox"/> No If yes, different lettered courses may be taken for credit: <input type="checkbox"/> No <input type="checkbox"/> Yes, repeat(s) <input type="checkbox"/> Yes, no limit <i>Note: The specific topic will be recorded when offered.</i>	
Lecture hours	45																		
Seminars/tutorials/workshops																			
Laboratory hours	15																		
Field experience hours																			
Experiential (practicum, internship, etc.)																			
Online learning activities																			
Other contact hours:																			
Total	60																		
		Maximum enrolment (for information only): 35 Expected frequency of course offerings (every semester, annually, every other year, etc.): Twice a year																	
Department / Program Head or Director: Dan Harris		Date approved: September 2017																	
Faculty Council approval		Date approved: October 13, 2017																	
Campus-Wide Consultation (CWC)		Date of posting: January 19, 2018																	
Dean/Associate VP: Tracy Ryder Glass		Date approved: October 13, 2017																	
Undergraduate Education Committee (UEC) approval		Date of meeting: January 26, 2018																	

Learning Outcomes

Upon successful completion of this course, students will be able to:

- describe the representation of information in the computer
- describe memory addressing modes
- describe high level language parameter-passing conventions
- demonstrate interfacing with the operating system
- demonstrate memory interfacing
- demonstrate I/O interfacing and I/O programming
- describe basic hardware decoding, memory
- use I/O devices including the timer, serial port and programmable interrupt controller as used in the IBM PC
- write programs which control computer hardware
- write assembly language programming

Prior Learning Assessment and Recognition (PLAR)

☒ Yes ☐ No, PLAR cannot be awarded for this course because

Typical Instructional Methods (guest lecturers, presentations, online instruction, field trips, etc.; may vary at department's discretion)

Lectures and labs with hands-on lab exercises involving digital design software, assembly language and high level language programming to control the PC.

Grading system: Letter Grades: ☒ Credit/No Credit: ☐ Labs to be scheduled independent of lecture hours: Yes ☐ No ☐

NOTE: The following sections may vary by instructor. Please see course syllabus available from the instructor.

Typical Text(s) and Resource Materials (if more space is required, download Supplemental Texts and Resource Materials form)

Author (surname, initials)	Title (article, book, journal, etc.)	Current ed.	Publisher	Year
1. Kip Irvine	Assembly Language for x86 Processors	<input checked="" type="checkbox"/>	Pearson	2015
2. Rob Williams	Computer Systems Architecture: A Networking Approach	<input type="checkbox"/>	Pearson	
3. Muhammad Ali Mazidi & Janice Gillispie Mazidi	The 80x86 IBM PC and Compatible Computers (Volumes 1 & II)), Assembly Language, Design, and Interfacing	<input type="checkbox"/>	Prentice Hall	2002
4.		<input type="checkbox"/>		
5.		<input type="checkbox"/>		

Required Additional Supplies and Materials (software, hardware, tools, specialized clothing, etc.)**Typical Evaluation Methods and Weighting**

Final exam:	40%	Assignments:	%	Midterm exam:	35%	Practicum:	%
Quizzes/assignments:	25%	Lab work:	%	Field experience:	%	Shop work:	%
Other:	%	Other:	%	Other:	%	Total:	100%

Details (if necessary):

Typical Course Content and Topics

- Number systems review
- Bit manipulation and bit masking
- Computer architecture review
- Intel architecture
- Assembly language
- BIOS interrupts
- Software interrupts
- Passing parameters
- Interface cards (ISA)
- 8254 timer
- Serial communication, 8250 UART