NORTHWESTERN POLYTECHNIC

SCIENCE DEPARMENT

COURSE OUTLINE – Fall 2024

CS2290 – COMPUTER ORGANIZATION AND ARCHITECTURE I

3 (3-0-3) 90 Hours for 15 Weeks

Northwestern Polytechnic acknowledges that our campuses are located on Treaty 8 territory, the ancestral and present-day home to many diverse First Nations, Metis, and Inuit people. We are grateful to work, live and learn on the traditional territory of Duncan's First Nation, Horse Lake First Nation and Sturgeon Lake Cree Nation, who are the original caretakers of this land.

We acknowledge the history of this land and we are thankful for the opportunity to walk together in friendship, where we will encourage and promote positive change for present and future generations.

INSTRUCTOR	Dr. Mohammad Mirbagheri	**PHONE**	780 539 - 2739
OFFICE	C419	**E-MAIL**	MMirbagheri@NWPolytech.ca
OFFICE HOURS	appointment by email		

CALENDAR DESCRIPTION
General introduction to number representation, architecture and organization concepts of von Neumann machines, assemble level programming, exception handling, peripheral programming, floating point computations and memory management.

PREREQUISITE(S)/COREQUISITE CS1150

DELIVERY Mode Onsite face-to-face

REQUIRED TEXT/RESOURCE MATERIALS
Assembly Language for x86 Processors, 8th Edition (older versions are acceptable) By Kip R. Irvine, Pearson Publishing

LEARNING OUTCOMES
- Understand computer data representation
- Know basic processor architecture and memory management
- Be able to write, assemble, and debug Intel Assembler code
- Be able to perform conditional processing and Integer arithmetic, use code libraries, code procedures and advanced procedures and use string manipulation routines
- List the basic components of a modern CPU

TRANSFERABILITY

Please consult the Alberta Transfer Guide for more information. You may check to ensure the transferability of this course at the Alberta Transfer Guide main page http://www.transferalberta.alberta.ca.

** Grade of D or D+ may not be acceptable for transfer to other post-secondary institutions. **Students are cautioned that it is their responsibility to contact the receiving institutions to ensure transferability**.

EVALUATION

Lab/HomeworkAssignments:	30%
Quizzes:	10%
Midterm Exam:	20%
Final Exam:	40%
Instructor Discretion	+/- 5%
Total	100%

Note: The student must pass the theory/concepts portion of the course in order to qualify for a passing grade for the term. That means, a student must obtain 35 out of a possible 70 points (from exams/quizzes) before adding the lab assignment marks to compute the final grade. If you cannot achieve the required 50% (on exams) then regardless of your lab assignment grades, you cannot pass the course.

Grading Chart

Alpha Grade	4-point Equivalent	Percentage Guidelines	Alpha Grade	4-point Equivalent	Percentage Guidelines
A+	4.0	95-100	C+	2.3	67-69
A	4.0	85-94	C	2.0	63-66
A-	3.7	80-84	C-	1.7	60-62
B+	3.3	77-79	D+	1.3	55-59
B	3.0	73-76	D	1.0	50-54
B-	2.7	70-72	F	0.0	00-49

Attendance & Participation

Lectures are **interactive** discussions, meaning that **attendance** is expected. Regular and **punctual** attendance in lectures is crucial to academic success in the course and is expected of all students. Students should make every effort not to miss class. There is always a possibility of missing important information if you skip class. Furthermore, students who skip class usually do not do as well as those who do not.

You are responsible for material covered if you are late or absent. Note that attendance and participation may form part of your final grade to be adjusted by +/- 5% at the instructor's discretion.

Lab Sessions
Lab sessions will occur weekly**.** Attendance is **mandatory**.

COURSE SCHEDULE/TENTATIVE TIMELINE

- Introduction to Computer Architecture
    - Microprocessor and computer architecture
    - Operations and operands of computer hardware
    - Representing instructions

- Number systems and Arithmetic
    - Signed and Unsigned Numbers
    - Addition and Subtraction
    - Logical Operations
    - Multiplication and Division
    - Floating Point numbers

- 80x86 Assembly
    - Overview of 80x86 assembler (segments, registers and organization)
    - Program structure
    - I/O operations
    - Data movement instructions
    - Conditionals and Branching instructions
    - Arrays
    - Procedures
    - Interrupts
    - String processing
    - Video operations (text and graphics)
    - Parameter passing and stack operations

Academic Integrity
Academic integrity requires students be honest. Assignments and exams are to help students learn; grades show how fully this goal is attained. Thus, all work and grades should result from a student's own understanding and effort.

Acts of academic misconduct violate academic integrity, and are considered serious offences by the College. Examples include, but are not limited to, cheating on tests or exams, plagiarizing, copying from others, submitting the work of others as your own, etc. Instances of academic misconduct will be reported. More details are provided at
https://www.nwpolytech.ca/about/administration/policies/index.html

STATEMENT ON ACADEMIC MISCONDUCT

Cheating will not be tolerated and there will be penalties**.** Co-operation on programming assignments is generally encouraged, but it must be limited to verbal discussion of concepts; not program code or any other written documentation that is submitted for grading. Copying of assignments or previous solution keys, and submitting this as your own work is plagiarism, a serious form of academic misconduct. Knowingly **allowing an assignment to be copied** will also be treated as plagiarism. See the previous section on Academic Integrity for more information on what to do and what to avoid.
For a more precise definition of plagiarism and its consequences, refer to the Student Conduct section of the College Calendar at https://www.nwpolytech.ca/programs/calendar/ or the College Policy on Student Misconduct: Plagiarism and Cheating at
https://www.nwpolytech.ca/about/administration/policies/index.html

**Note: all Academic and Administrative policies are available on the same page.

Generative AI
The use of generative artificial intelligence tools or applications in this course, including tools like **ChatGPT** and other AI writing or coding assistants, is **strictly prohibited**. Their usage is restricted solely to serve as learning aids and is not permitted for the completion or support of examinations, term tests, assignments, projects, or any other form of academic assessment. Any such usage is considered an academic offense in this course.

Policies and Procedures
Below is a list of general policies and procedures for this course:

1. If you have any concerns regarding a class mark, then take the following two steps: (i) clearly explain your concerns in an email; (ii) send the email to the course instructor. You must submit any concerns about an assigned mark **within one week** after it is posted.
2. When necessary, lab time may be utilized for lecturing on specific Assembly language features. The remainder of the lab time will generally be used as "hands-on" programming time.
1. There are no make-up quizzes or exams. Students who miss any of the quizzes or exams must provide a valid reason and include **supporting documentation**. In such cases, marks for the missed quiz or examination will be reallocated to the final examination.
2. Exams are "closed-book," with no additional material permitted.
3. Deferred final examinations can only be granted by the Chairperson. Deferred final examinations cannot be granted by the course instructor.

Assignments
Over the course of the semester, students will be required to complete assignments. Below is a list of policies and procedures regarding the assignments.

1. Assignments will be posted on Myclass.

2. Assignments are to be electronically submitted via MyClass. A hard copy of the report should also be submitted when it is due at the next available lecture or lab session. Keep a copy of all submitted work.
3. For each programming question, you will submit **source code** of your program files. To prove that your code works correctly, you will also submit screenshot of the output of your program in a file with an extension of **.png or .pdf**.
4. Assignments are to be handed in by the due date; otherwise, a **20% penalty per day** will be incurred; **after 2 days no credit** will be given for a late assignment. Homework that is habitually late will not be accepted.
5. If exceptional circumstances have occurred that have kept you from submitting your assignment on time, you should contact the course instructor as soon as possible. In some cases, an extension **may** be granted on the assignment. In other cases, the grades for the assignment **may** be reallocated.
6. Any incomplete work that is submitted will be considered for partial scores.