CISC-221 Computer Architecture

Course Syllabus

Revised Dec. 20,2017

Table of Contents

Quick Reference

Course title: Computer Architecture

Course number: CISC_221

Course dates: Monday, January 9, 2017 through Friday, April 7, 2017

Lecture Location: Stirling Hall Auditorium (Stirling D)

Meeting day(s): Tuesday 6:30 p.m. – 8:00 p.m., Thursday 6:30 p.m. – 8 p.m.

Labs: One of the following 2 hour slots/week: Monday @ 8:30-10:30, 11:30 – 1:30, 2:30 – 4:30,

Tuesday 11:30 – 1:30

Prerequisite(s): CISC-124 or equivalent, CISC-220 is recommended

Instructor Information

- **Name:** Dave Dove
- **Email:** dove@cs.queensu.ca
- **Office location:** Goodwin Hall Room 753
- **Office hours:** Tuesday 4:00 – 5:30, Thursday 4:00 – 5:30
- **Phone:** 533-6053

Teaching assistants: to be announced

Textbook: Computer Systems, A Programmer's Perspective (Third Edition)
Bryant, O'Hallaron
Prentice Hall ISBN 978-0-13-409266-X

The second edition is also acceptable: ISBN 978-0-13-610804-7

Note: This is a required text. It may be purchased or rented from the Campus Bookstore or purchased online. Readings, Labs, Quizzes, and in-class activities are based largely on material presented in this text. There is also extensive online material.

Student Assessment:

Individual Assessment: 70%
Team Assessment: 20%
Peer Assessment: 10%

Individual Assessment

- Weekly 2 hour labs: 10%
- Quiz 1: 15%
- Quiz 2: 15%
- Quiz 3: 15%
- Quiz4: 15%

Dates are tentative. Consult the course web site for current information. Quizzes are 60 minutes in length and take place in the normally scheduled lecture period. Quizzes are closed book. There is no midterm or final exam.

Team Assessment (by Instructor)

- Team readiness (RAT) tests 10%
- Team in-class activities 10%

Peer Assessment (by Students)

Mid-term Peer Assessment 5%
End-of-term Peer Assessment 5%

Course Topics

1. Data Representation
    - Binary representation of numeric and non-numeric data
    - Bit level manipulation and computer arithmetic
2. Machine-Level Representation of Programs
    - Basic instruction set requirements, operand specification.
    - Machine architecture – ARM, IA-32, others
    - Assembly Language Programming:
        - Assembly language program structure, linkers, loaders
        - Mapping of high level languages to machine language instructions.
3. Digital Logic
    - Combinational networks, truth tables, Boolean algebra, sequential networks
4. Enhancing performance
    - pipelining, caches
    - instruction-level parallelism
    - superscalar processors, multiprocessors and clusters
5. The Memory Hierarchy
    - Virtual Memory
6. Exceptions, Devices, Interrupts
    - I/O, role of operating systems, interrupts and traps
    - Interrupt service routines, event driven programming.

Why Study Computer Architecture in Computing Science?

1. An understanding of computer architecture will help you write computer programs that are faster, smaller, and less prone to error
2. An understanding of computer architecture enables you to appreciate the relative performance cost of software operations and the effect of the programming choices that you will make.
3. An understanding of computer architecture will help you debug your programs when problems arise.
4. An understanding of computer architecture will give you a better understanding of the resources needed to run your programs effectively.
5. Computer architecture fundamentals are not directed toward any specific class of application or computer language. The fundamentals learned in this course are timeless. What you learn will assist you every day for the rest of your computing career!
6. The assembly language skills acquired in this course can be listed directly on your CV. this skill may be judged as important by your future employer, not only as a specific skill, but also as a reflection of your general understanding of how computers work.
7. If you ever program an embedded system (including mobile devices, gaming platforms and real-time data acquisition systems), you will need to understand the platform's computer architecture because resources will be limited.
8. Technology changes rapidly making past choices often obsolete. An understanding of computer architecture will help you understand the effects that technological advances will have on your programs – past, present, and future.
9. An understanding of computer architecture will help you adapt to new technologies. It will be your task to turn advances in technology into speed, usability, and even new classes of applications that aren't even thought of yet!
10. If you ever buy a computer, how will you know what the new features are, and how do you judge their importance?

Course Goals:

After the course, you will be able to:

1. Demonstrate a knowledge and understanding computer architecture and programming languages as they pertain to how data is represented, manipulated and stored in a computer system.
2. Demonstrate a knowledge and understanding of the relationships between computer architecture, programming language and programming choices and the role they each play in a program's performance.
3. Demonstrate the ability to make informed, effective programming choices that will best satisfy the performance, reliability, and functional objectives of the programs that you create.

4. Demonstrate the use of critical thinking skills to solve problems and evaluate alternatives relating to computer programs.
5. Demonstrate interpersonal and team interaction skills.
6. Say that you enjoyed the course!

Course Objectives:

To meet the goals, you will individually, and as a team demonstrate knowledge and understanding of course concepts, including but not limited to the following, by:

1. Creating and/or analyzing program segments relating to computer data representation and manipulation.
2. Creating and/or analyzing program segments focusing on the performance differences of alternative programming choices that are affected by a computer's architecture.
3. Demonstrating a knowledge and understanding of how a computer's resources are managed by hardware and the Operating System.
4. Demonstrating proficiency in using Unix (linux) editing, compilation and debugging tools to create, analyze and debug short C and assembly language programs.
5. Discussing, in a team environment, case studies and issues relating to computer architecture and making defensible programming choices to best satisfy given requirements.

You will also demonstrate interpersonal and team interaction skills in in-class activities.

Course Outcomes:

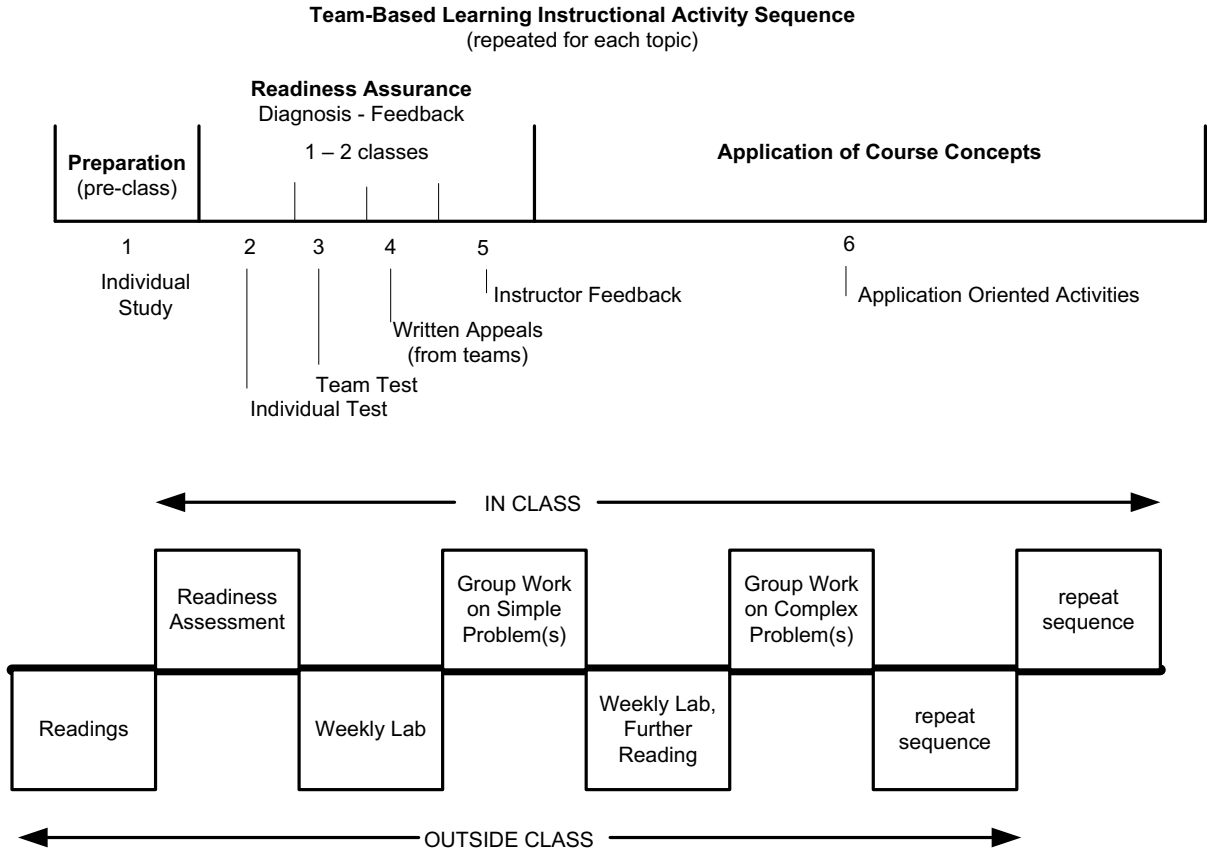
Your success will be measured in these ways:

1. Assessment of preliminary multiple choice quizzes, both individually and as a team, relating to concepts covered in reading assignments that precede major topics (Readiness Assessment Tests).
2. Assessment of in-class team- based problem solutions.
3. Assessment of individual weekly lab submissions.
4. Assessment of individual in-class quizzes consisting of multiple choice and short answer questions given approximately every three weeks (4 per term).
5. Peer assessment of your work as a team member.

Course Delivery

Starting in 2013, this course has used a team-based learning approach for course delivery. In this approach, the focus is on what the learner learns rather than on what the instructor teaches. Course content is acquired by the learner under the guidance of the instructor. The instructor will give a brief

overview of the topic. Students will be given a reading/research assignment that covers the topic and includes basic questions and sample problems with solutions. Students will then be tested on basic comprehension of what they have learned, both individually and then as a team. Team solutions will be discussed and problem areas addressed, possibly with a short lecture from the instructor. The remainder of class time for the topic will be used to present problems and have teams develop, present, and discuss solutions. In preparation for in-class problems, students are required to complete weekly two hour labs. The labs provide hands-on experience demonstrating concepts covered in the topic readings and prepares them for in-class team discussions and critical thinking activities. This approach to learning involves active participation of the students as most of the lab and class time is spent applying course concepts and receiving immediate feedback on the student's level of understanding.
During the first class of the term, permanent teams of 7 or 8 students will be formed of students with diverse backgrounds. These teams will sit together in class and take part in team based discussions and in-class activities leading to team submissions for marking.



1st class of a topic

A multiple choice, Readiness Assurance Test (RAT) will be given individually. Then, the team will get together and discuss what the answers should be. The team will submit answers to the same test with immediate feedback as to whether their answers were correct or not. Team answers are submitted at the end of the class and contribute to course assessment. If a team feels strongly about the correctness of their answer but is marked wrong, there is an appeal process available to them. The individual tests

are submitted and marked but do not normally contribute to the student's course mark. An individual's credit for the team answers may be revoked if there is a pattern of low readiness or absenteeism. Known areas of difficulty relating to the week's lecture topic are discussed and clarified.

2nd class of a topic:

Any remaining questions about the topic are discussed. Problems relating to current topic are presented. All teams work on the same problems. Group solutions are compared and discussed. Solutions are submitted for marking.

3rd and subsequent classes of a topic:

Additional problems may be presented and solved.
Any unresolved problem areas with the topic are dealt with.
A short overview of the next topic is given.

There will be two assessments of your work as a member of a team. The first will be mid-term and the second will be at the end of the term. The assessments will be done by the other members of your team. Assessments will be submitted online. Comments on your team performance, submitted by the other members of your team will be made available to you (as anonymous comments). You will also have an opportunity to assess your own contribution to the team as part of this peer assessment.

Copyright of Course Materials

The material on this website is copyrighted and is for the sole use of students registered in this course. The material on this website may be downloaded for a registered student's personal use, but must not be distributed or disseminated to anyone other than students registered in this course. Failure to abide by these conditions is a breach of copyright, and may also constitute a breach of academic integrity under the University Senate's Academic Integrity Policy Statement.

Absenteeism

In a team-based learning environment, it is vitally important for you to attend every class.

In all in-class activities, student presence is recorded in some form. If a student is absent a significant number of times without giving the team notice or team members have doubts about the validity of reasons for absenteeism, it may affect their peer evaluation of that student. The instructor should also be informed if you are going to or have missed a class and the reason for it. Students who are not present for an in-class activity will not receive the team mark for the activity unless prior notification with a valid reason is given.

Students who miss a Readiness Assessment Test will not get team credit for it unless the instructor has been previously informed of the absence and the reason for it. With valid reasons and notification a student may miss up to 2 Readiness Assessment Tests and still get team credit.(See Course Policies, below).

Students who miss a Quiz must inform the instructor before or on the day of the Quiz of the reason for the absence and may be given a single opportunity to write a makeup Quiz.

Any absence that is the result of illness will require the student to complete a Queen's Declaration of Illness Form available here:
http://www.cs.queensu.ca/students/undergraduate/links/DeclarationOfIllness.pdf

and submit it to the course instructor before any accommodation for any missed work affecting academic performance assessment is considered.

Course Policies

- Students missing two Readiness Assessment Tests or getting an individual mark of less than 5 out of 10 on two tests may be asked to meet with the instructor to discuss how their performance can be improved. Without significant extenuating circumstances, the student may not benefit by the team mark on subsequent tests unless the student gets an individual mark of at least 5 out of 10 on that test.
- Students absent for an In-Class Activity will receive a mark of 0 for that activity.
- Late submissions for any work to be marked are not accepted. This includes assignments, labs, peer evaluations, and any other work that has a documented due date. There are no exceptions.
- Note that there is no Mid-Term or Final Exam in this course.
- Marking on RATs, Quizzes, or Labs may be contested by writing a scholarly appeal and submitting it to the instructor (printed or by email) in a timely manner.
- Solutions to Quizzes and Labs are not electronically published but the instructor will always be happy to review your answers and make sure any misconceptions are resolved.
- Calculators and any other electronic devices are now allowed to be used during quizzes or Readiness Assessment tests.

Peer Assessment

The students in each team will assess the other students in their team twice during the term – at mid-term and at the end of the term. Peer assessments are done using a web form and only take a few minutes of your time. You must submit peer assessments of the other students in your team by the listed due date in order to get peer assessment marks.

Your comments on the other students of the team are electronically collated and forwarded to the students as anonymous comments.

Students will also be given the opportunity to rate and comment on their own team performance as part of this peer assessment.

Statement on Academic Integrity

Academic Integrity is constituted by the five core fundamental values of honesty, trust, fairness, respect and responsibility (see www.academicintegrity.org). These values are central to the building, nurturing and sustaining of an academic community in which all members of the community will thrive. Adherence to the values expressed through academic integrity forms a foundation for the "freedom of inquiry and exchange of ideas" essential to the intellectual life of the University (see the Senate Report on Principles and Priorities http://www.queensu.ca/secretariat/policies/senate/report-principles-and-priorities).

Students are responsible for familiarizing themselves with the regulations concerning academic integrity and for ensuring that their assignments conform to the principles of academic integrity. Information on academic integrity is available in the Arts and Science Calendar (see Academic Regulation 1 http://www.queensu.ca/artsci/academic-calendars/regulations/academic-regulations/regulation-1), on the Arts and Science website (see http://www.queensu.ca/artsci/academics/undergraduate/academic-integrity), and from the instructor of this course. Departures from academic integrity include plagiarism, use of unauthorized materials, facilitation, forgery and falsification, and are antithetical to the development of an academic community at Queen's. Given the seriousness of these matters, actions which contravene the regulation on academic integrity carry sanctions that can range from a warning or the loss of grades on an assignment to the failure of a course to a requirement to withdraw from the university.

Grading Method

All components of this course will receive numerical percentage marks. The final grade you receive for the course will be derived by converting your numerical course average to a letter grade, according to Queen's Art's and Science Official Grade Conversion Scale.

Accessibility Statement

The Queen's University Equity Office has shared the following statement on their webpage for your use in ensuring that all course elements are fully accessible (http://www.queensu.ca/equity/accessibility/policystatements/accessibility-statement)
Queen's University is committed to achieving full accessibility for persons with disabilities. Part of this commitment includes arranging academic accommodations for students with disabilities to ensure they have an equitable opportunity to participate in all of their academic activities. If you are a student with a disability and think you may need accommodations, you are strongly encouraged to contact Student Wellness Services (SWS) and register as early as possible. For more information, including important deadlines, please visit the Student Wellness website at*:*
http://www.queensu.ca/studentwellness/accessibility-services/