Programming Paradigms and Systems Online Course Syllabus

Course:	**Programming Paradigms and Systems (COMP7050),** (3)
Location:	Online
Instructor:	Amittai Aviram
Telephone:	646-483-2639
Email:	avirama@wit.edu
Availability:	By e-mail, I will respond within 24 hours, and often much sooner. For telephone calls, please text me first. I can usually answer within an hour or two, if not immediately.
Office Hours:	Saturday, noon – 1:00 PM, on GoToMeeting.

TEXT AND REQUIRED TOOLS

Text:	Robert Sebesta, Concepts of Programming Languages (11th edition)

Required Materials:

Blackboard LMS: Students are required to access Blackboard to participate in this course

BOOKSTORE INFORMATION

Location:	Flannagan Campus Center, 103 Ward Street Boston MA 02115
Telephone:	617-445-8814
Website:	http://www.bkstr.com/Home/10001-10364-1?demoKey=a

COURSE DESCRIPTION

This course looks at the four main programming paradigms: imperative, functional, logic, and object-oriented. We compare the paradigms, and students gain experience with each of them. The second half of the course looks at how program execution systems are organized beyond the original von Neumann model. These include parallel and distributed systems, real-time systems, and embedded systems.

COURSE OUTCOMES

Upon completion of this course students will be able to do the following:

1. Define and demonstrate understanding of the major programming paradigms (imperative, functional, logic, and object-oriented).
2. Demonstrate understanding of type systems, including strong versus weak typing, polymorphism, and abstract versus concrete types.
3. Be familiar with the idea of domain-specific languages and their role.
4. Understand implementation issues surrounding each of the programming paradigms.
5. Recognize variations in system organization beyond the basic von Neumann model.

METHODOLOGY

This is an online course. Teaching methods include, but are not limited to, lectures, slides, class discussions, assignments, group activities, problem solving, case studies, literature review, and research projects posted within the online Blackboard course.
Each week, students are expected to do the following:

- Review the week's checklist
- Review the lecture(s)
- Complete the reading(s)
- Participate in online discussions prompted by discussion questions
- Complete all learning activities
- Complete the assignment(s)
- Complete the Assessment(s).

GETTING STARTED IN THIS COURSE

Student success in this online class depends on student active participation. Class participation is required and involves meeting discussion deadlines and learning activity requirements.

To get started, students are required to read the syllabus. Students will be asked to provide an initial introduction as a welcome activity on the first day of class. Please let the professor know immediately if there are any difficulties with this initial exploration.

COMMUNICATION

All assignments should be sent to the instructor through Blackboard. Feedback will be provided to you in the Gradebook by Wednesday of each week. Content-oriented communication can be posted to the Virtual Café so that all students can benefit from each other's learning.
I check my email and discussion boards daily and provide feedback on the work there. In general, I respond to emails and postings within 24-48 hours. If you need help on an assignment and you have not heard from me in that time frame, please feel free to call me.

I also post any updated course information or changes, as well as responses to questions that all students need to know about, on the announcements page of the course Blackboard website. Please check this page periodically during the week. Announcements also go automatically to your Wentworth e-mail address. If you do not check your Wentworth e-mail regularly, make sure you have forwarding set up so as to forward course-related e-mail to the personal e-mail address that you use most frequently. Call the Help Desk (617-989-4500) if you need assistance.

CLASS PARTICIPATION

Class participation and contributions to class discussions are essential. You should read all assigned materials and watch all videos. Not all material will be covered in online discussions, but it will be used throughout the course in a variety of different activities. Assignments should be submitted on time, unless you have obtained prior approval from the instructor. Keeping up with the workload allows you both to learn more from this course and to contribute to the learning experience of your classmates.

Participation in the discussion board is an essential aspect of this course. Each student is expected to **post a primary response** to the discussion board question(s) **by Friday at midnight** and a **minimum of two secondary responses** to a classmate's posting **by Monday at midnight**. A best practice is to check the discussion board daily, as people post early!

Your postings should

- Be substantive and clear, following the assignment directions and responding to all questions in the assignment
- *Refer to at least one theory or idea* from class readings and videos
- *Cite* any sources referenced using a standard format (such as APA), so that any of us can go find and read or view the source
- Be reread and *edited before submission* to ensure clarity of message
- Include a subject line and a signature
- Be focused – in general about 100-200 words in length.

Overly wordy postings are too hard to follow. Too-short postings do not provide enough specific information to be meaningful. Strike a balance, making sure that you have included all of the information required by the discussion board directions.

In your responses to others' postings, it is often helpful to post a follow-up question so as to deepen the discussion.

A discussion rubric is provided below and in the course in Blackboard

The discussion boards should become a forum for lively, thought provoking conversations that provide an opportunity for every voice to be heard. Listen carefully to one another. Watch for ideas or opinions that you find interesting. Ask clarifying questions when you do not understand something or need more information. Experiment with new ideas. That is what an educational experience should encourage you to do. Let your curiosity be stoked by the variety of responses from the members of your class. Your participation will be a very significant aspect of your grade for this course.

CPCE GRADUATE DISCUSSION RUBRIC

QUALITY OF DISCUSSION POSTS					
Criteria	**Missing or Serious Problems**	**Below Expectations**	**Meets Expectations**	**Exceeds Expectations**	**Exemplary Work**

Initial Response: Relevance of participation to topic under discussion *Weight 30.00%*	**0%** Contributions are off-topic or distract class from discussion.	**60%** Contributions are sometimes off-topic or incomplete.	**80%** Contributions are relevant to discussion.	**90%** Contributions are relevant to discussion and encourage others to engage.	**100%** Contributions are relevant and promote in-depth analysis of material.
Initial Response: Originality and References *Weight 15.00%*	**0%** Response is not submitted.	**60%** Responses are not original OR do not include any references (to course material) to help support points being made.	**80%** Responses are original and include at least one reference to course material.	**90%** Responses are original and well thought out and include references from the course materials. These materials are integrated into the response to display advanced learning.	**100%** Responses are original and well thought out and include references to additional beyond what is provided in the course. These materials are integrated into the response to display mastery learning.

Responses to Others *Weight 25.00%*	**0%** No responses to others are evident in this assignment.	**60%** Responds to only one student's initial discussions posting.	**80%** Responses to two students' initial discussion postings. Responses are not well developed and/or do not include both questions and comments.	**90%** Responses to two students' initial discussion postings with well thought out messages and replies to your post are developed with questions and comments.	**100%** Well-developed responses that engage other learners through responses to more than two students' answers as well as replying to messages from others on own post.

MECHANICS

	Missing or Serious Problems	Below Expectations	Meets Expectations
Grammar and Style *Weight 15.00%*	**0%** Response is not submitted.	**60%** Response needs additional work to meet academic standards and include APA formatting.	**100%** Well-developed, professional message using APA formatting and in text citations.
Frequency of Student Posts *Weight 10%*	**0%** Student did not post or respond on any day in this week.	**60%** Student responded to the Discussion Question on only one day which was a response to another student or faculty member.	**100%** Student responded to the Discussion Question on two separate days. The initial post plus two responses to others were observed.
Timeliness of Student Posts *Weight 5%*	**0 %** No posts or responses were submitted on time.	**60%** Only initial post OR responses were on time.	**100%** All posts and responses were submitted on time.

NETIQUETTE – COMMUNICATION COURTESY CODE

In an online course, we are creating a community of learners who are collaborating to enhance their own learning and the learning of others. Therefore, it is essential that all communications in discussion boards, chats, e-mails or group forums be polite, respectful, and reasonable. Creating respectful dialogue is an essential skill for every aspect of our lives. Hopefully, your experience in this online course will help to improve those skills. Here are a few helpful guidelines from which everyone will benefit:

- Positive comments go a long way to facilitate a conversation.
- Descriptive comments rather than evaluative (I'm right, you're wrong) are more likely to engage another student.
- Don't make disagreements personal, focus on the content.
- Avoid sarcasm, be curious, try to understand a different perspective.
- When you don't understand the response of another student, ask clarifying questions, be aware of assumptions (yours and those of others), and ask questions to test them out.
- Think about and review your post before you submit.
- Check the discussion threads frequently and respond appropriately and in a timely manner.
- Focus on one subject per message, and use pertinent subject lines in your e-mails and discussion thread postings.
- Capitalizing all the letters in a word is generally viewed as SHOUTING!
- Be professional and careful with your online interactions with classmates and instructors.
- Cite appropriately the sources that you quote or to which you refer.
- When you post a long message, it is generally considered courteous to warn readers at the beginning of the post that it is lengthy.
- Do not forward someone else's messages without that person's permission. This is considered rude, so always ask first.
- Students may use humor, but they should use it carefully. The absence of face-to-face cues can cause humor to be misinterpreted as criticism or flaming (i.e. angry, antagonistic criticism).

ACADEMIC ACTIVITY AND PARTICIPATION POLICY FOR ONLINE COURSES

Given the asynchronous format of online courses, students have the opportunity to take part in the class at multiple times during each weekly module. It is recommended that students login to their online course, at minimum, 4-5 times per week to participate in discussion boards, read materials, take assessments, and submit assignments. Active participation in online courses is required and is graded by the instructor.

To be considered active students will be expected to login to their online course in the Blackboard Learning Management System (LMS) and participate in at least two (2) "Learning Activities" each week:

- At least one Learning Activity by midnight ET Monday each week.
- At least one additional Learning Activity by midnight ET Sunday each week.

- Students who are inactive for more than 15% of the class can be administratively withdrawn from their online course by the instructor.
- During the first week of class students who fail to login to the course by the end of the Drop/Add period will be administratively withdrawn from the course.

“Academic activities" includes any combination of the following:

- Posting to discussion boards within the online course.
- Turning in an assignment within the online course.
- Taking an assessment within the online course.

Should a student find they are unable to take part in a given week’s activities they must contact the instructor prior to any assignment deadlines. Makeup work is accepted at the discretion of the instructor. Students should plan accordingly and make sure to read the schedule of deadlines listed in the course syllabus.

Their course instructor may administratively withdraw students who are considered inactive for more than a full week during the term without prior approval. Students who are administratively withdrawn will receive a grade of “WA” for the course. Students have the right to appeal a course withdrawal. Appeals must be in writing and submitted along with documentation to the appropriate Department Chair or Dean for their program of study.

LIBRARY SERVICES

Online students can receive assistance with research and citations as well as secure materials from the library. For a full list of hours and services please see: http://wit.edu/library/supplementals/online_student_guide2013.pdf

BLACKBOARD HELP DESK AND TECHNICAL QUESTIONS For Blackboard support during regular business hours please call the WIT Help Desk at 617-989-4500 and say "Blackboard" when prompted. For 24/7 support students should visit http://wit.echelp.org/ for the Online Support Center and Browser Test.

For additional documentation and Frequently Asked Questions (FAQs), visit http://www.wit.edu/ld/blackboard/

ACADEMIC SUPPORT

The Learning Center assists all Wentworth students with academic challenges in the areas of math, science, technical courses specific to majors, and writing. The Learning Center is a supportive and safe learning environment for students looking to improve or maintain their academic standing. In this student-based learning environment, students can receive individual help with their studies, meet and work in study groups, or find resources to assist them in meeting their goals for academic success. It includes tutors in many subjects, writing assistance and workshops. Make appointments at www.wit.edu/tlc

ACADEMIC HONESTY STATEMENT

Students at Wentworth are expected to be honest and forthright in their academic endeavors. Academic dishonesty includes cheating, inventing false information or citations, plagiarism, tampering with computers, destroying other people's studio property, or academic misconduct" (Academic Catalog). See the catalogue for a full explanation.

STUDENT ACCOUNTABILITY STATEMENT:

All work should be submitted through the course site as a permanent record. Assignments should be done in Excel or Word, in APA format. No late work is accepted without prior approval and if approved will be accepted at a 10% reduction in value for each week late. No late work will be accepted for or in week 7.
Plagiarism and cheating is not permitted. Students who are caught cheating will receive a zero for the assignment and may receive a failing grade for the entire course. See the Student Handbook for more information about plagiarism and cheating.

DISABILITY SERVICES STATEMENT

Any student who thinks s/he may require a disability related accommodation for this course should contact Disability Services to discuss your specific needs. Disability Services coordinates reasonable accommodations for students with documented disabilities. They are located in Watson Hall 003 (the Counseling Center) and can be contacted at 617-989-4390 or counseling@wit.edu. For more information on acceptable documentation and the Disability Services process, visit the Disability Services website at www.wit.edu/disabilityservices

ASSIGNMENTS AND GRADING

Final grades will be determined solely on the basis of total points acquired during the semester. There are no extra credit opportunities.

The course grade will be determined as follows:

Discussions:
30 points for an initial "introduce yourself" posting.
30 points each week, for a minimum three substantive postings by Monday midnight
= total 240 points

Assignments:
Assignment 1: 60 points.
Assignments 2- 5: 80 points.
Assignments 6 and 7: 100 points. = total 580 points.

Assessments:
Six weekly quizzes,

5 – 10 questions each,
30 points each. = total 180 points.

Course Total: 1000 points

GRADING:

For all Graduate courses, the grading scale is as follows:

Letter Grade	GPA Weight	Points	Definition
A	4.0	960 – 1000	Distinction
A-	3.64	920 – 959	High Pass
B+	3.33	880 – 919	Pass
B	3.0	840 – 879	Low Pass
B-	2.67	800 – 839	Provisional
F	0	799 or below	Fail

The quality of your work is what drives your grade earned. Written assignments are graded on the thoroughness of your work, use of concepts from the class and materials, your ability to analyze the problem or situation, and your ability to apply the principles and lessons from the readings and lectures. Your work should reflect critical thought. Whenever possible, please try to relate the course content to real-world applications and experiences, whether formal or informal, to strengthen your transfer of knowledge from class to your work. Please proofread your work carefully for spelling and grammar. Poor writing interferes with your ability to communicate your ideas.

All written work should be submitted meeting the Standards of English I. Poorly written papers may be returned ungraded for revision. All work should be proofed, typed in double space format with one-inch margins all around and 12-point type. Students are encouraged to utilize The Learning Center for help polishing their written assignment.

Individual and group papers should be submitted through Blackboard. Each written assignment includes specific directions about the topic, issues to be developed and resources required. APA format should be used to cite source material. Check the Gradebook to check for feedback on your work and your grade. I may use tracking changes to provide detailed feedback so please make sure you open the returned paper to see my comments. The Assignment instructions and grading rubrics are provided with success criteria for each assignment on Blackboard.

CPCE Undergraduate Assignment Rubric

LEVELS OF ACHIEVEMENT					
Criteria	**Unacceptable**	**Minimal**	**Competent**	**Effective**	**Mastery**
Quality of Assignment Content *Weight 70.00%*	**0%** Submission does not display an understanding of the associated learning objectives presented in the assignment.	**60%** Submission Demonstrates a minimal understanding of the associated learning objectives presented in the assignment.	**80%** Submission demonstrates some understanding of the associated learning objectives presented in the assignment.	**90%** Submission demonstrates a general understanding of the associated learning objectives presented in the assignment.	**100%** Submission demonstrates a full understanding of the associated learning objectives presented in the assignment.

LEVELS OF ACHIEVEMENT					
Criteria	**Unacceptable**	**Minimal**	**Competent**	**Effective**	**Mastery**
Mechanics and Readability of Assignment *Weight 15.00%*	**0%** Writing is unclear and/or disorganized in all literacy standards Assignment is not understandable to the reader Frequent errors in spelling and capitalization: profound and/or inaccurate punctuation which render the submission unreadable.	**60%** Writing is unclear and/or disorganized in most literacy standards Assignment is unclear and not understandable to the reader Contains many and serious intrusive errors of punctuation, spelling, and/or capitalization which partially inhibit readability.	**80%** Writing is unclear and/or disorganized in some areas of literacy standards. Assignment is partially clear and understandable to the reader Contains several punctuation, spelling, and/or capitalization errors, but does not inhibit readability.	**90%** Writing demonstrates mastery of most criteria for high academic literacy standards Assignment is mostly clear and understandable to the reader Contains only occasional punctuation, spelling, and / or capitalization errors.	**100%** Writing demonstrates mastery of all criteria for high academic literacy standards Assignment is clearly understandable to the reader Free of punctuation, spelling, and capitalization errors.

LEVELS OF ACHIEVEMENT					
Criteria	**Unacceptabl e**	**Minimal**	**Competent**	**Effective**	**Mastery**
Structure and Organization / Format/ Presentation of Assignment *Weight 15.00%*	**0%** Submission is not appropriate for assignment. Does not follow directions as indicated by instructor. References and citations are not included if requested. Thoughts are not expressed in a logical manner.	**60%** Submission follows minimal directions and is missing many components indicated in the instructions. If requested, references are not included and/ or properly cited. Errors in format several interfere with meaning, formatting is weak.	**80%** Submission does not follow all directions and is missing some components indicated in the instructions. If requested, some references are included and are partially cited. Several errors in formatting, or format is inconsistent.	**90%** Submission follows directions and includes most components indicated in the instructions. If requested, references are included and mostly cited properly. Few formatting errors.	**100%** Submission follows directions and includes most components indicated in the instructions. If requested, references are included and cited properly. Very few formatting errors.

Weekly Schedule

Week 1 Theoretical Models of Computation

Description:

In this week, you will learn the theoretical foundations for the study of programming paradigms and systems, starting with the idea of a *paradigm* itself, and including discussions of the Turing machine, the lambda calculus, type systems, and categories for the evaluation of programming languages and systems.

Estimated Completion Time:

Lectures:	3.5 hours
Readings & Resources:	7 hours
Assignment:	7 hours
Discussion:	1 hour
Assessment:	1 hour
	19.5 hours

Learning Objectives:

After this week, you will be able to do the following:

1. Know the outlines of the history of programming languages.
2. Describe the Turing machine and explain its importance.
3. Define the lambda calculus and illustrate its use in modeling computation.
4. Recognize the equivalence of the Turing machine and lambda calculus models.
5. Recognize, as two additional models of computation, first-order logic and discrete event simulation.
6. Know why type systems are important.
7. Invoke analytic criteria for evaluating programming languages.

Learning Activities:

Lectures:

Introduction to Programming Paradigms. (Required. Learning Objectives 2 – 7. 3.5 hours.)

Reading:

Robert Sebesta, *Concepts of Programming Languages*, Chapter 1 (“Preliminaries”). (Required/Learning Objectives 1, 6. 7 hours.)

Assignment:

Assignment 1: Programming for the Turing machine; Programming in the Lambda Calculus. (Learning Objectives 2 – 4, 6, 7. 7 hours. 60 points.)

Discussion:

Please introduce yourself to your classmates and me on the “Introduce Yourself!”

discussion thread. (Required. 30 points.)

Please post your answers to these questions on their respective discussion threads:

- **Computability:** What does "computable" mean? (Learning Objectives 2 – 4)
- **Intuitiveness:** Which representation of computability is more intuitive and why? (Learning Objective 7)
- **Untyped Systems:** Neither the Turing machine nor the lambda calculus have types. What consequences does this lack of types have?

(Required. Learning Objectives 2, 3, 6, 7. 1 hour. 30 points.)

Assessment(s):

Quiz 1. (Required. Learning Objectives 1 – 7. 1 hour. 30 points.)

Optional:

Recommended readings:

- Alan M. Turing, "On Computable Numbers, WIth an Application to the *Entscheidungsproblem*." (1936)
- Alonzo Church, "A Set of Postulates for the Foundation of Logic." (1933)
- Benjamin Pierce, *Types and Programming Languages*, Chapter 5 ("The Untyped Lambda Calculus"). (On 24-hour reserve at the library.)

Week 2 The Imperative Programming Paradigm: C

Description:

In this week, you will explore the imperative programming paradigm by learning the C programming language.

Estimated Completion Time:

Lectures:	3.5 hours
Readings & Resources:	7 hours
Assignment:	7 hours
Discussion:	1 hour

Assessment:	1 hour
	19.5 hours

Learning Objectives:

After this week, you will be able to do the following:

1. Write, compile, and run simple programs in the C programming language
2. Discuss C data types and semantic rules
3. Describe key features of C as an imperative programming language
4. Recognize conceptual relationships between C and the Turing machine model
5. Evaluate C and its trade-offs according to the criteria developed in week 1.

Learning Activities:

Lectures:

The Imperative Paradigm and the C Programming Language (Required. Learning Objectives 1 – 5. 3.5 hours.)

Readings:

Brian Kernighan and Dennis Ritchie, *The C Programming Language*, through chapter 8 (“Miscellaneous Functions”), section 5 (“Storage Management”).

Assignment:

Assignment 2: Programming in C. (Required. Learning Objectives 1 – 5. 8 hours. 80 points.)

Discussion:

Please post your answers to these questions on their respective discussion threads:

- What elements of the Turing machine or the lambda calculus correspond to features of C?
- What are the data types in C and why are they what they are? What motivates C’s type system?
- For what applications or areas of computing is C especially well suited?

(Required. Learning Objectives 2 – 5. 30 points. 1 hour.)

Assessment(s):

Quiz 2. (Required. Learning Objectives 1 – 5. 30 points.)

Optional:

Online C reference pages.

Week 3 The Functional Programming Paradigm: Haskell

Description:

In this week, we will get acquainted with the functional programming language paradigm through Haskell.

Estimated Completion Time:

Lectures:	3.5 hours
Readings & Resources:	7 hours
Assignment:	7 hours
Discussion:	1 hour
Assessment:	1 hour
	19.5 hours

Learning Objectives:

After this week, you will be able to do the following:

1. Write, compile, and run a simple program in Haskell
2. Identify key features of functional programming languages in contrast to imperative programming languages such as C
3. Compare and contrast data types in Haskell and C
4. Discuss the conceptual relation between Haskell and the lambda calculus
5. Identify and evaluate trade-offs in Haskell.

Learning Activities:

Lectures:

The Functional Paradigm and Haskell (Required. Learning Objectives 1 – 5. 3.5 hours.)

Reading:

Paul Hudak, John Petersen, and Joseph Fasel, *A Gentle Introduction to Haskell*, ver. 98.

(Required. Learning Objectives 1 – 5. 7 hours.)

Assignment:

Assignment 3: Programming in Haskell.

(Required. Learning Objectives 1 – 5. 7 hours. 80 points.)

Discussion:

Please post your answers to these questions on their respective discussion threads:

- What is a *function* in Haskell, as opposed to one in C? How do these compare?
- What are the advantages and disadvantages of a strict distinction between pure and impure computation?
- How does Haskell's type system differ from that of C?
- Which algorithmic concepts are easy and intuitive to express in Haskell? Which seem more intuitively expressed in C?
- C has been among the 3 most popular programming languages for many years, according to the Tiobe Index, while Haskell tends to rank below 35. Why is this?

Assessment:

Quiz 3. (Required. Learning Objectives 1 – 5. 1 hour. 30 points.)

Optional:

Robert Sebesta, *Concepts of Programming Languages*, Chapter 15, sections 1, 2, 3, and 8.

Week 4 The Logic Programming Paradigm: Prolog

Description:

In this week, you will get acquainted with the logic programming paradigm through the Prolog programming language.

Estimated Completion Time:

Lectures:	3.5 hours
Readings & Resources:	7 hours

Assignment:	7 hours
Discussion:	1 hour
Assessment:	1 hour
	19.5 hours

Learning Objectives:

After this week, you will be able to do the following:

1. Write and run a simple program in Prolog
2. Describe what a *program* is in logic programming terms
3. Explain the connection between Prolog and first-order logic as a model of computation
4. Identify and evaluate trade-offs in Prolog
5. Identify domains in which Prolog seems well suited.

Learning Activities:

Lectures:

The Logic Paradigm and Prolog (Required. Learning Objectives 1 – 5. 3.5 hours.)

Reading:

Robert Sebesta, *Concepts of Programming Languages*, Chapter 16 ("Logic Programming Languages") (Required. Learning Objectives 1 – 5. 5 hours.)

Resource:

Derek Banas, Prolog Tutorial (YouTube video). (Required. Objectives 1 - 2. 2 hours.)

Assignment:

Assignment 4: Programming in Prolog. (Required. Learning Objectives 1 – 5. 7 hours. 80 points.)

Discussion:

Please post your answers to these questions on their respective discussion threads:

- How does thinking about an algorithm change when you go from programming in C or Haskell to programming in Prolog?
- For what kinds of problems or algorithms does Prolog seem well suited, and why?

- Are there features you find missing in Prolog? Are there features of Prolog you find missing in C or Haskell?

(Required. Learning Objectives 2 – 5. 30 points.)

Assessment:

Quiz 4. (Required. Learning Objectives 1 – 5. 7 hours. 80 points.)

Optional:

- Learn Prolog Now! Tutorial.
- John R. Fisher, Prolog :- Tutorial.
- Daniel Diaz, Gnu Prolog Manual.

Week 5 Object-Oriented Programming: C++

Description:

In this week, you will get to know basic concepts of object-oriented programming (OOP) through C++.

Estimated Completion Time:

Lectures:	3.5 hours
Readings & Resources:	7 hours
Assignment:	7 hours
Discussion:	1 hour
Assessment:	1 hour
	19.5 hours

Learning Objectives:

After this week, you will be able to do the following:

1. Write, compile, and run a simple C++ program
2. Design an object-oriented program using encapsulation, inheritance, and polymorphism
3. Look up and use C++ Standard Library features
4. Relate object-oriented programming and C++ to the concept of discrete simulation

5. Identify and evaluate trade-offs of C++ in particular and object-oriented languages in general.

Learning Activities:

Lectures:

Object-oriented programming and C++. (Required. Learning Objectives 1 – 5. 3.5 hours.)

Readings:

- Robert Sebesta, *Concepts of Programming Languages*, Chapter 11 (“Abstract Data Types and Encapsulation Constructs”) (Required. Learning Objectives 1, 2, 5. 4 hours.)
- Alex Allain, C++ Tutorial. (Required. Learning Objectives 1 – 3. 3 hours.)

Assignment:

Assignment 5: Programming in C++. (Required. Learning Objectives 1 – 5. 7 hours. 80 points.)

Discussion:

Please post your answers to these questions on their respective discussion threads:

- What benefits does object orientation offer the programmer?
- Are there any costs to readability or writability associated with information hiding or other aspects of object orientation?
- How does the type system of C++ differ from that of C? How does thinking about types change?

(Required. Learning Objectives 2 – 5. 1 hour. 30 points.)

Assessment:

Quiz 5. (Required. Learning Objectives 1 – 5. 1 hour. 30 points.)

Optional:

Other online C++ tutorials:

- At cplusplus.com
- At tutorialspoint.com

By Derek Banas, on YouTube.

Week 6 Parallel and Distributed Systems

Description:

In this week, you will get introduced to systems supporting subprograms that run concurrently, as well as with systems whose components are distributed across clusters of hardware.

Estimated Completion Time:

Lectures:	3.5 hours
Readings & Resources:	7 hours
Assignment:	7 hours
Discussion:	1 hour
Assessment:	1 hour
	19.5 hours

Learning Objectives:

After this week, you will be able to do the following:

1. Describe the von Neumann computer architecture model.
2. Describe *shared-memory concurrency* and *message passing* as two parallel .programming models.
3. Define a *race condition*, the primary challenge to shared-memory parallelism.
4. Identify and describe the basic synchronization primitives used to prevent race conditions: mutual exclusion (mutex) *locks*, *condition variables*, *semaphores*, and *monitors*.
5. Describe *marshalling* as used in message passing.
6. Identify and describe problem decomposition models: *task* and *data* parallelism; the *parallel for-loop*, *specialization* parallelism, the *work queue*, and the *pipeline*.
7. Identify the essential elements of a distributed system for message-passing parallelism.

Learning Activities:

Lectures:

Introduction to parallel programming and distributed systems (Required. Learning Objectives 1 – 7. 3.5 hours.)

Readings:

- Robert Sebesta, *Concepts of Programming Languages*, Chapter 13 (“Concurrency”), Sections 1-5 and Section 10. (Required. Learning Objectives 1 – 7. 4 hours.)
- Jeff Bezanson, Using Remote Procedure Calls (RPCs) (Required. Learning Objectives 4 – 7. 3 hours.):
    - Background and Using the RPC Compiler
    - Writing the Server Code
    - Writing the Client and Putting It All Together

Assignment:

Assignment 6: Remote Procedure Calls in C++. (Required. Learning Objectives 1 – 7. 7 hours. 100 points.)

Discussion:

Please post your answers to these questions on their respective discussion threads:

- Name three applications are easy to parallelize. Name three applications are hard to parallelize.
- Suppose you need to sort a huge number of employee records. How would you scale the algorithm up to huge numbers?
- Is it possible to distribute a database over a network of computers? How?

Assessment:

Quiz 6. (Required. Learning Objectives 1 – 7. 1 hour. 30 points.)

Optional:

- Blaise Barney, Introduction to Parallel Computing.
- Ian Foster, Designing and Building Parallel Programs.
- Bill Godfrey, A Primer on Distributed Computing.
- “Distributed Computing,” Wikipedia.

Week 7 Real-Time and Embedded Systems

Description:

In this week, you will make acquaintance with systems designed to meet severe processing time or other resource constraints, posing challenges to the four programming paradigms.

Estimated Completion Time:

Lectures:	3.5 hours
Readings & Resources:	7 hours
Assignment:	7 hours
Discussion:	1 hour
Assessment:	1 hour
	19.5 hours

Learning Objectives:

After this week, you will be able to do the following:

1. Create a simulation of a real-time system.
2. Recognize the special characteristics defining real-time systems and those that typify embedded systems.
3. Understand how parallel processing techniques help support real-time systems.
4. Understand use cases in which embedded systems are necessarily also real-time systems.
5. Relate real-time and embedded system development to the idea of simulation lying behind object-oriented programming.

Learning Activities:

Lectures:

Introduction to real-time and embedded systems (Required. Learning Objectives 1 – 5. 3.5 hours.)

Readings:

- "Embedded System," Wikipedia.
- Embedded Systems Basics Tutorial, Radio-Electronics.com .

(Required. Learning Objectives 1 – 5. 5 hours.)

Resource:
Santanu Chaudhury, Embedded Systems, Lecture 1: Introduction (IIT Delhi). (Required. Learning Objective 2 and 3. 2 hours.)

Assignment:

Assignment 7. (Required. Learning Objectives 1, 3, 4, 5. 7 hours. 100 points.)

Discussion:

Please post your answers to these questions on their respective discussion threads:

- What are hard and soft real-time systems? To what problems would each of these be the most applicable, respectively?
- What is the difference between real-time and embedded systems and how are they related?

(Required. Learning Objectives 2 – 5. 1 hour. 30 points.)