CS 360 Course Syllabus
Fall, 2017

Contact Information

Instructor:	Dr. Becka Morgan
Email:	morganb@wou.edu
Office:	ITC 302C
Class Meeting Times:	MTWR 9 – 9:50
Office Hours:	See Instructor Website – www.wou.edu/~morganb
Textbook: OPTIONAL:	Programming Language Pragmatics, Author Michael L. Scott, ISBN-13: 978-0123745149, ISBN-10: 0123745144
	Concepts of Programming Languages, Author Robert W. Sebesta, ISBN-13: 9780131395312, ISBN: 0131395319

_____

Course Description & Prerequisites

This course will provide a comparative study of programming language paradigms and their application. Paradigms studied include Imperative, Functional, Logic, and Object Oriented languages. Modern scripting languages that are categorized under multiple paradigms are included. Students will learn how to approach problems from the viewpoint of each paradigm. Similarities and differences in syntax, control structures, types, scoping rules and execution model will be highlighted. The specific languages studied are selected by the instructor and may vary from year to year. Prerequisite: CS 260 and CS271

Course Outcomes/Objectives

At the end of this course students will be able to demonstrate:

1. proficiency with languages in multiple paradigms.
2. knowledge of the grammar of a programming language and how that relates to lexical analysis and parsing.
3. The ability to quickly learn new programming languages, and how to apply them to effectively solve programming problems

Course Outline

Week	Subject	Paradigm and Language	Labs
1	- History of Programming Languages - Paradigms (Imperative, Object-oriented, Functional, Logic, Declarative non-algorithmic, as well as a modern scripting language) and example languages of each - Interpretation vs. compilation to native code vs. compilation to portable intermediate representation		
2	- Language translation pipeline: parsing, optional type-checking, translation, linking, execution - Execution as native code or within a virtual	Declarative POV-Ray	

- MACHINE
  - ALTERNATIVES LIKE DYNAMIC LOADING AND DYNAMIC (OR "JUST-IN-TIME") CODE GENERATION
- GRAMMARS - BNF
- SCANNING (LEXICAL ANALYSIS) USING REGULAR EXPRESSIONS
- HIGH-LEVEL PROGRAM REPRESENTATIONS SUCH AS ABSTRACT SYNTAX TREES
- PARSING STRATEGIES INCLUDING TOP-DOWN (E.G., RECURSIVE DESCENT, EARLEY PARSING, OR LL) AND BOTTOM-UP (E.G., BACKTRACKING OR LR) TECHNIQUES; ROLE OF CONTEXT-FREE GRAMMARS

3	- A TYPE AS A SET OF VALUES TOGETHER WITH A SET OF OPERATIONS - PRIMITIVE TYPES (E.G., NUMBERS, BOOLEANS) - COMPOUND TYPES BUILT FROM OTHER TYPES (E.G., RECORDS, UNIONS, ARRAYS, LISTS, FUNCTIONS, REFERENCES) - ASSOCIATION OF TYPES TO VARIABLES, ARGUMENTS, RESULTS, AND FIELDS	DYNAMIC, SCRIPTING (FUNCTIONAL, OBJECT-ORIENTED (CLASS-BASED), IMPERATIVE) PYTHON	PROGRAMMING LAB 1 DUE
4	- SCOPE AND BINDING RESOLUTION - TYPE CHECKING	PYTHON	PROGRAMMING LAB 2 DUE
5	CONTROL STRUCTURES, POINTERS, UNION, STRUCT, GOTO, PREPROCESSOR	IMPERATIVE, PROCEDURAL ANSI C	MIDTERM
6	SUBROUTINE CALLS AND RETURNS, STACK EXECUTION, POINTER TO FUNCTION	ANSI C	PROGRAMMING LAB 3 DUE
7	- LAMBDA CALCULUS - EFFECT-FREE PROGRAMMING - FUNCTION CALLS HAVE NO SIDE EFFECTS, FACILITATING COMPOSITIONAL REASONING - VARIABLES ARE IMMUTABLE, PREVENTING UNEXPECTED CHANGES TO PROGRAM DATA BY OTHER CODE - DATA CAN BE FREELY ALIASED OR COPIED WITHOUT INTRODUCING UNINTENDED EFFECTS FROM MUTATION	FUNCTIONAL HASKELL	
8	- PROCESSING STRUCTURED DATA (E.G., TREES) VIA FUNCTIONS WITH CASES FOR EACH DATA VARIANT - ASSOCIATED LANGUAGE CONSTRUCTS SUCH AS DISCRIMINATED UNIONS AND PATTERN-MATCHING OVER THEM - FUNCTIONS DEFINED OVER COMPOUND DATA IN TERMS OF FUNCTIONS APPLIED TO THE CONSTITUENT PIECES - FIRST-CLASS FUNCTIONS (TAKING, RETURNING, AND STORING FUNCTIONS)	HASKELL	PROGRAMMING LAB 4 DUE
9	- CLAUSAL REPRESENTATION OF DATA STRUCTURES AND ALGORITHMS - UNIFICATION - BACKTRACKING AND SEARCH - CUTS	LOGIC (DECLARATIVE) PROLOG	
10	OVERALL COMPARISONS, ANALYSIS, SUMMARY AND CONCLUSIONS	PROLOG	PROGRAMMING LAB 5 DUE
11	FINAL EXAM		

Course Requirements

Notes
It is the policy of the Computer Science department that you must receive a passing grade on the Final exam (60% or higher) in order to pass the class.

Exams
There will be one 50-minute midterm exam and a written final, which will be comprehensive. These will each be closed book, closed notes, written exams.

Overall, you must assume that ANY material covered during lecture, assigned for homework or labs, or assigned for readings may appear on an exam.

Homework & Programming Assignments
All assignments will be given a due date when assigned. If your work is not turned in by the assigned time you will receive a zero - I will not accept any late work. Start your assignments early to avoid missing the due date and time!

There may be periodic homework assignments, posted electronically on the class web page. They will include required reading and questions pertaining to that (or previously covered) material. The work you turn in must be typed (preferably using the same electronic version I `handed' out) and composed in proper English - free from grammar and spelling mistakes. When calculations, algebraic equations or complex mathematical symbols are needed, they may be written in by hand on your printed HW and scanned. Your homework will be submitted electronically on Moodle.

There will also be four or more programming intensive assignments throughout the term. These programming labs will require you to solve simple problems by writing code in different languages - likely languages that you have not previously learned. The purpose of these labs is to give you first-hand experience using features of these languages that we have examined during lecture. Therefore, prior knowledge of these languages is not required. You will need to turn in your source code electronically.

Attendance
Attendance of lecture is very important to your success in this course. If attendance becomes a problem, it will become mandatory and will be included as a graded item in your course grade.

Grading Scale

Components				
20% Midterm	A	92-100%	C	72-77%
50% Programming labs, written homework or papers	A-	91-92%	C-	70-71%
30% Final written exam	B+	88-89%	D+	68-69%
	B	82-87%	D	62-67%
	B-	80-81%	D-	60-61%
	C+	78-79%	F	0-59%

Academic Honesty Policy

CODE OF STUDENT RESPONSIBILITY 574-031-0030 SPECIFIC STANDARDS AND POLICIES

THE FOLLOWING LIST OF PROHIBITED FORMS OF CONDUCT IS NOT ALL INCLUSIVE SINCE IT IS NOT POSSIBLE TO LIST ALL POTENTIAL   VIOLATIONS. THE UNIVERSITY REQUIRES THAT ALL STUDENTS BEHAVE IN A MANNER CONGRUENT WITH ESTABLISHED COMMUNITY STANDARDS AND IN A MANNER CONDUCTIVE TO THE DEVELOPMENT OF THE INDIVIDUAL. ACTIONS DETRIMENTAL TO THE MISSION OF THE UNIVERSITY AND THE LEGITIMATE ACTIVITIES OF THE ACADEMIC COMMUNITY WHICH CONSTITUTE THE UNIVERSITY ARE IN VIOLATION OF THIS CODE AND MAY BE SUBJECT TO JUDICIAL PROCEDURES.

*ACADEMIC DISHONESTY, WHICH INCLUDES BUT IS NOT LIMITED TO:

CHEATING: INTENTIONAL USE OR ATTEMPTED USE OF ARTIFICE, DECEPTION, FRAUD, AND/OR MISREPRESENTATIONS OF ONE'S ACADEMIC WORK;

FABRICATION: UNAUTHORIZED FALSIFICATION AND/OR INVENTION OF ANY INFORMATION OF CITATION IN ANY ACADEMIC EXERCISE;

FACILITATING DISHONESTY: HELPING OR ATTEMPTING TO HELP ANOTHER PERSON COMMIT AN ACT OF ACADEMIC DISHONESTY. THIS INCLUDES STUDENTS WHO SUBSTITUTE FOR OTHER PERSONS IN EXAMINATIONS OR REPRESENT AS THEIR OWN PAPERS, REPORTS, OR ANY OTHER ACADEMIC WORK OF OTHERS;

PLAGIARISM: REPRESENTING WITHOUT GIVING CREDIT THE WORDS, DATA, OR IDEAS OF ANOTHER PERSON AS ONE'S OWN WORK IN ANY ACADEMIC EXERCISE. THIS INCLUDES SUBMITTING, IN WHOLE OR IN PART, PREWRITTEN TERM PAPERS OF ANOTHER OF RESEARCH OF ANOTHER, INCLUDING BUT NOT LIMITED PRODUCT OF COMMERCIAL VENDOR WHO SELL OR DISTRIBUTE SUCH MATERIALS. AND THE APPROPRIATION OF AND/OR USE OF ELECTRONIC DATA OF ANOTHER PERSON OR PERSONS AS ONE'S OWN, OR USING SUCH DATA WITHOUT GIVING PROPER CREDIT FOR IT; OR

ANY USE OR ATTEMPTED USE OF ELECTRONIC DEVICES IN GAINING AN ILLEGAL ADVANTAGE IN ACADEMIC WORK IN WHICH USE OF THESE DEVICES IS PROHIBITED, AND SUCH DEVICES INCLUDE BUT ARE NOT LIMITED TO CELL PHONES, PDAS, LAPTOPS, PROGRAMMABLE CALCULATORS, ETC.

PROGRAMMING LANGUAGE SOURCE CODE IS NO DIFFERENT WHERE ACADEMIC HONESTY IS CONSIDERED. THE CODE YOU WRITE IN A COMPUTER SCIENCE COURSE IS NOT ESSENTIALLY DIFFERENT FROM THE PAPER YOU WRITE FOR A LITERATURE OR HISTORY CLASS. THEY ARE BOTH YOUR OWN WORK AND IDEAS.

DISABILITY STATEMENT

IF YOU HAVE A DOCUMENTED DISABILITY THAT REQUIRES ACADEMIC ACCOMMODATIONS AT WESTERN OREGON UNIVERSITY, YOU ARE REQUIRED TO HAVE YOUR ACCOMMODATIONS COORDINATED THROUGH THE OFFICE OF DISABILITY SERVICES (ODS). ODS IS LOCATED IN APSC, RM. 405. PHONE: 503-838-8250 VTTY. EMAIL: ODS@wou.edu

VETERANS AND ACTIVE DUTY MILITARY PERSONNEL WITH SPECIAL CIRCUMSTANCES ARE WELCOME AND ENCOURAGED TO COMMUNICATE THESE, IN ADVANCE IF POSSIBLE, TO THE INSTRUCTOR.

WOLF CONNECTION SYSTEM REFERRAL PROGRAM

STUDENTS IN THIS CLASS MAY BE REFERRED TO THE WOU STUDENT SUCCESS SPECIALIST (SSS) IF THE INSTRUCTOR DETERMINES THEIR PERFORMANCE IN THE CLASS IS PLACING THEM AT ACADEMIC RISK. THE SSS WILL OFFER TO WORK WITH REFERRED STUDENTS TO ADDRESS ISSUES AND DEVELOP A STUDENT SUCCESS STRATEGY. IRRESPECTIVE OF WHETHER A REFERRAL HAS OR HAS NOT BEEN MADE, YOU ARE ULTIMATELY RESPONSIBLE FOR TRACKING YOUR OWN PROGRESS IN THIS COURSE.