COMP 333 Concepts of Programming Languages Fall 2016

INSTRUCTOR: Professor Diane Schwartz

# OFFICE: JD 4401

EMAIL: diane.schwartz@csun.edu

Moodle Site: moodle.csun.edu

OFFICE HOURS: TTH 9:30 – 10:30

Section: 15965 Meets TTH 8:00-9:15 in JD 3508

Section: 15966 Meets TTH 11:00-12:15 in JD 3508

PREREQUISITES: Comp 282

COURSE DESCRIPTION: Discussion of the fundamental concepts that underlie programming language syntax and semantics through a comparative study of several language paradigms and their features. Language paradigms studied: object-oriented programming, functional programming, logical programming and concurrent programming. Overview of the programing language translation process.

REQUIRED TEXTBOOK: Programming Language Pragmatics (Fourth Edition) (Third Ed ok)

 by Michael L Scott, Published by Morgan Kaufmann

GRADING: Midterm Exams (2) 40% (20% each)

 Final Exam 25%

 Quizzes/Programming Projects 30%

 Class Participation 5%

 (Attendance, Classroom activities)

Grades will be based on percentage of total points earned. Plus and minus grading will be used.

A-, A : 90 – 100%; B -, B, B+: 80 – 89%; C -, C, C+: 65 – 79%; D: 60 – 64%; F: < 60%

COURSE OBJECTIVES

A successful student will be able to:

1. Summarize the evolution of programming languages illustrating how this history has led to the paradigms available today.
2. Explain the models of translating high level languages to machine language, including the phases and files in the compilation process.
3. Demonstrate the use of BNF to describe concrete syntax, and to apply the BNF structures to translation of programs.
4. Demonstrate different forms of binding, visibility, scoping, and lifetime management.
5. Explain the importance of types and type-checking in providing abstraction and safety.
6. Evaluate the tradeoffs between the different paradigms, considering such issues as space efficiency, time efficiency, safety, and power of expression.
7. Design, code, test, and debug basic programs using the functional, logical, and concurrent paradigm
8. Explain the communication and synchronization models of concurrent processes.
9. Explain the importance of abstractions, especially how abstraction mechanisms support the creation of reusable software components.

Additional Class Information

1. Class attendance/participation counts for 25 points and will be scaled to 5% of your grade. You will get the full 25 points if you miss no more than 3 class meetings. Absences beyond 3 class meetings, will cause you to lose 5 pts per missed class. You will not lose more than 25 points total. To be counted as attending class you need to sign the attendance sheet at the beginning of class, participate in the classroom activities and remain in class until the end of class.
2. Students are expected to read and abide by the University's Academic Honesty statement printed in the current catalog. Academic dishonesty will result in a zero on the assignment and may result in class failure.
3. There are no make-up exams.
4. Lecture power points will be posted on Moodle.

Class Etiquette

1. No use of cell phones, lap tops or other communication/computer devices is allowed unless it is for the purpose of taking class notes. You should have your full attention on the class.
2. Please do not eat in class. Don’t bring in breakfast, lunch, dinner or snacks to eat in class. Bottles of water, small sodas or cups of coffee or tea are ok.
3. Come to class on time and do not leave early. The attendance sign-up sheet will be available only at the beginning of class.
4. Full participation in class activities is expected.

COURSE OUTLINE (Tentative)

|  |  |  |
| --- | --- | --- |
| **Lecture Days** | **Lecture Topics** | **Related Material****from the Scott text Ed. 4** |
| 8/30- 9/1 | Introduction | Chap 1 |
| 9/6 – 9/8; | Programming Language Syntax (BNF and EBNF) | Chap 2  |
| 9/13 -9/15 | Programming Language Syntax (BNF and EBNF) | Chap 2 |
| 9/20 – 9/22 | Names, Scopes and Bindings | Chap 3 |
| 9/27 – 9/29  | Data Types and Type Checking | Chap 7,8 |
| 10/4 | Parameter Passing | Chap 9.3 |
| **10/6** | **Midterm 1** |  |
| 10/11 – 10/13 | Data Abstraction and OOP | Chap 10 |
| 10/18 – 10/20 | Data Abstraction and OOP | Chap 10 |
| 10/25 – 10/27 | Functional Programming (Scheme) | Chap 11 |
| 11/1 – 11/3 | Functional Programming (Scheme) | Chap 11 |
| **11/8**  | **Midterm 2** |  |
| 11/10; 11/15 – 11/17;  | Logic Programming (Prolog) | Chap 12 |
| 11/22 | Logic Programming (Prolog) | Chap 12 |
| 11/29-12/1;  | Concurrent Programming (in Java) | Chap 13 |
| 12/6 – 12/8 | Concurrent Programming (in Java) | Chap 13 |
| **Final Exam** | **8am class Thursday Dec 15 from 8:00 – 10:00** **11am class Tuesday Dec 13 from 10:15 – 12:15** |  |

**See the class Moodle page for homework and programming assignment postings and other information.**