

Computer Science 365

Organization of Programming Languages

Professor – Tom Stokke, Ph.D.

Office – Upson II 366C

Office Hours – Tuesday or Thursday 1:30 – 2:30, or by appointment

Email – thomas.stokke@und.edu

Phone - 777-3337 or 777-4107 (CSci Office)

Required texts

Concepts of Programming Languages, Robert W. Sebesta, 11th ed.

ISBN: 9780133943023

Websites of interest

VirtualBox – <http://www.virtualbox.org>

Download image – examples link in Blackboard

GTA – Mark Arinaitwe

Email - mark.arinaitwe@und.edu

Purpose

This course introduces fundamental concepts in computer programming languages, enabling students to understand the essential aspects of programming languages. Programming paradigms from logic, functional, procedural, imperative, and object-oriented programming will be presented. The course will help students choose an appropriate language for a given task, learn new languages easily, and effectively use features of current languages.

Anticipated material covered to be covered – this is subject to change

Introduction - Chapter 1

History - Chapter 2

Syntax and Semantics - Chapter 3 (3.1 through 3.3)

Scanning and Parsing - Chapter 4

Names, Bindings and Scopes - Chapter 5

Midterm

Imperative Programming – Chapters 6 – 8

Subprograms – Chapters 9 – 10

Object Oriented Programming – Chapters 11 – 12

Functional programming – Chapter 15 (if time)

Logic Programming – Chapter 16 (if time)

Final

Final Schedule

Wednesday, May 13, 1:00 PM

General

- 1) Exams will only be given at the scheduled times unless prior arrangements have been made.
- 2) Regular attendance is expected and students are responsible for any/all material covered in class.
- 3) All in-class assignments are due on the assigned dates. No late assignments are accepted. There are no make-ups for in-class activities unless it is an approved absence. Late programming assignments will be accepted **up to one week late with a 20% deduction**, or **up to two weeks late with a 50% deduction**. No assignments will be accepted more than two weeks late.
- 4) If you have any special needs or concerns, which could include but are not limited to note taking or audio or visual difficulties, you must talk to your instructors. We will try to accommodate your needs, and if needed, will work with you and Disability Support Services to create a productive learning environment.

Grading

Chapter and in-class assignments	25%
Term paper	25%
Midterm exam	25%
Final exam	25%

Letter grades will basically follow 90, 80, 70, and 60 percent. The instructor reserves the right to alter the percentages, but only to the advantage of the student.

Learning Outcomes:

At the end of this class, the students will be able to:

- Describe the general principles that involved in the design of programming languages.
- Categorize programming languages.
- Explain the relationship between syntax and semantics as well as the differences between the two.
- Describe the principal features of different programming language paradigms.
- Demonstrate knowledge of various programming languages syntax, control structures, and compiler features.
- Describe the structure of a typical compiler and interpreter.
- Demonstrate understanding of formal programming language development techniques.
- Identify which type of programming language is applicable to a particular type or class of problem.

Plagiarism

Students are allowed and even encouraged to discuss ideas. Students can “help” other students with algorithms and small sections of code. Students are also expected to do their own work. Students handing in assignments or programs that are not their own work may be given the opportunity to “share” the points. For blatant first-time cases or any second offense all students involved will receive an F for the course and may be reported for academic fraud.