CMPE 260 PRINCIPLES OF PROGRAMMING LANGUAGES
SYLLABUS

Instructor: F. Başak Aydemir (E-mail: basak.aydemir@boun.edu.tr, Room: BM 45)
Assistant: Alper Ahmetoğlu (E-mail: almetoglu.alper@gmail.com)

Course Description:
This course is intended to introduce the student to the main paradigms of computation, languages representative of these paradigms, techniques of implementing various programming language constructs, as well as basic concepts relating to the specification of programming languages.

Syntax and semantics of programming languages, basic constructs that are common to all languages (control structures, subprograms, etc.), and basic programming paradigms (functional, logic, etc.) will be covered. The intent of the course is not teaching a specific programming language; instead, using several languages to exemplify the main concepts and constructs.

Text Book: Concepts of Programming Languages

Reference Books:
• Foundations of Programming Languages, Seyed H. H. Roosta, Course Technology, 2002
• Programming Language Pragmatics, Michael L. Scott, Elsevier, 2005
• Programming Languages and Methodologies, Robert J. Schalkoff, Jones & Bartlett Pub., 2007

Lecture Hours and Rooms:
To Be Announced

Course Contents:
Introduction (Reasons for studying PLs, Evaluation criteria and related factors)
Syntax (Recognizers, Generators, Grammar types, Context-Free Grammar (CFG), Backus-Naur Form (BNF),
Relationship between CFG and BNF, Derivations, Parse tree, Ambiguity, Extended BNF (EBNF), Syntax graph)
Static Semantics (Attribute grammar)
Semantics (Operational semantics, Denotational semantics)
Logic Programming (Declarative languages, Predicate calculus, Clausal form, Resolution, Prolog programming
language, Facts and rules, Backtracking, Lists, Limitations of Prolog)
Functional Programming (Lambda notation, Referential transparency, Functional forms, Lisp and Scheme
programming languages, S-expression, Define and Eval functions)
Binding (Binding time, Early/Late binding, Static/Dynamic binding, Name-declaration/Declaration-reference/Reference-value binding, Blocks)
Lifetime
Referencing Environment (Visibility, Local/Nonlocal/Global variables)
Scope (Scope rules, Referencing operations, Static/Dynamic scoping)
Subprograms (Protocol and parameter profile (signature), Formal/Actual parameters, Positional/Keyword
parameters, Default parameters, Feature interaction, Subprogram overloading, Data exchange between
subprograms, Parameter passing, Aliasing, Eager/Normal-order/Lazy evaluation)
Subprogram Implementation (Activation record, Static/Dynamic chain, Static chain method, Display method,
Deep/Shallow access, Subprograms as parameters, Labels as parameters, Generic subprograms,
Separate/Independent compilation)
Control Structures (Selection statements, Two-way/Multiple selectors, Iterative statements, Guarded commands)

Evaluation (subject to change):
• Midterm 1: 15%
• Midterm 2: 20%
• 2 Projects: 40% (2 * 20%)
• Final: 25%

Notes:
• The midterm and final examinations will be “closed books and notes”.

• You can follow the announcements about the course from Moodle.
• Attendance for both the midterm exams and the final exam, and submitting both projects are obligatory. Otherwise, you will fail the course, regardless of the grades obtained in other parts of the course.