



UNIVERSITY OF NEW YORK IN TIRANA Course Syllabus

Course : **Compilers** (4 credit hours)
Lecturer : **Vijaya Raju. M**
Office Hours : **By Appointment.**
Phone : **0693 44 88 36**
E-mail : vrnullagiri@unyt.edu.al

Catalog Description:

The course will deal with Modern Compiler Principles and Techniques applied to general-purpose programming language. The goal of this course is to give students a working knowledge of the foundations, tools, and engineering approaches used in developing Formal Language Translators. The emphasis is on the construction of Compilers to position students to build translators for simple High level languages.

Prerequisite: Programming in C, Assembly Language Programming.

Required Readings,

Text : **Compilers** Principles, Techniques & Tools, by Alfred V. Aho, Ravi Sethi, Jeffrey D. Ullman. Pearson Education, ISBN 81-317-0789-X

Required Additional Materials

Students will be provided hand outs and extra notes as and when required.

Contents:

1. Introduction to Compiling

- Overview of Compilers
- Analysis of the Source Program
- The phases of a compilers
- Compiler Construction tools

2. Simple One-pass Compiler

- Overview
- Syntax-directed translation
- Parsing

3. Lexical analysis:

Role of Lexical analyser

Tokenization

Finite automata

Common patterns

4. Parsing Techniques :

Parsers,

Types of Parsing

LR Parsers

SLR Parser

CLR Parser

LALR Parser

5. Syntax Directed Translation

SDT schemes and Implementation

Intermediate code

Postfix notation

Three address coding

Translations

Procedure calls

6. Code Optimisation:

Sources of Optimisation

DAG representation

Global Data flow Analysis

7. Code Generation

Problems in code generation

Simple Code generator

Symbol tables

Error detection and recovery

Runtime storage administration

Course Objective:

- To provide students with an understanding of the major phases of a compiler.
- To introduce students to the theory behind the various phases, including regular expressions, context-free grammars, and finite state automata.
- To provide students with an understanding of the design and implementation of a Compiler
- To have knowledge of how to build a Compiler, through type checking and intermediate code generation, for a high level language.

Course Requirements

Participation: Participation extends beyond mere attendance. Expect your instructor to keep track of how often you contribute to class discussion (as a whole), particularly during the panel discussion section. You may miss up to three classes without penalty - your first two absences count whether you have a good excuse or not. Each absence beyond the first three will cost you points off of your participation grade. The only exceptions to this rule are severe illness (doctor's note required) and UNYT approved trips/activities. Appropriate documentation for absences beyond the first three is necessary the class day directly before or after the one you miss. In general: this class is intensive and interactive. Missing class could seriously affect your grade! Students are reminded not to approach the instructor for copies of the previous week's materials during immediately before, during, or immediately after class. Students are expected to collect materials from their classmates or see the instructor during consultation hours.

Exams: Two examinations will be taken, a midterm and a final exam covering all course content during the final examination period. Test format may combine a mixture of short answer, true/false, matching, sort answer, and one or two essay questions covering all readings, lecture, hand-out and class discussion content.

Final Examination: TBA

General Requirements

Late assignments and absence from tests will not be tolerated. In the event of illness or emergency, contact your instructor IN ADVANCE to determine whether special arrangements are possible. The University's rules on academic dishonesty (e.g. cheating, plagiarism, submitting false information) will be strictly enforced. Please familiarize yourself with the STUDENT HONOUR CODE, or ask your instructor for clarification.

Criteria for Determination of Grade, including Evaluation Methods

Assignments(Home Works /Seminars)	20%
Midterm	30%
Final	40%
Attendance	10%

Total	100%
--------------	-------------

Grading Scale

Letter Grade	Percent (%)	Generally Accepted Meaning
A	96-100	Outstanding work
A-	90-95	
B+	87-89	Good work, distinctly above average
B	83-86	
B-	80-82	
C+	77-79	Acceptable work
C	73-76	
C-	70-72	
D+	67-69	Work that is significantly below average
D	63-66	
D-	60-62	
F	0-59	Work that does not meet minimum standards for passing the course

Bibliography (Additional Readings)

- AHO, A.V. and J.D. Ullman The Theory of Parsing ,translation and Compiling, Vol 1:Parsing Prentice-Hall Englewood Cliffs, N.J.
- AHO, A.V and J.D. Ullman , Principles of Compiler Design, Addison- Wesley, reading, Mass.
- Ershov, “ The Alpha Automatic Programming System, Academic Press, New York.
- Systems Programming by John J. Donovan Tata Mc. Hill Edition (Chapter 8)
- Theory of Computer Science by KLP Mishra& N.Chandra Sekhran(Chapters 2,3,4).
- Compiler design in C- Allen I.Holub, Prentice Hall (PHI).
- System Programming & Operating Systems by Dhamdhare , Tata Mc. Hill Edition.

**Prepared by
Vijaya Raju.M**