NATIONAL UNIVERSITY
of Computer & Emerging Sciences, Lahore

Department of Computer Science

CS402 – Compiler Construction

Spring 2020

Instructor Name: Aamir, Raheem
Office Location/Number: Liberty lab
Email address: aamir.raheem@nu.edu.pk
Office Hours: (after your class)
Phone: 111-128-128 X 257

Course Information

Program: BS **Credit Hours:** 3 **Type:** Elective
Pre-requisites (if any): Data Structures, Automata Theory

Course Description/Objectives/Goals:

Compilers have been an essential part of any computer system. These are used not only to translate high-level programs into low-level code, but also to translate documents from one format to another; and to parse SQL statements, HTML code and shell scripts.

Besides its applications, the course has an academic value as well. The course helps students to develop a better understanding of the Theory of Automata, Formal Languages, and Computation. It also teaches students how to develop formal specifications of computational machines, and then how to realize these specifications into live working software.

Course Learning Outcomes (CLOs):

At the end of the course students will be able to:	**Domain**	**BT* Level**
Design and build a lexical analyzer		
Design and build a parser		
Design and build a translator		
Construct a virtual machine or interpreter		
Implement a set of DFA's in a programming language		
Implement a given translation scheme		

* BT= Bloom’s Taxonomy, C=Cognitive domain, P=Psychomotor domain, A= Affective domain.

Bloom's taxonomy Levels:1. Knowledge, 2. Comprehension, 3. Application, 4. Analysis, 5. Synthesis, 6. Evaluation

Course Textbook

Compilers – Principles, Techniques and Tools (Second Edition)
By Aho, Lam, Sethi and Ullman
(Commonly known as the Dragon Book)

Additional references and books related to the course:

Compiler Construction – Principles and Practice
By Kenneth C. Louden

Tentative Weekly Schedule

Week	Topics to be covered	Assignments
1	Introduction	
2	Regular Expressions & DFA	
3	Lexical Analysis	Lexical Analyzer
4	Context Free Grammars	
5	Parsing	Parser
6	Syntax-Directed Translation	
7	Handling Left Recursion	
8	Intermediate Code Generation	Translator
9	Type Checking & Scope	
10	Interpreting Intermediate Code	Interpreter
11	Stack-dynamic Storage	
12	Automated Compiler Generation	
13	Bottom-Up Parsing	
14	(Exercises)	

(Tentative) Grading Criteria

1. Assignments (30%)
2. Midterm Exams (30%)
3. Final Exam (40%)

Course Policies

1. 80% attendance is required to appear in the exam
2. Plagiarism is not tolerable in any of its form. Minimum penalty would be an ‘F’ grade in the course. Automated tools may be deployed to detect pirated copies. Students bear all the responsibility for protecting their assignments. In case of cheating, both parties will be considered equally responsible.
3. Project deliverables must be submitted in time. Late submissions (maximum one week) would result in deduction in marks. Only the submitted articles will be marked!