Course Title : [Sylabus] Formal Languages and Compilers 2022-1st Semester

Course	Category	Major selection (major selection)	Instructor	Department or Division	School of Electrical and Computer Engineering
	Number(section)	40105(01)		Name	
	Title	Formal Languages and Compilers		Phone	
	Credit(Hours)	3 Credit(3 Hours)		E-mail	
	Type	lecture		Homepage	
	Time(Room)	Mon 07,08,09/19-108/109		Office Hours	
	school year	4 years	Assistant	name & phone	

Grading	Evaluation Method	absolute evaluation			
	□ Attendance (10%)	□ Portfolio (0%)	□ Participation (0%)		
	□ Assignment (10%)	□ Quiz (0%)	□ Midterm Report (0%)	□ Midterm Exam (40%)	
	□ Final Report (0%)	□ Final Exam (40%)	□ Other (0%)		
Type	Lecture and Practice , PBL , Foreign Language , Convergence				
Teaching Method	Lecture , Practice , Design , Project				
Plagiarism Policy	It is considered plagiarism to draw any idea or any language from someone else wihout adequately crediting that source in your work. It doesn't matter whether the source is a published author, another student, a Web site without clear authorship, a Web site that sells academic papers, or any other person: Taking credit for antone else's work is stealing, and it is unacceptable in all academic situations, whether you do it intentionally or by accident.				

Any student with a disability is welcome to contact the instructor to get academic accommodations, and may be in touch with the Student Accessibility Services by calling 02-6490-6273 to discuss the process for requesting accommodations.

Course Objectives

This course provides fundamental concepts of formal languages and skills to design a compiler. Topics included are finite automata, regular expressions, regular languages, regular grammar, finite automata with output, pushdown automata, context-free languages and context-free grammars, parsing techniques, turing machine and unrestricted grammars.

Course Description	Textbooks and Reference Materials
Yuhan Automata, Pushdown Automata, Regular Language, Non-Context Language, Turing Machine, etc. Learn the basic theory of compiler in Vocabulary analysis, syntax analysis, semantic analysis, code generation, optimization, etc. The roles and principles of each step are covered. Also, the compiler auto-generation tool Learn the skills to implement a small-scale compiler using	David Galles, Modern Compiler Design

Specialty competency	Representative competency
Knowledge Application	Primary
Analysis Experiment	
Problem Definition	
Resource Utilization	Secondary
Planning Ability	
cooperative ability	
Communicative Skills	

Specialty competency	Representative competency
Effect Understanding	
Vocational Ethics	

Course Title: Formal Language and Compiler **2022year 1st Semester**

[Weekly Lesson Plan]

Week	Contents	Teaching Method	Teaching Materials	Requirements, Assignments, etc.
One	Introduction	Lecture		Textbook
2	Lexical Analysis I	Lecture		Textbook
3	Lexical Analysis II	Lecture		Textbook
4	Context-free Grammers	Lecture		Textbook
5	Top-down Parsing I	Lecture		Textbook
6	Top-down Parsing II	Lecture		Textbook
7	Bottom-up Parsing	Lecture		Textbook
8	Review and Evaluation	Questions and Answers, Test		Textbook
9	Abstract Syntax Trees	Lecture		Textbook
10	Semantic Analysis I	Lecture		Textbook
11	Semantic Analysis II	Lecture		Textbook
12	Supplementary Week			
13	Assembly Trees	Lecture		Textbook
14	Code Generation	Lecture		Textbook
15	Memory Management	Lecture		Textbook
16	Review and Final exam	Questions and Answers, Test		Textbook