Course Code	CSE 301/CSE 601
Course Name	Compilers
Credits	4
Course Offered to	UG/PG
Course Description	The course covers principles of compiler design including introduction to compiler technology, overview of compiler followed by in depth study of various phases and component of compilers including lexical analysis, syntax analysis, semantic analysis, intermediate code generation, run time environment as well as machine code generation and optimization. The student will learn writing a complete working compiler in a modular fashion through intensive lab assignments and will be exposed to some advanced topics in compilers

Pre-requisites		
Pre-requisite (Mandatory)	Pre-requisite (Desirable)	Pre-requisite(other)
MTH100, CSE101, CSE201, CSE102	Automata Theory	

*Please insert more rows if required

Post Conditions*(For suggestions on verbs please refer the second sheet)				
CO1	**CO2**	**CO3**	**CO4**	**CO5**
Srudents are able to learn the applications and impoirtance of compiler technology as well as various phases of compilers	Students are able to learn the algorithms and tools for lexical analysis phase of compiler and will gain hands on experience building lexical analyzer for complete object oriented language	Students are able to learn the algorithms and tools for syntax analysis phase of compiler including different kind of parsing and will gain hands on experience building parser for an object oriented language	Students are able to learn the algorithms and techniques for semantic analysis and type checking phase of compiler including data structure and techniques for run time support and will gain hands on experience building type checker for a strongly typed object oriented language	Students are able to learn the issues and algorithms involved in code generation and important optimization and build a code generator for an object oriented language

Weekly Lecture Plan			
Week Number	**Lecture Topic**	**COs Met**	**Assignment/Labs/Tutorial**
1	Compiler introduction and overview	CO1	Cool Tutorial
2	Lexical analysis	CO1,CO2	Flex Tutorial and Lexical analyzer assignment
3	Parsing	CO1, CO3	Bison tutorial
4	Parsing	CO1,CO3	Parser development cool language
5	Syntax directed translation	CO1,CO4	
6	Syntax directed translation	CO1,CO4	Type Checking for COOL Language
7	Run time environment	CO1, CO4	

8	Run time environment	CO1, CO4	
9	Intermedia Code generation	CO1, CO5	Code Generation for Cool
10	Machine independent optimization	CO1, CO5	
11	Machine Code generation	CO1, CO5	
12	dependent optimization	CO5	
13	dependent optimization	CO5	

14			
Week Number	**Laboratory Exercise**	**COs Met**	**Platform (Hardware/Software)**
1,2	Lexical Analyzer	CO1, CO2	Linux/COOL
3,4	Syntax Analyzer	CO1, CO3	Linux/COOL
5,6,7,8	Semantic Anlyzer	CO1, CO4	Linux/COOL
9,10,11,12	Code Generation and Optimization	CO1, CO5	Linux/COOL/SPIM/LLVM

*Please insert more rows if required

Assessment Plan

Type of Evaluation	**% Contribution in Grade**
Mid Term	20
Final	30
Lab Assignment (4 to 5)	40
Class Participation/Quiz/Homework	10

Resource Material

Type	**Title**
Textbook	Compiler Principles, Techniques and Tools by Alfred Aho, Monica Lam, Ravi Sethi and Jeffrey Ullman
Reference Book	Compiler Design and Implementation by Steven Muchnick
Reference Material	Papers from Relevent Conferences and Journals