Department of Computer Engineering

Compiler Construction (66416)

Total Credits	**3**

major elective

Prerequisites	P1 : Data Structure and Algorithms (66211)

Course Contents

1. Overview of compiler components Scanner Parser Symantec Checker Code Generator Symbol Table. Optimization Write the Symbol Table for the Course Project. 2. Scanner- Tokens Regular Expressions Finite State Automata DFSA and NDFSA Building Scanners based on FSA Ad-hoc Scanner. Write the Scanner for the course project. 3. Grammar and Parsing Context Free Grammar -CFG Language Definition Operator Precedence Associcativity Push-Down Automata 4. Building Predictive Parsers FIRST and FOLLOW sets LL(1) Parsers Recursive Descendent Parsers Table Driven Parsers Abstract Syntax Trees Write the Parser for the Course Project. 5. MIPS architecture- Target Machine for course Project. Call By Value Call By Reference Call By Value Return 6. Variables Scopes Lexical Scoping Dynamic Scoping Modify the Symbol Table for the Course Project. Write the Symantec Checked for the course Project. 7. Dynamic Memory Allocation and Garbage Collection Allocating Memory Reference and Counting Mark and Sweep Generation Scavenging 8. Code Generation Preorder and Post Order Traversals of AST Generating code for Expressions A Generating code for assignment statements Generating code for If statements Code Generation for Loops Start the Code Generator for the course project. 9. Register Allocation Allocation in Expressions Allocation in Basic Blocks Global Register Allocation Inter-procedural Register Allocation 10. Register Allocation Allocation in Expressions Allocation in Basic Blocks Global Register Allocation Inter-procedural Register Allocation 11. Assemblers Assemblers, Linkers, and Debuggers Writing Assemblers 12. Optimization Program and Flow Analysis Constant Propagation Live Variable Analysis Dead Code Detection Available Expressions Transformations.

Intended Learning Outcomes (ILO's)		**Student Outcomes (SO's)**	**Contribution**
1	Gain Fundamental Knowledge in the theory of Compilers Construction and apply the knowledge in developing a Compiler	K	25 %
2	Acquire knowledge in Finite State Automata, regular Expressions, Context Free Grammar, optimizations	C	25 %
3	Gain Practical Knowledge of compiler main components: Scanner, Parser, Symantec Checker, Optimizer, and Code Generator	K	25 %
4	The ability to describe and define a New Programming Language and write a complete	E	25 %

Textbook and/ or Refrences

Text Book:: Compilers: Principles, Techniques, and Tools, Alfred V.Aho, Ravi Sethi, and Jeffrey D. Ulman, Addison Wesley

Assessment Criteria	**Percent (%)**
First Exam	20 %
Second Exam	20 %
Homeworks	15 %
Final Exam	45 %

Course Plan

Week	Topic
1	Overview of Compiler
2	Regular Expressions and finite State Automata
3	Scanners: AD-HOC, Table Driven
4	Context Free Grammar
5	Expressions, LL1 Grammar
6	Building Predictive Parsers
7	First Exam
8	Building Predictive Parsers, Symantec Checkers
9	Building Predictive Parsers, Symantec Checkers.
10	MIPS, Variables: Scoping
11	Dynamic memory Allocation, Garbage collection
12	Code generation: Assignments, Expressions and Loops
13	Code Generation: Functions, Register Allocations
14	Second Exam
15	Assemblers, Optimizations
16	Optimizations.
16	Final Exam