

INDIAN INSTITUTE OF TECHNOLOGY TIRUPATI
PROFORMA FOR NEW COURSE

1.	Title of the Course	Compiler Design
2.	Course Number	CS3103
3.	Status of the Course	Core
4.	Structure of Credits	3-0-0-3
5.	Offered To	UG
6.	New Course/Modification to	New
7.	To be Offered by	Department of Computer Science and Engineering
8.	To take effect from	July 2018
9.	Prerequisite	Nil
10.	Whether approved by the Department	Yes
11.	Course Objective: To learn the principles, techniques and tools required in developing compilers in a systematic way; To gain an understanding on different theoretical and systems concepts from computer science coming together in building a compiler.	
12.	Course Content: Lexical analysis: regular expressions, tokens, Lex; Syntax analysis: CFG, top-down parsing, bottom-up parsing, SLR, LR(1), LALR, parsers for ambiguous grammar, Yacc; Semantic analysis: attribute grammars, SDDs, SDTs, evaluation order, static checking, intermediate representations, type expressions/conversions, expression translation, control flow; Run time environments: storage organization, stack allocation, heap management, garbage collection; Code generation: programs, instructions, addresses, basic blocks and flow graphs; Optimization of basic blocks, register allocation and assignment; Machine independent optimizations: sources of optimization, data flow analysis.	
13.	Text book(s): 1. Aho A, Lam M, Sethi R and Ullman J D, <i>Compilers: Principles, Techniques</i> , Addison-Wesley (2007). 2. Appel A W and Palsberg P, <i>Modern Compiler Implementation in Java</i> , Cambridge University Press (2002).	
14.	Reference(s): 1. Allen R and Kennedy K, <i>Optimizing Compilers for Modern Architectures: A Dependence-based Approach</i> , Morgan Kaufmann (2001). 2. Cooper K and Torczon L, <i>Engineering a Compiler</i> , Morgan Kaufmann (2003). 3. Mogensen T E, <i>Introduction to Compiler Design</i> , Springer (2011).	