Compiler Design - Video course

COURSE OUTLINE

This course aims to teach students the principles involved in compiler design. It will cover all the basic components of a compiler but not the advanced material on optimizations and machine code generation.

The treatment will be at the level of a graduate course.

COURSE DETAIL

Module No.	Topic	Number of Hours	Lecture Numbers in which module is covered
1	Overview of compilation	1.5	1,2
2	Run-Time Environments	2	2,3,4
3	Local Optimizations	1	4,5
4	Machine code generation	3	5,6,7,8
5	Global Register Allocation	2	8,9,10
6	Implementing Object-Oriented Languages	1	10,11
7	Introduction to Machine-Independent Optimizations	1	11,12
8	Data-Flow Analysis	2	12,13,14

NPTEL

http://nptel.iitm.ac.in

Computer Science and Engineering

Pre-requisites:

An undergraduate course in automata theory and good knowledge of programming and Linux. No prior knowledge of compiler design will be assumed.

Additional Reading:

Mini projects involving compiler implementation.

Coordinators:

Prof. Y.N. Srikanth
Department of Computer Science and AutomationIISc Bangalore

9	Control-Flow Analysis	1.5	14,15
10	Machine-Independent Optimizations	2.5	16,17,18
11	Data-Flow Analysis: Theoretical Foundations	1	18,19
12	Partial Redundancy Elimination	1.5	19,20
13	The Static Single Assignment Form	3	21,22,23
14	Automatic Parallelization	4	24,25,26,27
15	Instruction Scheduling	3	28,29,30
16	Software Pipelining	1	31
17	Energy-Aware Software Systems	4	32,33,34,35
18	Just-In-Time Compilation	1	36
19	Garbage Collection	1	37
20	Inter-procedural Data-Flow Analysis	1	38
21	Worst Case Execution Time Estimation	2	39,40
	Total	**40**	

Note: Some of the lectures partially cover two modules. For example, Lecture no. 10 contains ending part of module 5 and beginning part of module 6. In such cases, the same lecture number is mentioned in two successive modules.

References:

- A.V. Aho, M.S. Lam, R. Sethi, and J.D. Ullman, Compilers: Principles, Techniques, and Tools, Pearson Education, 2007 (second ed.).

- K.D. Cooper, and L. Torczon, Engineering a Compiler, Elsevier, 2004.