

# Principles of Compiler Design - Video course

## COURSE OUTLINE

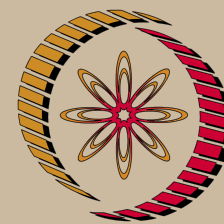
This course aims to teach students the principles involved in compiler design. It will cover all the basic components of a compiler but not the advanced material on optimizations and machine code generation. The treatment will be at the level of a graduate course.

## COURSE DETAIL

Module No	Topic	No. of Hours
1	An overview of a compiler	1
2	Lexical Analysis	3
3	Syntax Analysis	7
4	Semantic Analysis	5
5	Intermediate Code Generation	3.5
6	Run-Time Environments	3
7	Local Optimizations	1
8	Machine Code Generation	4
9	Global Register Allocation	2.5
10	Machine-independent Optimization	6.5
11	Instruction Scheduling and Software Pipelining	2
12	Automatic Parallelization	1.5
Total Hrs		40

## 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.



NP-TEL

# NPTEL

<http://nptel.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 Automation IISc  
Bangalore