CODE GENERATION AND OPTIMIZATION

Every Computer Architect should know how a compiler generates code to maximize the efficiency of the processor (micro) architecture. In CGO, you will learn the basics of an optimizing compiler and implement your own optimization passes in a production compiler (LLVM). The topics include control-flow analysis, data-flow analysis, and instruction selection and scheduling, register allocation and a variety of optimizations.

Week	Date	Topic	Handouts	Due
1	Aug 28th	Welcome	R1	
2	Sept 2nd	-		
	Sept 4th	Lexical Analysis	H1	R1
3	Sept 9th	Parsing		
	Sept 11th	Intermediate Representation		
4	Sept 16th	Intro to Control Flow		H1(a)
	Sept 18th	Loop Structures		
5	Sept 23rd	Traces/Superblocks	H2	H1(b)
	Sept 25th	-		
6	Sept 30th	If-conversion		
	Oct 2nd	Liveness	H3	H2
7	Oct 7th	Reaching Definitions		
	Oct 9th	Available Definitions		
8	Oct 14th	Static Single Assignment	H4	H3
	Oct 16th	SSA-Based Optimizations		
9	Oct 21st	Top-Down/Bottom-Up		
	Oct 23rd	Graph Coloring		
10	Oct 28th	**Midterm I**	P1	H4
	Oct 30th	Instruction Selection		
11	Nov 4th	Classic Optimizations		P1.a
	Nov 6th	Loop Optimizations		
12	Nov 11th	Instruction Scheduling		
	Nov 13th	List/Global Scheduling		P1.b
13	Nov 18th	Software Pipelining		
	Nov 20th	I/D-Cache Optimizations		P1.c
14	Nov 25th	DOALL Parallelism		
	Nov 27th	DOACROSS Parallelism		P1.d
15	Dec 2nd	Compiler-driven DVFS		
	Dec 4th	**Midterm II**		P1.e
16	Dec 9th	Code Review		
	Dec 11th	**Final Report**		

Key: P(roject updates); H(omeworks); R(eading assignments – there will be more)

Instructor: Vijay Janapa Reddi (ENS 530), vj@ece.utexas.edu
Teaching Assistant:
Jingwen Leng (ENS 623), jingwen@utexas.edu,
Riddhi Shah (ENS 110), riddhi.j.shah@utexas.edu

Course Code: EE382V
Unique ID: 17285

Course Times: MW 1:30pm – 3:00pm, ENS 126
Office Hours: TBD
TA Hours: TBD

Grading Policy:

Class Problems / Reading Assignments:	20%
Programming Assignments:	20%
Midterm 1:	20%
Midterm 2:	20%
Project:	20%

Pre-reqs: 460N (Computer Architecture)