Syllabus

- Compiler Construction, phase order, compilation as optimisation
- Architecture costs: Parallelism and Latency in uni and multi-processors
- Architecture "independent" optimisation, dataflow analysis, lattices and fixed points
- Code generation, register allocation and scheduling in superscalar and vliw processors
- High level analysis based on dependence analysis. Intra and inter procedural analysis, whole program analysis.
- High level transformations including linear algebraic formalisation, unimodular transformations and space/time representation
- Automated parallelisation. shared and distributed memory models. Linear algebraic approach to parallelisation.
- Adaptive optimisation: Feedback directed optimisation, iterative compilation, program specialisation and dynamic compilation eg JIT, DBT
- Compiler infrastructure case studies: SUIF, Machine SUIF, JIKES
- Current themes:low power compilation, automatic compiler generation and machine learning.
- Relevant QAA Computing Curriculum Sections Compilers and Syntax Directed Tools