# **Final Philosophy**

What the exam looks like.

- Definitions, comparisons, advantages & disadvantages
  - what is it?
  - how does it work?
  - why have it?
  - pay particular attention to the terminology that was highlighted in the slides
- Apply the concepts and techniques you have learned to situations you have (hopefully) not seen before.

The goal is to test your knowledge of the material and how well you can apply it, **not how fast you can tell me what you know**.

- Andrew will take the exam beforehand to make sure that he can finish it in **half** the time you will have.

# **Topics**

Architecture vs. implementation

Design principles

RISC vs. CISC

- Enforcing backwards compatibility
- New instructions & the rationale for including them

ILP & TLP

Pipelining

- Dependences vs. hazards
- Techniques to eliminate hazards
- Precise Interrupts & pipelining

Techniques to reduce branch delays

- Dynamic branch prediction
- Branch target buffers

# **Topics**

Static & dynamic scheduling

- Techniques for static scheduling
- Implementations of dynamic scheduling
  - Tomasulo
  - Physical register pool
  - Reorder buffers
- Preserving precise interrupts
- VLIW computers
- Superscalars vs. VLIW processors

Multithreaded processors

- Coarse-grain, fine-grain & SMT

# **Topics**

Caches

- Configuration tradeoffs

- Hardware & compiler techniques to hide memory latency, reduce memory latency, eliminate memory ops, increase memory bandwidth

Multiprocessors

- The religious war: SM vs. MP

- Cache coherency on bus-based & distributed MIMDs

- Synchronization

Dataflow machines