What does it mean to compute?

• Given problem -> solution
• Input/Output
• Number crunch?
Instruction

• Way to modify state
  – Input (r0, r1, etc)
  – Procedure (ADD, SUB, etc)
  – Output (r0, r2, etc)

• Execution model
  – Von Neumann
What is state?

• Registers
  – Few of them but fast
  – Directly addressed
  – Some have special semantics

• Main memory
  – Large, but slow
  – Direct / Indirect addressed
Instructions

• Minimum set?
  – Access your state
  – Something to change state
  – NOR, subtract-branch-less-than-zero

• Useful set
  – Basic arithmetic, logic
  – Control
    • Comparisons
    • Branches, jumps
  – Memory access
Why add more (CISC)?

• Convey of high-level semantic knowledge to hardware for it to exploit run-time information
• Convenience for assembler programmer
• Legacy support
• Marketing
• Because you can
Does it matter (CISC/RISC)

• Not much if they are converging