Lecture 3

Assembly language

Say something once, why say it again?

CS 241: Foundations of Sequential Programs
Winter 2018

Troy Vasiga et al
University of Waterloo
Review

- MIPS has 18 instructions
- CPU contains 32 32-bit registers
- MIPS reference sheet for encoding of instructions
- cs241.wordasm to create binary files
- mips.twoints to start MIPS machine
Special Registers

- $0
- $29
- $30
- $31
Assembly language

- encoding details can be automated
- 1-1 correspondence with machine language (almost)
- easier on the human eye
- uses different version of the assembler (cs241.binasm)
Example 0 revisited

- A1 level (machine language):
  00000000 10100111 00011000 00100000
  00000011 11100000 00000000 00001000

- A2 level (assembly language):
  add $3, $5, $7
  jr $31

- Which one would you rather read and debug?
Examples 1, 2, 4

- Example 1 – adding two immediate values
- Example 2 – finding absolute values
- Example 4 – loop to sum integers from 1 to 13
See ex4asm.pdf for nicely documented code.
Example 3

- array access
  - need to use mips.array which asks about the size of the array (say \( n \)) then reads in \( n \) integers
Memory Revisited
Example 5 – Output