

# CSE 378, 06wi – Lecture 2 Main Points

## Introduction to the MIPS ISA

January 6, 2006

### Hardware (as exposed to software)

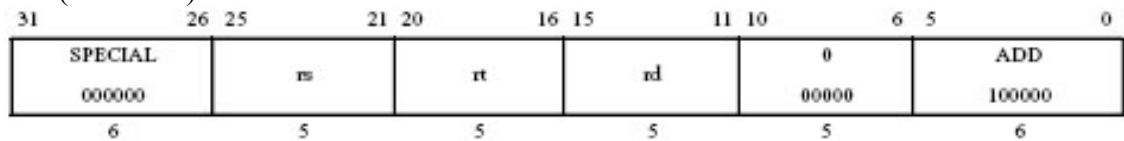
- $1+31 = 32$  registers (register 0 always contains 0)
- $2^{30}$  words of main memory (1 word = 4 bytes = 32 bits)
  - $2^{32}$  bytes of main memory
- a program counter (PC)
  - “points to” the next instruction to execute

### Instruction Set (expressed in assembler)

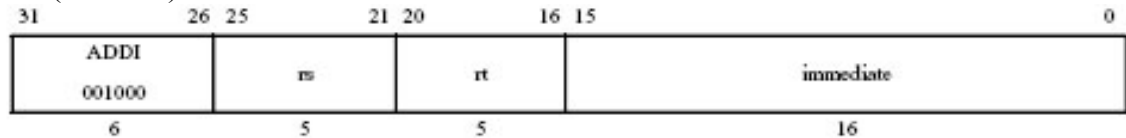
- add rd, rs, rt            Example: add \$8, \$4, \$5
- sub rd, rs, rt            sub \$8, \$5, \$6
- addi rt, rs, immed        addi \$8, \$4, 100
- lw rd, offset(base)      lw \$8, 20(\$4)
- sw rt, offset(base)      sw \$8, 24(\$5)

### Instruction Encoding (binary – what’s stored in memory)

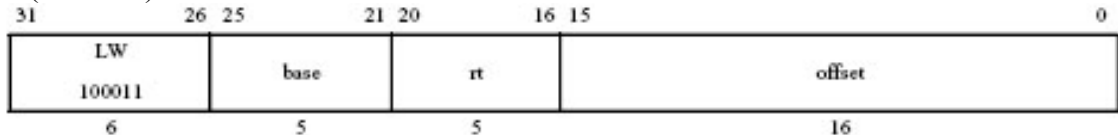
- add (R-format)



- addi (I-format)



- lw (I-format)



### Compiling

- Going from a high-level language representation of a program to an equivalent sequence of assembler instructions