Pico Assembly Language (PAL)

(adapted from Wakerly)

CS 3210 – Spring 2018 – B. Cohen

Typical Three-Address/Operand Instruction Set

Instruction	*Meaning*	*Notes*
`COPY s, d`	`s → d`	`copy`
`MOVE v, d`	`v → d`	`move`
`ADD s1, s2, d`	`s1 + s2 → d`	
`INC s`	`s + 1 → s`	`increment`
`SUB s1, s2, d`	`s1 – s2 → d`	
`DEC s`	`s – 1 → s`	`decrement`
`MUL s1, s2, d`	`s1 * s2 → d`	
`DIV s1, s2, d`	`s1 / s2 → d`	`I nteger division`
`BEQ s1, s2, addr`	`if s1 == s2 branch to address addr`	`branch if equal`
`BGT s1, s2, addr`	`if s1 > s2 branch to address addr`	`branch if greater than`
`BR addr`	`branch to address addr`	`unconditional branch (goto)`
`END`	`halts program`	

---

notes

- s, s1, and s2 are source locations; d is a destination location; addr is a memory location (label)
  - sources and destinations may be a named memory location or a register
  - there are 8 registers, named R0 through R7
  - memory identifiers are composed of letters only, max length of 5
- v is an immediate value
  - all numeric data is of type unsigned integer, represented in octal
- labels (on a line of code) are terminated by a colon
- note that comparisons are part of the branch instructions
- one instruction per line
- white space will be spaces only (no tabs)
  - commas in operand lists may be followed by zero or more spaces
  - spaces may appear after the end of the instruction
  - colons in labels may be followed by zero or more spaces
  - instructions may be followed by zero or more spaces
- comments begin with a semi-colon and end with EOL; they may be on a line alone, or following code on a line