26 March 2001

Important announcements

- Upcoming Quiz – 4 April
- Homework
- Labs

Stacks

- For storing temporary values in expression evaluation

```
x = (a+b) * (c+d) ;
```

- For storing activation records for recursive computations

```
int fact(int x) {
    return x<=1 ? 1 : x*fact(x-1) ;
}
```

History

- *Polish notation*
  - Invented by Jan Lukasiewicz (1878-1956)
  - Eliminates parentheses in mathematical expressions
    - (a+b) * (c+d) ⇒ * + a b + c d
  - Also, called *prefix* notation
    - `times(plus(a,b),plus(c,d))`
- Reverse Polish notation (RPN)
  - Puts the operators after the operands
    - (a+b) * (c+d) ⇒ a b + c d + *

Computational examples of RPN

- Popularized in HP RPN calculators
  - Operands held in a *stack*
  - 17 `[enter]` 23 `[enter]` + 43 `[enter]` 57 `[enter]` + *
  - Widely done in Java
    - CoCalc RPN Scientific Calculator
- Stack computers
  - Intel x86 stack pointer stored in register SP (or ESP)
    - SP grows downward in memory
    - Push bytes, words, double words, all registers
  - Primary use of stack is to store activation records
    - Stack overflow is a serious Internet security problem!

Stack based programming languages

- Forth
  - Invented by Charles Moore in early 60's
  - Used to control many real-time devices (FedEx wand)
  - Open Firmware standard used to boot devices
- Postscript
  - Invented by Adobe
  - Used in many laser printers
  - How is this document generated?
    - MS Word ⇒ PostScript ⇒ PDF
  - How about the trees at the right?
    - Pure PostScript

Stack on the LC-2

R6 points to top value on the stack

- It's a little messy when the stack is empty

LC-2 stack grows upward

- Most stacks grow downward

Simple PUSH operation

```
PUSH    ADD     R6,R6,#1
        STR     R0,R6,#0
        RET
```

Simple POP operation

```
POP     LDR     R6,R6,#0
        ADD     R6,R6,#-1
        RET
```

Robust implementations check for stack underflow and overflow

Special arithmetic operations can use the stack

```
OpAdd   ADD     R6,R6,#-1
        LDR     R0,R6,#0
        LDR     R1,R6,#1
        ADD     R0,R0,R1
        ST      R0,R6,#0
        RET
```

Try an example (or two)

```
x = (a+b) * (c+d) ;
x = a*(b+c) + d*(b-c) ;
```