CSCI 361 Week 12 Notes

Kyle Krstulich

April 15, 2025

Virtual Machine part 2

We will be implementing the quadratic formula

```
x_1 = (-b + sqrt(pow(,b2) - 4ac)) / 2a
x_2 = (-b - sqrt(pow(,b2) - 4ac)) / 2a
```

Functions can be a problem, you need to jump to subroutine code. This is an integer only computer, so rational numbers will be a problem. We need a mechanism to context switch between functions. The function doesn't know anything about the caller.
We need to save the state of the caller.

- **Local**: Stores local variable ID callers environment(function). Take the local memory segment and save it somewhere. When you jump to functions you jump to a different local memory segment. From caller local to callee local.
- **Argument**: Every function has arguments. The argument segment will store these arguments. Caller puts values in argument segment, a deep level of recursion can overflow the stack by filling it with argument segments.

  To pass arguments to subroutine: push stack. To return pop stack.
- **THIS**: Maintains state of stack.
- **THAT**: Maintains state of stack.

Homework

New VM keywords to be implemented in chapter 8.

- Flow control: goto label, if-goto label
- Function Use: call f m, function f m, return
  - f is just a label, and address in RAM.
  - m is the number of arguments that you pass into the function.
  - n is start of function in VM, n is the number of local variables.
  - return just makes top of stack the result.