CichoIVM

Florian Mayer
Jonas Windhager

Source language

- **Subset of Python** 3
- Limited to integer arithmetic
- Only functions (no OO)
- No global variables
- No standard library
- Special I/O functions

For example

```
def fact(n):
    f = 1
    while n > 0:
        f = f * n
        n = n - 1
    return f
def main():
    n = read()
    prnt(fact(n))
    return 0
```

Compiler

- Implemented in Python 3
- Uses built-in AST module
- **Two passes** to resolve forward references
  - Generate intermediate representation
  - Generate byte-code

Virtual machine

- **Register-machine**
- Implemented in C++11
- Virtual registers allocated at runtime
- Only 64-bit integer arithmetic

VM instructions

- Arithmetic: add, sub, …
- Control flow: conditional jump, unconditional jump, function call
- Comparisons
- Allocate virtual registers
- Simple I/O

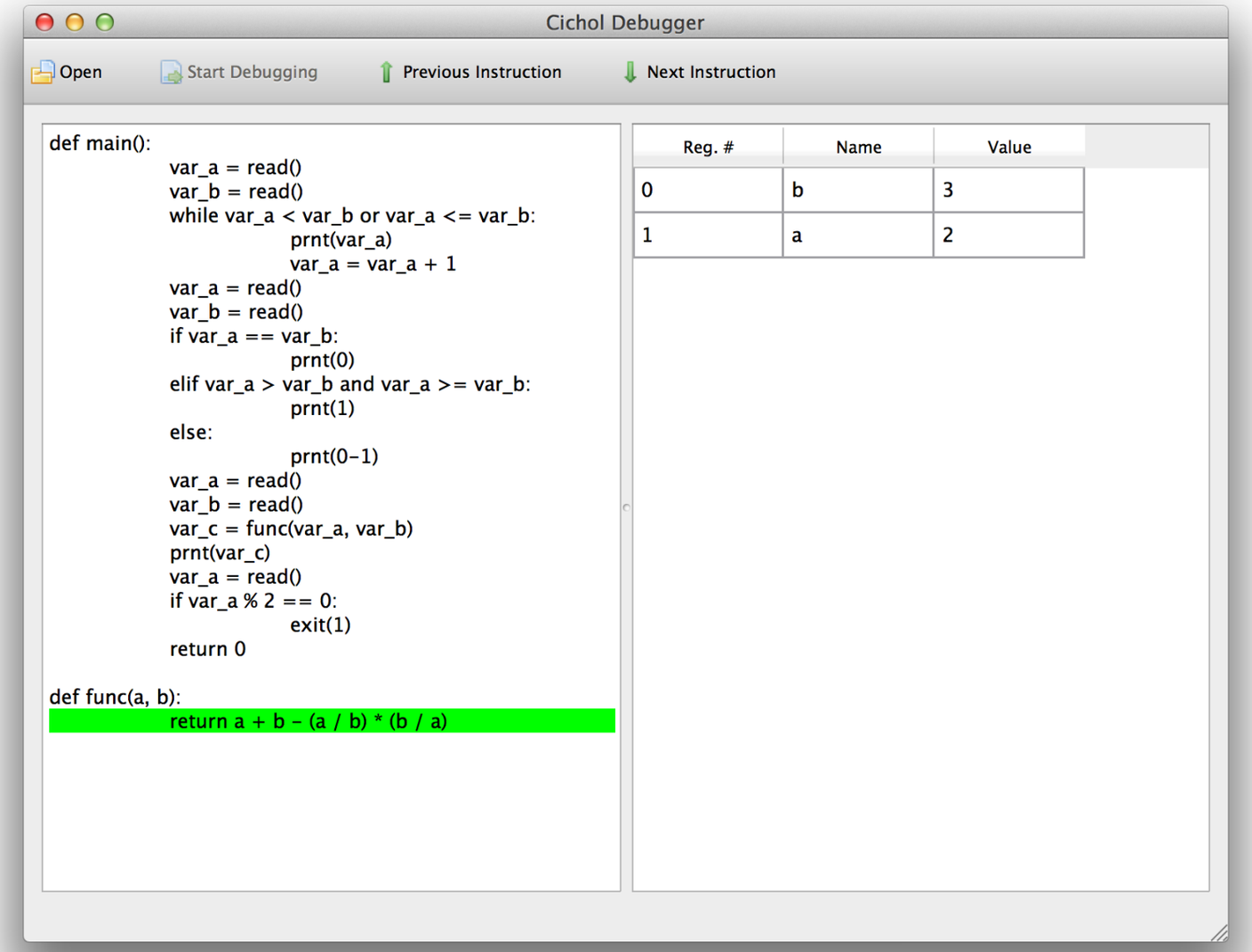
Post-mortem debugging

- Debugging after the program has terminated
- Inspect state of crashed program
- CicholVM writes out **inverse instructions**
- VM state saved on exit

CicholGUI

- Graphical user-interface for **reverse execution**

- written in C++ and Qt



Benchmark

- Calculate **fact(25)** 1000 times
- ~300 times slower