Bluebird (wrap-up)

Cardinal

C Calling Conventions

Bluebird : binary representation is a mapping to/from binary machine values and values in our lang.

Bluebird	machine
0	00 ···· 000
false	00 ··· 001
1	00 ···· 010
none	00 ··· 011
2	00 ···· 100

tag bit

lecture representation

and	0	1
0	0	0
1	0	1

clearing bits

or	0	1
0	0	1
1	1	1

setting bits

xor	0	1
0	0	1
1	1	0

flipping bits

	b_{63}	b_{62}	b_{61}	·····	b_2	b_1	b_0
AND	0	0	0		0	1	1
	0	0	0		0	b_1	b_0

Bluebird

is int (false) ⟹ false

is int (true) ⟹ false

is int (3) ⟹ true

for lecture:

R(false) = 000 ···· 00000001

R(true) = 000 ···· 10000001

isint (e) → mov rax, ···· → YOUR ASM

write logic for isint without using cmp, jmp, ...

instructions	true	false	3
	···· 010000001	···· 000000001	···· 000000110
and rax, 1	···· 000000001	···· 000000001	···· 000000000
shl rax, 7	···· 010000000	···· 010000000	···· 000000000
or rax, 1	···· 010000001	···· 010000001	···· 000000001
xor rax, 128	···· 000000001	···· 000000001	···· 010000001
	false	false	true

BAD IDEA

Bluebird	machine
0	00 ···· 000
2	00 ··· 001
1	00 ···· 010
4	00 ··· 011
3	00 ···· 100

Cardinal

1 Syntax

⟨expr⟩ ::= ... | print(⟨expr⟩)

2 Semantics

code	Bluebird →	Cardinal ←	
1+3	4	4	
true || false	true	true	
1 + true	undefined	exit code 1	
print(4) + 2	N/A	6	(also prints 4)
print(print(3))		3	(also prints 3 twice)
isInt(print(4))		true	(also prints 4)
print(2) + print(3)		5	(prints 2 and 3 in that order)

to do this: call C function

In Cardinal: runtime errors are handled by calling exit (a C function)

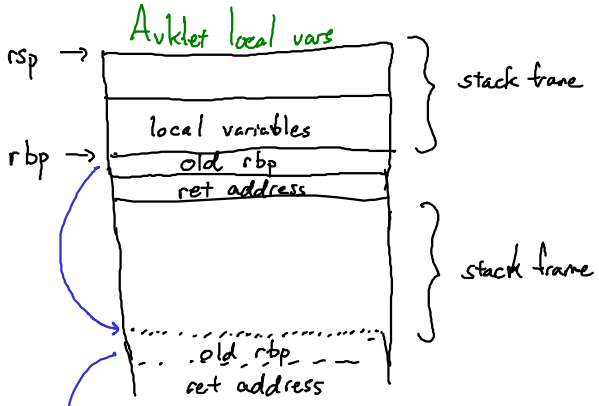
C Calling Conventions

rip – instruction pointer

Cardinal has to be a good C caller and callee

Chronologically: x86-64

1. Caller sets up and calls
   a. push caller-saved regs to stack
   b. assign arguments to parameter locations
      - rdi
      - rsi
      - rdx
      - rcx
      - r8
      - r9
      - on top of stack in reverse order
   c. calls function (call instruction)
      - pushes rip and jmp label
   ⋮
5. Restore caller-saved regs



volatile (caller-saved)
might change

stable (callee-saved)