

Eagle

63-bit signed integers
 booleans
 tuples

5
 true
 (1, 2)

floats primitive arrays

pointers/references mechanic

functions !!

strings
 dictionaries
 lists
 sets
 data structures

4K "pages"



0xffffffffffffffff

In Eagle:

- driver will allocate a slab of memory
- driver will pass a pointer into bird_main
- bird_main stores pointer as "heap_cursor"
- need memory? move heap_cursor [↑] next free byte

We will not

- allocate just enough memory
- recognize out-of-memory conditions
- expect or allow programmer to manage memory
- clean up anything

Not Eagle

Syntax

```

<expr> ::= ...
| (<expr>, <expr>)
| fst <expr>
| snd <expr>
| ispair(<expr>)

```

Semantics

← make a pair holding the result of these two exprs
 ← get first element from a pair

$(2, 3) \implies (2, 3)$

$(1+1, 5+5) \implies (2, 10)$

$\text{fst}(\text{true}, \text{false}) \implies \text{true}$

$\text{snd}(\text{true}, 5) \implies 5$

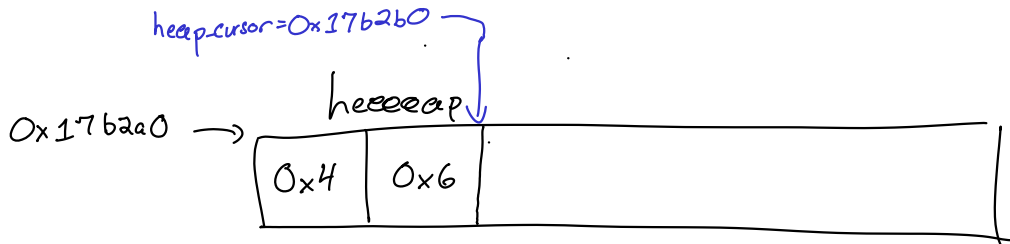
$\text{fst}(\text{true}, 0+\text{false}) \implies \text{H}$

Expect errors:
fst/snd on a non-pair

Binary Representation

	Machine form	Bird form
integers	$0x\text{NNNNNNNNNNNNNNNNNNNN}[n\text{nn}0]$	$0x\text{NNNNNNNNNNNNNNNNNNNN}[nnn]$
booleans	$0x[\text{b}111]\text{FFFFFF FFFFFFFF}$	true/false
pointers (to pairs)	$0x\text{NNNNNNNNNNNNNNNNNNNN}[n001]$	$0x\text{NNNNNNNNNNNNNNNNNNNN}[n000]$

$(2, 3)$



$\text{rax} = 0x17b2a1$

```

.section data
align 8
heap-cursor:
dq 0

```

.section text

bird_main:

push rbp

⋮

mov r10, [heap-cursor]

b = byte
 w = "word" (16 bits)
 d = double word
 q = quad word