

- Static scoping vs. dynamic scoping
lexical

↓
Exceptions (in Java)
behave like dynamic
scoping

- Pure functions: No side effects
On the same input, always
produces the same output

```
int x = 3  
int foo(int y) {  
    x = x + 1  
    return x + y  
}
```

Imperative Programming in OCaml

① Ref cells

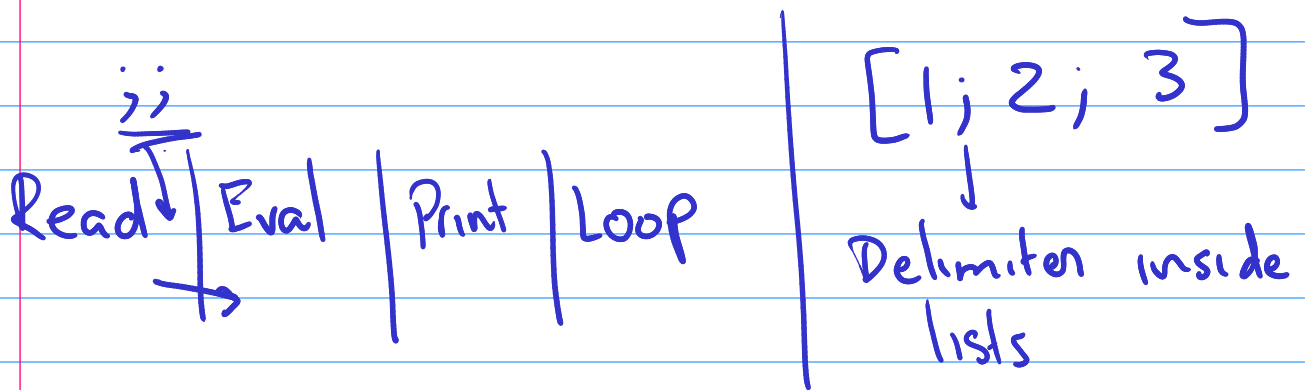
let x = ref 3

!x

x := 3

x := !x + 2

② Semicolons



Sequencing construct

$e_1 ; e_2$:— Finish evaluating e_1
Throw its value away
Now evaluate e_2
Return its value.

③ Loops!

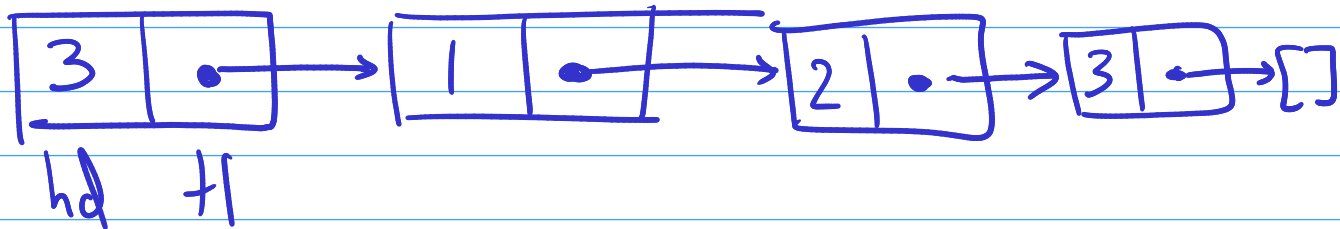
while e do
 e'

done

for x = b
do
 down to

done

- Back to lists



Challenge: Build a doubly-linked list

