

Num


num	2	num	3
-----	---	-----	---

Lists

link	$l_1$	$l_2$
link	$l_1$	$l_2$
link	$l_1$	$l_2$

Point

Point	4	5	7
Point	6	7	

 = don't  
 need to  
 store tags,  
 either  
 (fixed-size data)

B;BOP

Don't stop  
 the  
 B;BOP

x = 123478

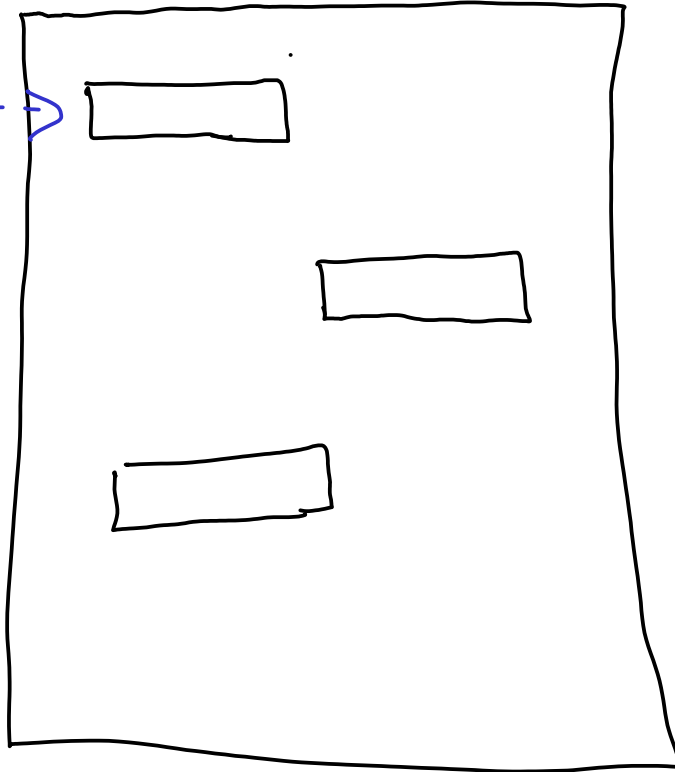
y = 44

z = 456789

Can't be pointers  
(too small,  
odd)

"Conservative"

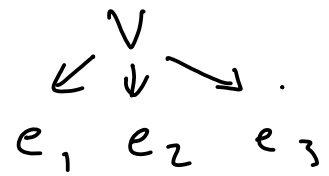
GC



# TRUTH and PROVABILITY

```
(let (x 5)  
  (if false x 2))
```

# CONTROL



if when cases switch

for/while break continue

try/catch/finally throw/raise with (Python)

||  
^ goto

yield (Python)

return

j sequence

end exit

```

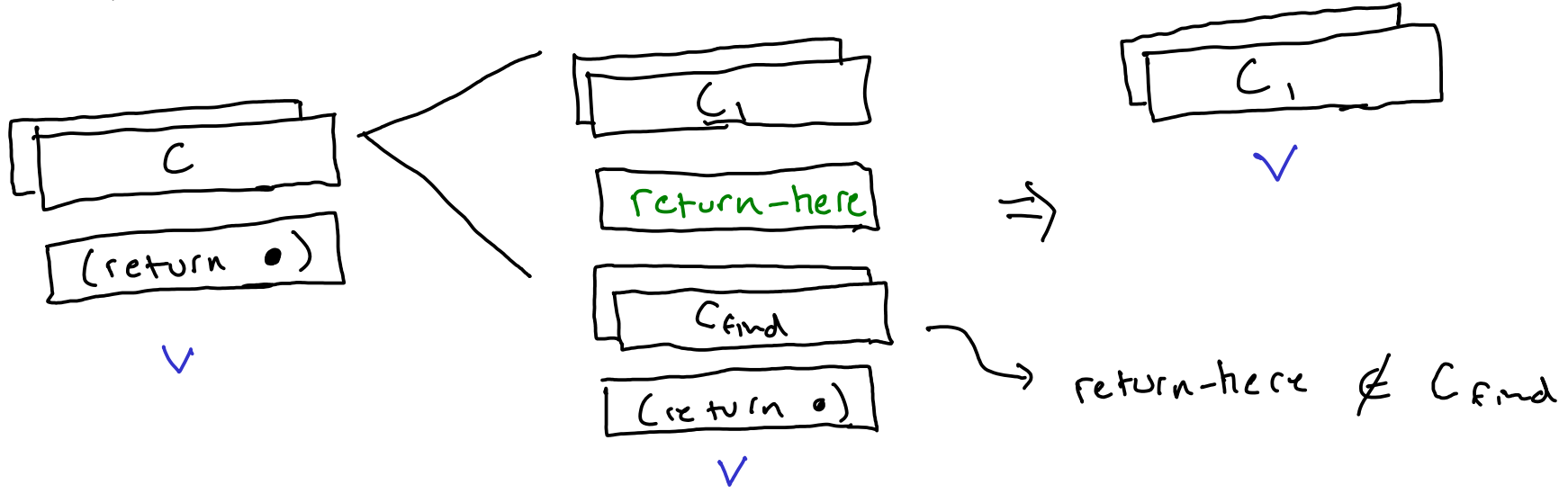
def find(lst, to-find):
    for elt in lst:
        if elt == to-find:
            return true
    return false

```

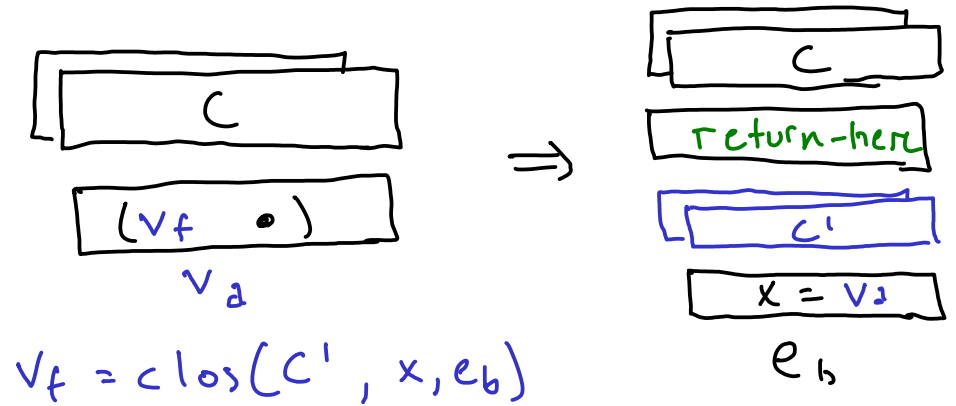
```

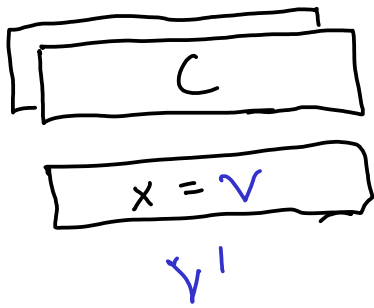
if find([1,2,3], 2):
    print("found")

```

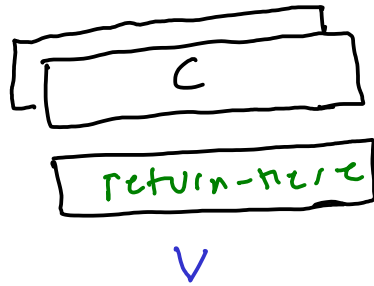
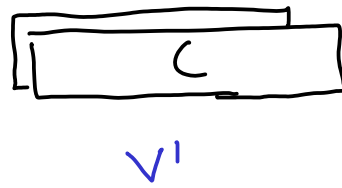


ABOVE, RULE  
THREW OUT  
SOME CONTEXT  
(`Cfind`, `return-here`)





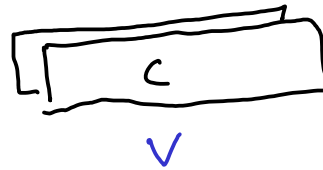
⇒



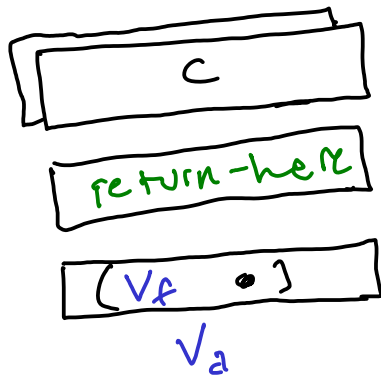
⇒



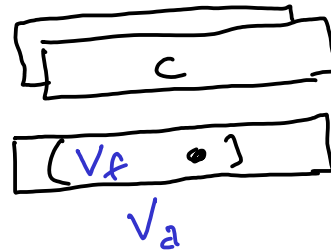
(Python)



(Ruby)



⇒



Is this OK  
(relevant for tail calls)

