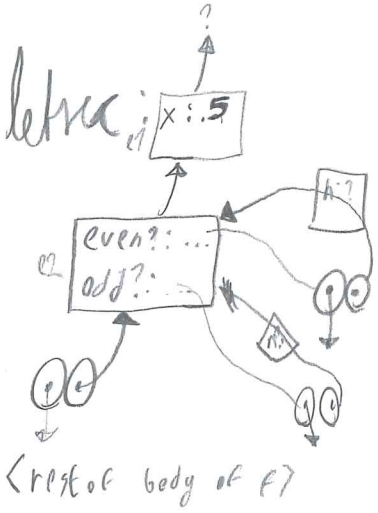
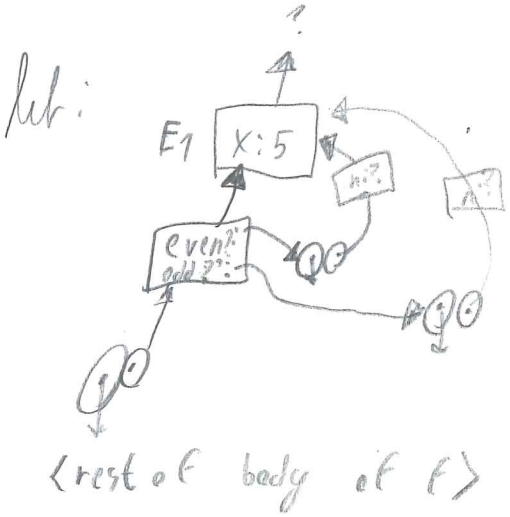
4.20.b!

```
(define (f x)
  (letrec
    ((even? (lambda (n)
              (if (= n 0) #t (odd? (- n 1)))))
     (odd? (lambda (n)
             (if (= n 0) #f (even? (- n 1))))))
    <rest of body of f>))
```





So using let is like

```
(let ((<v1> <e1>) ... (<vn> <en>))
  (b))
```

↓

```
((λ (<v1> ... <vn>)
   (b))
 <e1> ... <en>)
```

Which clearly does show how <e1> ... <en> will be evaluated in the "outside" scope.