

Welcome to [DrRacket](#), version 8.11 [cs].

Language: **racket**, with **debugging**; memory limit: **128 MB**.

```
> (top-level-eval '(let ([x (call/cc (lambda (k)
  k))])
  (x (lambda (i) 7))))
```

```
>(extend-env
  '(k)
  '((acontinuation
    (eval-cont
      ())
    (let-cont
      (x)
      ((app-exp ((var-exp x) (lambda-exp (i)
        ((lit-exp 7))))))
      (empty-env-record)
      (halt-cont))
    (empty-env-record))))
  '(empty-env-record))
<'(extended-env-record
  (k)
  #((acontinuation
    (eval-cont
      ())
    (let-cont
      (x)
      ((app-exp ((var-exp x) (lambda-exp (i)
```

```

((lit-exp 7))))))
      (empty-env-record)
      (halt-cont))
      (empty-env-record))))
      (empty-env-record))
>(extend-env
  '(x)
  '((acontinuation
    (eval-cont
     ())
    (let-cont
     (x)
     ((app-exp ((var-exp x) (lambda-exp (i)
((lit-exp 7))))))
      (empty-env-record)
      (halt-cont))
      (empty-env-record))))
  '(empty-env-record))
<'(extended-env-record
  (x)
  #((acontinuation
    (eval-cont
     ())
    (let-cont
     (x)
     ((app-exp ((var-exp x) (lambda-exp (i)
((lit-exp 7))))))
      (empty-env-record)
      (halt-cont))
      (empty-env-record))))
  (empty-env-record))
>(apply-proc
  '(acontinuation

```

```

    (eval-cont
      ()
      (let-cont
        (x)
        ((app-exp ((var-exp x) (lambda-exp (i)
((lit-exp 7))))))
        (empty-env-record)
        (halt-cont))
      (empty-env-record)))
'((closure
  (i)
  ((lit-exp 7))
  (extended-env-record
    (x)
    #((acontinuation
      (eval-cont
        ()
        (let-cont
          (x)
          ((app-exp ((var-exp x) (lambda-exp (i)
((lit-exp 7))))))
          (empty-env-record)
          (halt-cont))
        (empty-env-record))))
      (empty-env-record))))
  (halt-cont))
> (extend-env
  '(x)
  '((closure
    (i)
    ((lit-exp 7))
    (extended-env-record
      (x)

```

```

        #((acontinuation
          (eval-cont
            ()
            (let-cont
              (x)
              ((app-exp ((var-exp x) (lambda-exp (i)
((lit-exp 7))))))
              (empty-env-record)
              (halt-cont))
              (empty-env-record))))
          (empty-env-record)))
      '(empty-env-record))
< '(extended-env-record
  (x)
  #((closure
    (i)
    ((lit-exp 7))
    (extended-env-record
      (x)
      #((acontinuation
        (eval-cont
          ()
          (let-cont
            (x)
            ((app-exp ((var-exp x) (lambda-exp (i)
((lit-exp 7))))))
            (empty-env-record)
            (halt-cont))
            (empty-env-record))))
        (empty-env-record))))
    (empty-env-record)))
  (empty-env-record))
>(apply-proc
  '(closure

```

```

(i)
((lit-exp 7))
(extended-env-record
 (x)
 #((acontinuation
      (eval-cont
        ())
      (let-cont
        (x)
        ((app-exp ((var-exp x) (lambda-exp (i)
((lit-exp 7))))))
      (empty-env-record)
      (halt-cont))
      (empty-env-record))))
 (empty-env-record)))
'((closure
  (i)
  ((lit-exp 7))
  (extended-env-record
   (x)
   #((closure
        (i)
        ((lit-exp 7))
        (extended-env-record
         (x)
         #((acontinuation
              (eval-cont
                ())
            (let-cont
              (x)
              ((app-exp ((var-exp x) (lambda-exp
(i) ((lit-exp 7))))))
            (empty-env-record)

```



```

        ()
        (let-cont
          (x)
          ((app-exp ((var-exp x) (lambda-exp (i)
((lit-exp 7))))))
          (empty-env-record)
          (halt-cont))
          (empty-env-record))))
      (empty-env-record)))
< '(extended-env-record
  (i)
  #((closure
    (i)
    ((lit-exp 7))
    (extended-env-record
      (x)
      #((closure
        (i)
        ((lit-exp 7))
        (extended-env-record
          (x)
          #((acontinuation
            (eval-cont
              ()
              (let-cont
                (x)
                ((app-exp ((var-exp x) (lambda-exp
(i) ((lit-exp 7))))))
                (empty-env-record)
                (halt-cont))
                (empty-env-record))))))
          (empty-env-record))))))
    (empty-env-record))))))
  (empty-env-record))))

```

```
(extended-env-record
  (x)
  #((acontinuation
    (eval-cont
      ())
    (let-cont
      (x)
      ((app-exp ((var-exp x) (lambda-exp (i)
((lit-exp 7))))))
      (empty-env-record)
      (halt-cont))
      (empty-env-record))))
  (empty-env-record)))
```

<7

7

>