(define-datatype proc-val proc-val?

[prim-proc

(name symbol?)]

[closure

(ids (lambda (x) (or (var-type? x)

((my-list-of var-type?) x)

((my-list-of symbol?) x))))

(bodies (list-of expression?))

(env environment?)]

**[continuation-proc**

**(k continuation?)**])

(define apply-prim-proc

(lambda (prim-proc args k)

(case prim-proc

[(+) (apply-k k (+ (1st args) (apply + (cdr args))))]

[(-) (apply-k k (- (1st args) (2nd args)))]

**[(call/cc) (apply-proc (car args)**

**(list (continuation-proc k))**

**k)]**

[else (error 'apply-prim-proc

"Bad primitive procedure name: ~s"

prim-proc)])))

(define apply-proc

(lambda (proc-value args k)

(cases proc-val proc-value

[prim-proc (op) … ]

[closure (ids bodies env) …]

**[continuation-proc (k) (apply-k k (car args))]**

[else (error 'apply-proc

"Attempt to apply bad procedure: ~s"

proc-value)])))