Problems

1. **Lambda** - constructor/accessors

   (define (lambda? x)

   (define (make-lambda params body)

   (define lambda-params
   (define lambda-body

   The value of a lambda is a double bubble. Thus, a double-bubble abstraction:

   (define (make-procedure params body env)

   (define proc-params
   (define proc-body
   (define proc-env

   Finally:

   (define (eval-lambda exp)


2. **Combination**

   (define (eval-combination exp env)
Some environment stuff:
the-empty-environment - the “root” of the environment chain
(extend-environment vars vals enclosing-env) - drop a frame

(define (mapply proc args)

---

Extending the Evaluator