Can we assume that the function definition will be of the form

(define (id ...) ...) 

and not 

(define id (lambda (...) ...) )

?

Also, just to clarify, any internal function definition will be of the form 

(define square (lambda (x) (* x x)))

and not 

(define (square x) (* x x))

right?

For your first my understanding is that you will always have a function of the form 

(define (name args) 
  ; defines
Also the result should be

(define (name args)
  ((lambda ( ... ) execution) a1 a2 a3 ...)
)
That's what the spec says.

New question for problem 4: If a defined value is never used, do we need to create a lambda for it? For example, the function (define (test x) (define y (+ 1 x)) (+ x x)) just returns (+ x x) and never uses y. I wrote my code assuming that we could eliminate that local define by just excluding it, but the test case seems to want a lambda over y that would return (+ x x) for all inputs.

Never mind, I just changed how I was implementing it to match the test cases.