Week 1, Monday 8/28/17

Book: Essentials of programing languages

Weekly bible essays due Fridays

First programming assignment due Wednesday

Writing in Scheme, only use what you’ve learned in class

Scheme interpreter needs to be downloaded, is on blackboard under content

Read section 1 and Exercise 1.1

Scheme Primitive Functions

1. List based language
   1. ‘()🡪empty list
      1. Example: ‘(1 2 3)

Functions

1. Car—returns the first element in a list
   1. (car ‘(123))🡪 returns 1
   2. (car ‘((1)(2)(3)))🡪 returns list ‘(1)
   3. (car(car ‘((1)(2)(3))))🡪 returns 1
      1. Shortcut: (ca**a**r ‘((1)(2)(3)))🡪 returns 1
2. Cdr—returns the rest of the list minus the first element
   1. (cdr’(123))🡪 returns list ‘(23)
   2. (cddr’(123))🡪 returns list ‘(3)
3. Cons—constructs lists where parameters are an element and a list, used to construct a list
   1. (cons 1(cons 2’()))🡪 returns lists ‘(12)
4. Define—defines a variable name with a corresponding value
   1. (define a 5) sets variable “a” equal to 5
   2. (define b ‘(123))
   3. (define c(lambda(x)(+x1)))
      1. Lambda—an un-named function, and defines functions

Week 1, Wednesday 8/30/17

Scheme interpreter: Racket (download off of blackboard)

-real time interpreter

Function ‘define’

Example: (define lst’(123))🡪remembers list ‘lst’

Predicate—A fancy way of saying a function that returns a Boolean

If—(if bool\_expression\_true\_expression\_false\_expression)