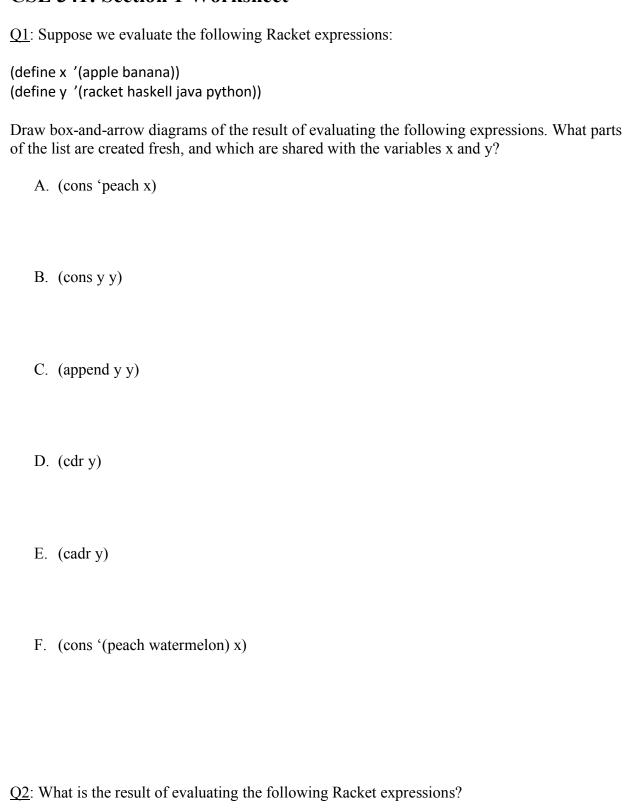
CSE 341: Section 1 Worksheet

(a)



```
(let ([x 100]
     [y 5]
 (let([x 1])
  (+xy)))
(b)
(let ([x 5]
     [y 4]
 (let ([z 5]
      [y (+ x 2)]
  (let([x(+zy)])
   (+ x y z))))
(c)
(let* ([x 100]
      [y (*2 x)]
      [x 50]
      [z (+ y x)])
 (+yz)
(d) Note: (list xy ...) creates a list of the form '(xy ...)
(define bob (lambda (x) (list x)))
(define jane "Hamburger")
(let ([cathy (list jane jane)]
     [hank (bob jane)]
     [jane "Hotdog"])
     (let* ([bob (lambda (x) (list (car x) jane))]
            [jane (bob hank)])
               (cons (car cathy) (bob jane))))
```

Q3: Define a function, numOdds, that returns the number of odds within a list of numbers. (Challenge: Can you make this tail recursive?)