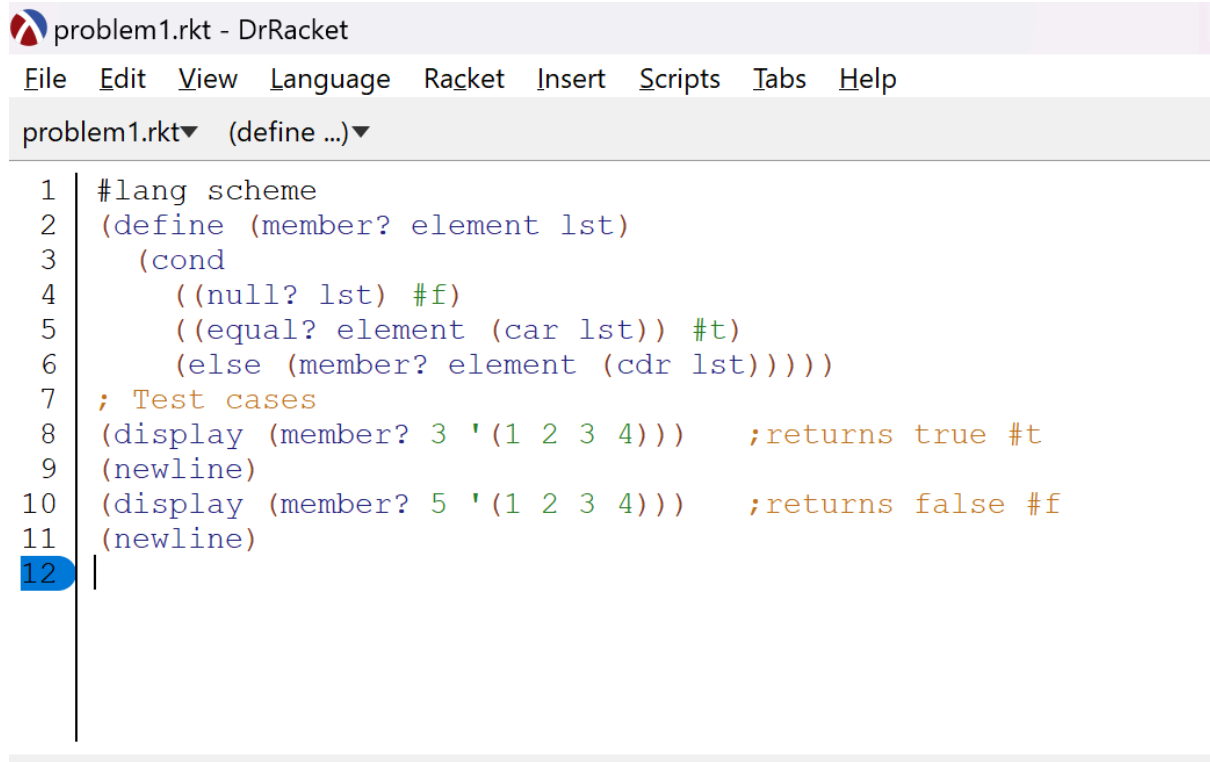


Classwork 13

Problem1)



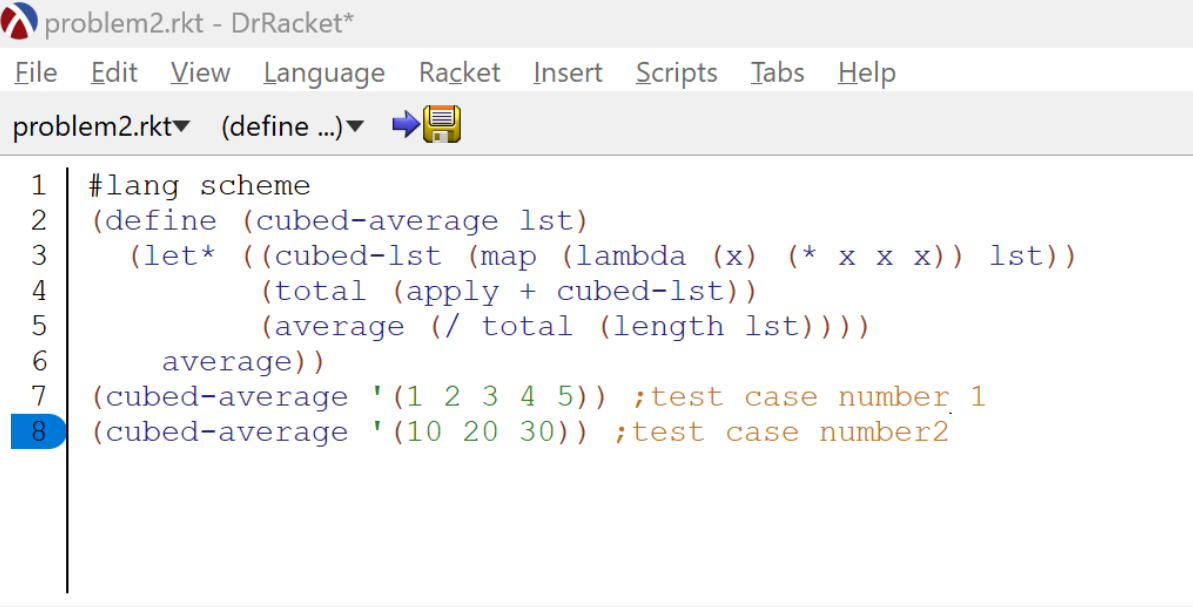
The image shows a DrRacket window titled "problem1.rkt - DrRacket". The menu bar includes File, Edit, View, Language, Racket, Insert, Scripts, Tabs, and Help. The toolbar shows "problem1.rkt" and "(define ...)". The code editor contains the following Scheme code:

```
1 | #lang scheme
2 | (define (member? element lst)
3 |   (cond
4 |     ((null? lst) #f)
5 |     ((equal? element (car lst)) #t)
6 |     (else (member? element (cdr lst)))))
7 | ; Test cases
8 | (display (member? 3 '(1 2 3 4))) ;returns true #t
9 | (newline)
10 | (display (member? 5 '(1 2 3 4))) ;returns false #f
11 | (newline)
12 |
```

Welcome to [DrRacket](#), version 8.14 [cs].
Language: **scheme**, with **debugging**; memory limit: 128 MB.

```
#t
#f
>
```

Problem 2)



The image shows the DrRacket environment. The title bar reads "problem2.rkt - DrRacket*". The menu bar includes File, Edit, View, Language, Racket, Insert, Scripts, Tabs, and Help. The toolbar shows icons for opening a file, saving, and running. The editor window contains the following Scheme code:


```
1 | #lang scheme
2 | (define (cubed-average lst)
3 |   (let* ((cubed-lst (map (lambda (x) (* x x x)) lst))
4 |         (total (apply + cubed-lst))
5 |         (average (/ total (length lst))))
6 |     average))
7 | (cubed-average '(1 2 3 4 5)) ;test case number 1
8 | (cubed-average '(10 20 30)) ;test case number2
```

Below the editor, the DrRacket welcome message is displayed: "Welcome to [DrRacket](#), version 8.14 [cs]. Language: **scheme**, with **debugging**; memory limit: 128 MB." The output area shows the results of the first two test cases: "45" and "12000", followed by a prompt ">".

Problem 3)

problem3.rkt - DrRacket*

File Edit View Language Racket Insert Scripts Tabs Help

problem3.rkt (define ...) 

```
1 | #lang scheme
2 | (define (find-max lst)
3 |   (if (null? lst)
4 |       #f
5 |       (foldl (lambda (x cur-max)
6 |               (if (> x cur-max)
7 |                   x
8 |                   cur-max))
9 |               (car lst)
10 |              (cdr lst))))
11 |
12 | (find-max '(20 39 45 300 67 100))
13 | (find-max '(2000 1000 30000 4040332))
14 | (find-max '())
```

Welcome to [DrRacket](#), version 8.14 [cs].
Language: **scheme**, with **debugging**; memory limit: 128 MB.
300
4040332
#f
>