

King Fahd University of Petroleum & Minerals  
Information & Computer Science Department

ICS 410 Programming Languages

Assignment 02

---

| <u>Due Date</u>             | <u>Weight</u> | <u>Semester</u> |
|-----------------------------|---------------|-----------------|
| 11 <sup>th</sup> April 2007 | 5%            | 062             |

**Problem 1:**

**(5 points).**

Define a Scheme function `squareperimeter` which receives, as a parameter, the area of a square `S` and returns the perimeter of `S`.

**Example:**

```
> (squarePerimeter 25)
20
```

```
> (squarePerimeter 16)
16
```

---

**Problem 2:**

**(5 points).**

Define a Scheme function `numberoftiles` that receives, as parameters, the length and width of a floor and the edge length of a square tile (*all in centimeters*). It returns the *whole* number of tiles needed to cover the floor *completely*. Hint: you may need to use the function `ceiling`.

**Examples:**

```
> (numberoftiles 200 200 20)
100
```

```
> (numberoftiles 400 300 30)
134
```

---

**Problem 3:**

**(5 points).**

Define a Scheme function `zeros` that receives, as a parameter, a simple list of numbers and it returns the number of zeros in that list.

**Examples:**

```
> (zeros '(0 4 6 0 7 0 9))
3
```

```
> (zeros '(1 5 6 9 3 2 1))
0
```

---

**Problem 4:****(5 points).**

Define a Scheme function **simpledelete** that receives, as parameters, a simple list of atoms and an atom and it returns the list after deleting all the occurrences of that atom.

**Examples:**

```
>(simpledelete '(a b c a b d e) 'a)
(b c b d e)

>(simpledelete '(a b c a b d e) 'f)
(a b c a b d e)
```

---

**Problem 5:****(10 points).**

Define a Scheme function **delete** that receives, as parameters, a list that can contains lists or atoms and an atom and it returns the list after deleting all the occurrences of that atom in any level of the list.

**Examples:**

```
>(delete '(a b (a c) (b a d) e) 'a)
(b (c) (b d) e)

>(delete '(a b c (a b) d e) 'f)
(a b c (a b) d e)
```

---

**Problem 6:****(10 points).**

Define a Scheme function **smallest** that receives, as a parameter, a simple list of numbers and it returns a list that contains the smallest numbers in the list.

**Examples:**

```
>(smallest '(23 45 79 12 9))
9
```

---

**Problem 7:****(10 points).**

Define a function **bubbleup** that takes a simple list of numbers as argument. This function moves the maximal number to the end of the list without changing relative order of other number elements in this list.

**Examples:**

```
>(bubbleup '(2))
(2)

>(bubbleup '())
nil

>(bubbleup '(2 5 1 3))
(2 1 3 5)

>(bubbleup '(8 6 5))
(6 5 8)
```

## Submission Requirements

1. Submit a CD (or a floppy disk) containing 7 Dr. Scheme files (*squareperimeter.scm*, *numbroftiles.scm*, *zeros.scm*, *simpledelete.scm*, *delete.scm*, *largestsmallest.scm* and *sort.scm*) containing the Scheme code for the seven given functions. Make sure that your CD/floppy is virus-free and has your name and ID # written on it.
2. Submit a printed report in MS Word that includes the following:
  - Course title, number, and section number
  - Your Name and ID number
  - The statement of the problem (The text in the previous two pages.)
  - A list of all the function you developed, 1 through 7 with some description of how it should work. This will be like a documentation of your functions.

## Important Notice

1. No assignment will be accepted after the due date
2. Any students who cheat from any other students (even for one function) will get Zero (0) in the whole assignment.
3. Only working (or partially working) programs will be graded. Please do not submit a program that does not compile.
4. The assignment will be graded out of (50) points distributed as follows:
  - Problems (1) - (4): **5 points each**  
(3 for the correct code, 1 for the documentation, and 1 for the execution with test cases)
  - Problems (5) - (7): **10 points each**  
(7 for the correct code, 1 for the documentation, and 2 for the execution with test cases)