CS2104 - Programming Language Concepts

HOMEWORK 7

October 9, 2002

Guidelines

Please prepare your homework according to the following guidelines:

1. Please prepare your answers in MSWord (`.doc` file) or plain text (`.txt` file) format.
2. All answers should be placed in a single `.doc` file or `.txt` file.
3. You have to upload your homework before the deadline. **No late submission is allowed!**
4. You must upload your homework using the file name given below.
5. To upload your homework, follow the link **Workbin** from the course web-site: `http://www.comp.nus.edu.sg/~cs2104`
6. Upload your file into the **HW7** folder of workbin. For file transfer, please FTP only in binary mode (not in ASCII mode).
7. **In the "*Description*" of the file just input a single digit indicating your tutorial group number.**

Questions

Deadline:	Thu 10 Oct 2002, 11:59 PM (*i.e.* befor midnight) **The system might be busy just before the deadline.** **It is your responsibility to submit well ahead of deadline.**
File name:	<*NUSNet (Windows) user-name*>`.doc` (for example: `isc90000.doc`) OR <*NUSNet (Windows) user-name*>`.txt` (for example: `isc90000.txt`)
First two lines of file:	Your name (first line), Your matric (second line)

You can use SML syntax to write your functional programs. Use the tutorials posted in the course web-site.

QUESTION 1 (1 mark) Define a function *member*, which takes in two arguments k and x and returns true if element k appears in list x (and false otherwise).

QUESTION 2 (2 marks) Define a function *mergesort* which takes in a list of integers, and returns a sorted list (Look up any standard textbook such as the CS 1102 text to find out the mergesort algorithm).

QUESTION 3 (1 mark) We define a *predicate* as a function which returns true/false. Then, define a higher order function *filter*. The function *filter* takes in two arguments: the first argument is a predicate P and the second argument is a list L. It returns another list L', containing those elements in L, for which P is true. For example the following function application

filter((fn x => x > 10), [1,23,10,15])

should return [23,15], the list of elements in [1,23,10,15] which are greater than 10.