CS 2SC3 and SE 2S03 Fall 2008

Programming Exercise 1

Instructor: William M. Farmer

Revised: 3 October 2008

Files due: 2 and 9 October 2008

The purpose of this programming exercise is to learn how to (1) implement a specification of a simple function in OCaml and C and (2) test the implementation using a representative set of inputs.

Part A

Write an OCaml program that satisfies the requirements listed below. Put your program in a file named `prog-1.ml` and your log book in a file named `log.txt`. Put both of these files into a directory named `ex-1-a`. Using subversion, import this directory into your directory in the course subversion repository at

`https://websvn.mcmaster.ca/se2s03`

The details of how to do this will be explained in the tutorials. Your files must be submitted no later than **10:30 on Thursday, October 2, 2008.**

Part B

Write a C program that satisfies the requirements listed below. Put your program in a file named `prog-1.c` and your log book in a file named `log.txt`. Put both of these files into a directory named `ex-1-b`. Using subversion, import this directory into your directory in the course subversion repository. Your files must be submitted no later than **10:30 on Thursday, October 9, 2008.**

Program Requirements

1. The program implements the binary function

   $$\mathsf{bismuth} : \mathbf{R}, \mathbf{R} \to \mathbf{R}$$

   specified by the following table:

condition	$\mathsf{bismuth}(x, y) =$
$0 \leq \lvert x\rvert, \lvert y\rvert < 1$	$\sqrt[3]{\lvert x + y\rvert}$
$\lvert x\rvert = 1$ and $0 \leq \lvert y\rvert \leq 1$	2
$\lvert y\rvert = 1$ and $0 \leq \lvert x\rvert \leq 1$	2
$1 < \lvert x\rvert$ or $1 < \lvert y\rvert$	$x^2 + xy + y^2$

R is the set of real numbers.

2. **R** is represented by the set of double precision floating-point numbers.
3. The program tests the implementation of bismuth on a representative set of inputs.
4. When the program is executed, it prints out the the results of testing bismuth.
5. All the procedures defined in the program (other than the C function `main`) are side-effect free.

Notes:

1. Please put your name and MAC ID at the top of each of your files.
2. Your programs must be your own work.
3. Your programs must compile and execute correctly on either mills or moore to receive full marks.
4. Your program may contain more than one procedure.
5. Files submitted late will receive no marks.