

**CS 2SC3 and SE 2S03**  
**McMaster University, Fall 2009**  
**Assignment 1**

Instructor: William M. Farmer

Revised: 16 September 2009

**Files due: 2 October 2009**

## 1 Overview

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

## 2 Background

A *vector* is a mathematical entity that has direction and magnitude. A vector can be identified with a point in Euclidean space. A point in 2-dimensional Euclidean can be represented with *Cartesian coordinates* as a pair  $V = (a, b)$  of real numbers where  $a$  is the  $x$ -coordinate and  $b$  is the  $y$ -coordinate of the point, respectively. (A point in 2-dimensional Euclidean could also be represented in other ways such as with *polar coordinates*.)

## 3 Requirements

### 3.1 Program Requirements

Write an OCaml program that:

1. Includes a function named `direction` of type

`float * float -> string`

specified by the table below. Informally, `direction` computes the “direction” of a vector in 2-dimensional Euclidean space represented with Cartesian coordinates.

condition	direction $(x, y) =$
$x = 0$ and $y = 0$	"no direction"
$x = 0$ and $y < 0$	"south"
$x = 0$ and $y > 0$	"north"
$x < 0$ and $y = 0$	"west"
$x > 0$ and $y = 0$	"east"
$x < 0$ and $y < 0$	"southwest"
$x < 0$ and $y > 0$	"northwest"
$x > 0$ and $y < 0$	"southeast"
$x > 0$ and $y > 0$	"northeast"

2. The program tests the implementation of `direction` on a representative set of inputs.
3. When the program is executed, it prints out the results of testing `direction`. For example, the program could print something like

```
The test results for the function direction are:
  direction (2.3,0.) = "east", which is correct.
  direction (2.3,1.9) = "north", which is incorrect.
  .
  .
  .
```

(Note that the program should not ask the user for input.)

### 3.2 Submission Requirements

Put your program in a file named `prog1.ml`, and put a copy of your log book in a file named `log1.txt`. (Make sure that the files are named exactly as specified. Case matters!) Put your name and MAC ID at the top of each of these files. Create a directory named `assign1`. Put the files `prog1.ml` and `log1.txt` into this directory. Using subversion, import this directory into your directory in the course subversion repository at

<https://websvn.mcmaster.ca/se2s03>

Your files must be submitted no later than **10:30 a.m. on Friday, October 2, 2009**.

## 4 Marking Scheme

This assignment is worth 100 points allocated as follows:

1. **Objective** (checked automatically by software)

- (a) Program file is present \_\_\_\_\_/10 pts.
- (b) Program compiles \_\_\_\_\_/10 pts.
- (c) Program runs \_\_\_\_\_/10 pts.
- (d) Program prints test results \_\_\_\_\_/10 pts.
- (e) Program passes objective tests \_\_\_\_\_/20 pts.

2. **Subjective** (assessed by TAs)

- (a) Program satisfies the requirements \_\_\_\_\_/20 pts.
- (b) Choice of test inputs \_\_\_\_\_/10 pts.
- (c) Quality of print out of test results \_\_\_\_\_/10 pts.
- (d) Style (comments only)

3. **Penalties**

- (a) Missing or substandard log book \_\_\_\_\_/-10 pts.

Notes:

1. A program that is submitted late will receive 0 points.
2. Your program must compile and execute correctly on mills to receive full marks.
3. Your program must be your own work.