SE 2AA4 Winter 2007

Software Design Exercise 1

Instructor: William M. Farmer

Assigned:	12 January 2007
Files due:	26 January 2007
Lab report due:	2 February 2007
Revised:	14 January 2007

The purpose of this software design exercise is to write a C program that creates, uses, and tests a simple data structure that stores a single two-dimensional vector. The program will consist of three modules.

Background

A *vector* is a mathematical entity that has direction and magnitude. A two-dimensional vector is represented by a pair $V = (a, b)$ of real numbers where a and b are the x- and y-coordinates of V. Given a real number r and two vectors $V_1 = (a_1, b_1)$ and $V_2 = (a_2, b_2)$, the *scalar multiple* of r and V_1 is the vector $rV_1 = (ra_1, rb_1)$ and the *sum* of V_1 and V_2 is the vector $V_1 + V_2 = (a_1 + a_2, b_1 + b_2)$.

Step 1

Write a first module that creates a vector data structure. It should be a C file named `vector.c` that contains the following five procedures:

- A constructor named `make_vector` that takes two real numbers a and b as input and stores a representation of the vector (a, b).

- Two selectors named `x_coordinate` and `y_coordinate` that returns the x and y coordinates of the stored vector, respectively.

- A mutator named `mul` that takes a real number r as input and changes the stored vector V to the scalar multiple of r and V.

- A mutator named `sum` that takes two real numbers a and b as input and changes the stored vector V to the sum of V and the vector (a, b).

Step 2

Write a second module that contains several procedures derived from the procedures of the first module. It should be a C file named `derived.c` that imports the file `vector.c` and contains the following five procedures:

- A selector named `magnitude` that returns the length of the stored vector.

- A selector named `angle` that returns the angle from the x-axis to the stored vector measured in radians.

- A predicate named `orthogonal` that takes two real numbers a and b as input and returns 0 if the stored vector is not orthogonal to (a, b) and returns 1 if the stored vector is orthogonal to (a, b).

- Two mutators named `set_x` and `set_y` that take a real number r as input and changes the x-coordinate and the y-coordinate of the stored vector to r, respectively.

Step 3

Write a third module that tests the first and second modules together. It should be a C file named `test.c` that imports the files `vector.c` and `derived.c`. Record the results of using this module to test the procedures in the other two modules.

Step 4

Submit the three files `vector.c`, `derived.c`, and `test.c` using subversion (how to do this will be explained later). E-mail the `vector.c` file to your assigned partner (your partner will be posted on the course web site on Thursday, January 25, 2007.). Your partner will likewise e-mail his or her `vector.c` file to you. You must finish this step not later than midnight Friday, January 26, 2007.

Step 5

After you have received your partner's `vector.c` file, replace your `vector.c` file with your partner's. Do not make any modifications to any of the code. Run your test module and record the results.

Step 6

Write a report that includes the following:

1. Your name and MAC ID.

2. Your `vector.c`, `derived.c`, and `test` files.

3. Your partner's `vector.c` file.

4. The results of the test of your `vector.c` and `derived.c` files.

5. The results of the test of your partner's `vector.c` file and your `derived.c` file.

6. A discussion of the test results and what you learned doing the exercise. List any problems you found with (1) your program, (2) your partner's `vector.c` module, and (3) the specification of the three modules. Discuss how these problems could have been avoided.

7. A copy of the part of your log book relevant to this lab exercise.

The lab report is due no later than the beginning of the tutorial session on Friday, February 2.

Notes:

1. Please put your name and MAC ID at the top of each of your source files.

2. Your program must work in the ITB labs when compiled by gcc.

3. If your partner fails to provide you with a copy of his or her `vectors.c` file by the deadline, please tell the instructor via e-mail as soon as possible.