CS 2SC3 and SE 2S03

McMaster University, Fall 2009

Assignment 6

Instructor: William M. Farmer

Revised: 27 November 2009

Files due: 11 December 2009

1 Overview

The purpose of this programming assignment is to create in C a mutable data structure for storing vectors using linked lists and multiple code files.

2 Background

For the purposes of this assignment, let us define a *vector store* to be a data structure that holds a finite sequence of vectors ordered by magnitude from smallest to largest. A vector store can be represented by a linked list of vectors and an integer called the *size* of the store.

3 Requirements

3.1 File requirements

This assignment requires four files: `prog6.c`, `prog6.h`, `test6.c`, and `log6.txt`. The `prog6.c` file holds an implementation of a vector store as a linked list. `prog6.h` is a header file for `prog6.c`. The `test6.c` file holds testing code for the vector store implementation; it contains a directive to include the `prog6.h` header file. The testing code in `test6.c` calls the functions defined in `prog6.c`. And the `log6.txt` files holds a copy of your log book.

3.2 Implementation Requirements

Write an implementation of a vector store that includes the following components. Put the implementation code in `prog6.c`.

1. A definition of a type `vector` as a type of records with two mutable fields `x` and `y` of type `double` such that the pair $(\texttt{x}, \texttt{y})$ represents a vector in 2-dimensional Euclidean space.

2. A function named `vec_mag` of type

   `vector` → `double`

   that maps a vector to its magnitude.

3. A vector store data structure defined by the following:

   (a) A type `node_record` of records having mutable fields `data` of type `vector` and `next` of type (equivalent to) `node_record *`.

   (b) A type `node` defined as the type `node_record *`.

   (c) A type `vec_store` defined as a type of pointers to records with mutable fields `seq` of type `node` and `size` of type `int`. The `seq` field holds either the value `NULL` (if the size of vector store is 0) or the first node of a linked list of nodes.

   (d) A constructor `make_vec_store` of type

   → `vec_store`

   that constructs an empty vector store whose size is 0.

   (e) A selector named `get_vec` of type

   `vec_store,int` → `vector`

   such that (1) `get_vec(`s, i`)` gets the ith vector in s if $0 \leq i \leq$ s`->size` $- 1$ and (2) prints an error message that i is out of bounds if $i < 0$ or s`->size` $- 1 < i$.

   (f) A mutator named `delete_vec` of type

   `vec_store,int` → `void`

   such that (1) `delete_vec(`s, i`)` deletes the ith vector in s (and thus decrements the size of s) if $0 \leq i \leq s$`->size`-1 and (2) prints an error message that i is out of bounds if $i < 0$ or s`->size`$-1 < i$.

   (g) A mutator named `insert_vec` of type

   `vec_store,vector` → `void`

   such that (1) `insert_vec(`s, v`)` inserts vector v into s (and thus increments the size of s).

   (h) The vectors in s are stored as a linked list of vectors ordered by magnitude from smallest to largest. (The order is preserved by the mutators.)

3.3 Testing Requirements

Write code that tests the vector store implementation and that satisfies the following requirements:

1. The implementation of the vector store is tested by making a representative series of calls to the constructor, selector, and mutators.
2. The results of these test calls are printed out when the program is executed.

Put the testing code in `test6.c`.

3.4 Submission Requirements

Create a directory named `assign6`. Put the files `prog6.c`, `prog6.h`, `test6.c`, and `log6.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, December 11, 2009.**

4 Marking Scheme

This assignment is worth 100 points allocated as follows:

1. **Objective** (checked automatically by software)
   - (a) Program files are 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.

5 Extra Challenge

1. Do the assignment over with a vector store defined as a data structure that holds a queue of vectors.