CIS*2500 W20 Assignment 4

Questions & Answers Part 5

“I was just wondering how to deal with the square function in Q1b I realise that it returns another list but I'm confused about how to store it, should I just assign it to one of the ten sorted lists? Or should I just print it out and dispose of it afterwards?”

Print it out and dispose of it afterwards.

I thought of storing the result, but thought it would over complicate an already large assignment.

Mark Wineberg

“Is it possible to define denom part of struct Fraction as long, and not unsigned long? We could then forget about any type problems between num and denom -- which makes both print_fract, simplify and add_fract (handling overflows) much easier as we are dealing with one type. It anyway covers overflows etc. correctly (with LONG_MAX, not both LONG_MAX and ULONG_MAX)”

I agree. I didn't think about the interaction between type conversion, and the need to have two overflow functions. I was only thing that you need to perform a cast and the unsigned long would allow for larger denominators. I didn't mean to overly complicate the question by requiring two different overflow functions.

Consequently, you can change the denominator to long instead of unsigned long. I will change the assignment to give the option.

Mark Wineberg

“For assignment 4 question 2b when keeping the original commands without the '|' for specifying the list. when the command is run without the '|' does that mean that it is just executed to the first list or to all of the lists?

Neither, actually. There would be an 11th list not in the array to handle the 'a' instead of 'a|n' case. However, we won't be testing for that.

Mark Wineberg

"Do we have to have all assignment sources in one directory using same file of sort_list.c and sort_list.h? (which will include #ifdef for FRACT and also map/reduce/filter, so will be complete, even when some assignments do not use them)
So we will have content like that:
sort_list.c, sort_list,h, Makefile (with all targets for all assignments), fraction.c, fraction.h, a4q1a.c, a4q1b.c, a4q2.c, a4q3.c
OR
instead use separate directories for each assignment with each having its own sort_list.c and sort_list.h?"

You can have various sub directories for the programs if you like (as long as your make file can handle that), but you **must have only one file for any set of functions from Sort_List**.

As Sort_List is an ADT, you should only have one version of it that can be used in multiple places. Admittedly, this can only really be effectively done using void pointers and more thorough use of function pointers, but I want to simulate this as much as possible with as little copy code as possible.

Mark Wineberg

"Are we allowed to assume that no two input strings will have the same string length, and therefore the same key? I fear that it could add some complications in my code."

No. String lengths can be equal. There are equal string lengths in my example; not by design, but that happens to be true for the section I chose from Lewis Carroll's The Hunting of the Snark.

However, there is no extra code involved in handling this. Tied entries are allowed to be in any order when moving across the sort links.

Just use the sorting code from the lecture notes.

Mark Wineberg

"Just wanted to let you know of a typo in the input/output from the sum_sq_diff pdf.
Commands says sum|2
Output says sum list = 1, result = 242"

Thanks! I have now updated it.

Mark Wineberg

“In terms of sorting the list could I just make a function that performs bubble sort on the current linked list? or would I need to sort as I insert nodes?”

Sort as you insert the nodes.

The code is not just designed for the "commands", but in general. You cannot know when the next push or append is and to re-sort every time you use some other command (such as print or remove_smallest) is *much* more time consuming then doing the sort as an element comes in, as you are continually sorting and resorting the entire list after every command, rather than just when a push or append happens.

Re: the type of sort to use

Use the code in the lecture notes.

The sort algorithm implemented there is called "insertion sort". Bubble sort is not faster than insertion sort in general. In fact, bubble sort in *slower* than insertion sort for linked lists (more links being modified).

Mark Wineberg

PS Why did you want to use bubble sort?

“Hello, I’m working through q1b and I was wondering for the “diff” function, do you want us to always subtract the greatest value from the smallest or always subtract list2 from list1? Or maybe I totally misunderstood what the function is supposed to do, that’s always a possibility.”

No. Just take the difference in the order the lists are provided. If the first number is smaller than the second, the result will be negative.

What you are suggesting is called "absolute difference".

See "Output from sum_sq_diff.input" in the Help section of A4 for an example.

Mark Wineberg

“When taking the instructions from a file, can we assume a max length per line? Say 20 chars for the actual command, 80 characters for the input string, and 100 characters of extra tolerance for a total max line length of 200. Or should we make this dynamic, line in other assignments.”

You can assume a max length. The assignment is structured so this is natural. Just state your assumptions in the read.me

Mark Wineberg

“Suppose you have a line that starts with a push/append command -- how are we supposed to deal with the whitespace in between? We can't use fgets because we don't know the width of the line, nor can we use fscanf because it cannot deal with the spaces in between the words. fread doesn't seem feasible either because we need to know the number of elements -- same as fgets. How do we isolate the line of characters (that is supposed to go into the 80 char array) without knowing any of these ahead of time?”

Assume that the LewisCarroll.input file is the pattern upon which all other input will follow.

This file uses 2 spaces between the command and the string.

Mark Wineberg

“My question is regarding the append function in q1 of A4. Should the append function also sort in ascending order of key like push?”

Yes.

Mark Wineberg

“I was wondering if in Q1a we can assume that all the lines will be formatted correctly. For an example in the int file the command will always be followed by what is required :

$a 4 3.01

Or is the a chance that we come across something similar to this and have to handle it accordingly:

$p 3 <missing the key value>

Thanks”

Assume all lines will be formatted correctly. The assignment is long enough without thoroughly bullet proofing it.

That being said, it actually isn't that hard to error check the input: command not on the allowed list, number of arguments is wrong given the command, and argument type is wrong. All of this is concentrated during the reading phase of the command input.

Again, you can assume the formatting of the input will be correct. While not hard, writing the error checking code will take a bit of time, and it isn't the point of the assignment.

Mark Wineberg