Assignment 1 CS 4540 Due 9/18 before class submit to eLearning

Purpose: practice with C structs, pointers, and multiple files tied together by a make file and header files.

People, each person has linked list of phone numbers.

Peple Searchable by Nickname, or (Last, First) or nickname returns person information and list of phone numbers

Phone numbers: searchable by (999) 999-9999 returns list of people (First, Last) with that phone number with type like Home, Cell, School and Is primary t/f

Person struct,

Last name, first name, nickname

Head of linked list of phones

Phone Struct:

Number as string (999) 999-9999

Head of linked list of people with this phone number

Each person and phone struct is in its own array for **sequential search**, each has its own functions to search by given key

Linked list of phones, each node

1. pointer to phone struct
2. Type of phone from ENUM variable (Home, Cell, School)
3. Boolean is primary
4. Pointer to next

Linked list of all people, each node

1. Pointer to person struct
2. Pointer to next

Each struct and linked list must be defined in its own header file which also has function prototypes for its functions.

The functions for each struct (person, phone, linked list of (phones and people) must be in their own files

The input will be from a file whose name is in the command line arguments. If the file name is not there or can not be opened an error message and prompt for correct file loop until correct. The open file must be in its own function with a pointer to the FILE pointer to be filled as a parameter as well as the command line argument array and count.

Then main will call another function to read the file and create the data structures which receives the FILE pointer as a parameter as well as the two arrays for person and phone. Read first line after

<person> create person struct, put in array, read each phone number, create phone struct, put in linked list for the person, and put the person in the linked list in the phone struct, till </person>

Then main will call another function which is the user interface putting up a simple menu with numeric choices for

1. search person which returns information about the person and their phone numbers
2. Search phone which returns information about each person with that phone including type and if primary
3. Quit

On quit the program must free all allocated memory for the linked lists and person and phone structs before returning from main.

Input file uses standard html markup just to tell each person from the phone numbers

<person>

Last, Middle, First, nickname

(999) 999-9999 Type, true/false

(phone #) type, true/false

…

</person>

<person>

….

</person>

EOF

Remember, one and only one return or exit from any function, the last line of code. Return from main not exit.

No global variables that you define or declare, all variables in functions or passed in and out.

Rule of thumb no function longer than what will fit on one screen without scrolling.

Program must compile with -Wall with no errors on the CS department servers and run without errors. Use only standard libraries.

Turn into eLearning only makefile, source .c files and .h files. No .o or executables

No including .c files in another file only include header files. All dependences of .c to header files must be in the makefile, make must have clean target to remove .o files and executable. Can run make then touch any .h file and have only the .o files that include it recompiled then executable relinked. No extra lines of actions in makefile that make would generate automatically for you as shown in class.

Any code or ideas on how to write code from online or other sources MUST be documented and referenced.