Lab 2

Working with Strings

Strings in C are actually char arrays. In this lab you manipulate strings in its char array form with the help of the functions found in string.h in the C standard library.

1. In this lab, you will write 2 functions that manipulate strings. The function declarations should go in a header file called lab2.h and the function code should go in lab2.c.
2. The first function should be called convert_to_hex. This function takes a string representation of a hexadecimal digit and converts it to decimal. The function parameter should be a char pointer (char*) and should return an integer. The function can assume good input, which would be 0-9, a through f, and A through F.

   Important Notes: To convert to hex, you will need the pow function in the math library. In order to use that library, you need to include the math.h header file. You'll also have to link it to your program during compilation. Use this command to compile your code:

   ```
   gcc -o lab2 lab2.c -lm
   ```

   To use the lab2_main.c tester program, compile like this:

   ```
   gcc -o lab2 lab2_main.c lab2.c -lm
   ```

   If you are not familiar with converting from hex to decimal, see this link:

   https://www.rapidtables.com/convert/number/hex-to-decimal.html

3. Write a second function called remove_substring that searches a string for the delimiter &&| and removes it. The function should take a char pointer and return a new char pointer. You only need to remove the first instance of &&| found in a string not all instances of it.
4. You have been given the file, lab2_main.c to test your functions. Use that to confirm your functions are working. To compile lab2_main.c to test your program make sure to compile both lab2.c and lab2_main.c together like this:

   ```
   gcc -o lab2 lab2.c lab2_main.c -lm
   ```

5. Submit lab2.h and lab2.c to Blackboard.

Grading

80 points - Implementation

Each function is worth 40 points.

10 points

The code is in the correct file – header (lab2.h) and source (lab2.c).

10 points

Your name: Add your name in the comments section at the top of lab2.c

Readability: Make sure your code is indented and neatly commented. Do not leave commented out code in your submission. Comments describing what your code does is ok. Don’t leave old code in the source file.

Compilation: If your code does not compile, you will receive a 0 on the lab. Once notified, you will have 3 days to fix the lab to receive partial credit, up to 75%.