CSC 102 – Assignment #3

Assigned: Wednesday, Feb 25, 2010
Due: Tuesday, March 9, 2010 at beginning of class

Source Code: Please include a header section at the top of your source code with your name, date, purpose, and pseudocode. All submitted source code must compile as-is on orca and the executable must be able to run on orca.

Email submission: Send source code as an attachment to Preetam.Ghosh@usm.edu. The subject line of your email should have the following form: CSC 102, name, hw #

Purpose of this assignment:
1. Work with strings.
2. Learn to use structures.
3. Gain more practice with arrays, files and functions.

Assignment:

Write a program that can decrypt each of the strings stored in the file “input.txt” and print out the sum of the string lengths.

Each character must be decrypted by applying the decryption algorithm to its ASCII value. The decryption algorithm is:
• if array index is even, subtract 1
• if array index is odd, subtract 3

That is, the first character has an array index of 0, which is considered even, so subtract 1 from its ASCII value to produce the new ASCII value. The second character has an array index of 1, which is odd, so subtract 3 from its ASCII value to produce the new ASCII value. Repeat for all characters.

Example: The encrypted string
erh
can be decrypted to produce
dog

Additional requirements:
1. In main(), declare an array of structures variable. The two members of the structure will be the string (an array of chars for storing the encrypted string) and an integer (storing the corresponding string length).
2. Write a function that will be passed the strings one by one, which it then processes character by character, printing out each decrypted character as it is determined. The declaration will look like this:

```c
void decrypt(char input[]);
```
3. Don’t calculate the length of the array. Instead, look for the terminating null that is at the end of each string.
4. Don’t hard-code any ASCII values (or their character equivalents) in your program; it’s unnecessary.
5. If you do the math correctly to produce the decrypted ASCII values, then they will be in the range that is valid for ASCII characters.
6. From main, print out the total length of all the strings that will be the sum of the lengths of each string stored in your array of structures. You should store the string lengths into the second member of the structure using the “strlen” function.
7. Assume that your array of structures will have 3 elements (corresponding to the 3 strings from “input.txt”). Also, assume that each string can be at most of length 100, and use a char array to store your strings inside the structure.