CSCE 3600: Systems Programming

Recitation Assignment 7 – Implementing Mutex Locks

Available: Week 9 Due: Week 10

PROGRAM DESCRIPTION:

In this assignment, you are provided with working code that does the following:

1. You input a sentence (containing no more than 50 characters).
2. The program will read the sentence and put it into an array of characters.
3. Then, it creates one thread for each character in the sentence.
4. The goal of the program is to capitalize each letter that has an odd index.

The given program actually does this, but lacks the synchronization of the threads, so the output is not correct. You will need to provide the synchronization using mutex locks. Specifically, you are to (1) declare the mutex lock, (2) initialize the mutex lock, (3) lock and unlock the mutex lock at an appropriate location that results in the code working as expected, and (4) destroy the mutex lock. Be sure to place the mutex locks so that your program works correctly *every time*. Do not remove code or functions – you are to add the synchronization pieces only.

When compiling using the GNU C compiler, be sure to include the **–lpthread** flag option.

SAMPLE OUTPUT (user input shown in **bold green**)**:**

```
$ ./a.out
Please enter a phrase (less than 50 characters): when all else
fails, read the  instructions
The original sentence is: when all else fails, read the instructions
The new sentence is [0]:      w
The new sentence is [0]:      w
The new sentence is [0]:      w
The new sentence is [0]:      w
The new sentence is [0]:      w
The new sentence is [0]:      w
The new sentence is [3]:      N
The new sentence is [0]:      w
The new sentence is [0]:      w
The new sentence is [0]:      w
The new sentence is [0]:      w
The new sentence is [0]:      w
The new sentence is [9]:      E
The new sentence is [12]:     e
The new sentence is [14]:     f
The new sentence is [15]:     A
The new sentence is [16]:     i
The new sentence is [17]:     L
The new sentence is [17]:     L
The new sentence is [19]:     ,
The new sentence is [20]:
The new sentence is [21]:     R
The new sentence is [22]:     e
The new sentence is [23]:     A
The new sentence is [23]:     A
The new sentence is [25]:
The new sentence is [26]:     t
The new sentence is [27]:     H
The new sentence is [27]:     H
The new sentence is [27]:     H
The new sentence is [29]:
The new sentence is [31]:     N
The new sentence is [30]:     i
The new sentence is [31]:     N
The new sentence is [32]:     s
The new sentence is [35]:     U
The new sentence is [34]:     r
The new sentence is [35]:     U
The new sentence is [36]:     c
The new sentence is [37]:     T
The new sentence is [39]:     O
The new sentence is [41]:     S
The new sentence is [41]:     S
```

The problem is that the output should look something like:

```
$ ./a.out
Please enter a phrase (less than 50 characters): when all else
fails, read the  instructions
The original sentence is: when all else fails, read the instructions
The new sentence is [0]:      w
The  new  sentence  is  [1]:   H
The  new  sentence  is  [2]:   e
The  new  sentence  is  [3]:   N
The new sentence is [4]:
The  new  sentence  is  [5]:   A
The  new  sentence  is  [6]:   l
The  new  sentence  is  [7]:   L
The new sentence is [8]:
The  new  sentence  is  [9]:   E
The  new  sentence  is  [10]:  l
The  new  sentence  is  [11]:  S
The  new  sentence  is  [12]:  e
The new sentence is [13]:
The  new  sentence  is  [14]:  f
The  new  sentence  is  [15]:  A
The  new  sentence  is  [16]:  i
The new sentence is [17]:  L
The new sentence is [18]:     s
The new sentence is [19]:     ,
The new sentence is [20]:      
The new sentence is [21]:     R
The new sentence is [22]:     e
The new sentence is [23]:     A
The new sentence is [24]:     d
The new sentence is [25]:      
The new sentence is [26]:     t
The new sentence is [27]:     H
The new sentence is [28]:     e
The new sentence is [29]:      
The new sentence is [30]:     i
The new sentence is [31]:     N
The new sentence is [32]:     s
The new sentence is [33]:     T
The new sentence is [34]:     r
The new sentence is [35]:     U
The new sentence is [36]:     c
The new sentence is [37]:     T
The new sentence is [38]:     i
The new sentence is [39]:     O
The new sentence is [40]:     n
The new sentence is [41]:     S
The new sentence is [42]:
```

REQUIREMENTS:

- No comments are required for this recitation assignment, except for your name at the top of each program.
- Your program should be named “**`rec07.c`**”, without the quotes.
- Your program will be graded based largely on whether it works correctly on the CSE machines (e.g., `cse01`, `cse02`, …, `cse06`), so you should make sure that your program compiles and runs on a CSE machine.
- Although this assignment is to be submitted individually (i.e., each student will submit his/her own source code), you may receive assistance from your TA and even other classmates. Please remember that you are ultimately responsible for learning and comprehending this material as the recitation assignments are given in preparation for the minor assignments, which must be completed individually.
- Please do not share this assignment or your work with other students to allow them the opportunity to benefit from this exercise and learn this material.

SUBMISSION:

- You will electronically submit your program to the **Recitation 7** dropbox in Canvas by the due date and time. No late recitation assignments will be accepted.