PROG 20799

Quiz #4: Data Streaming

Prof. Georg Feil / Winter 2020

Working individually, complete the following programming question and hand in using SLATE when your professor indicates time is up (SLATE Assignments tab, Quiz 4). Attach your source files (main.c, Queue.c, Queue.h) individually. Do not use Zip or RAR.

This quiz is open book. You are allowed to use your notes and access these web sites:
– SLATE (slides and example programs)
– Our recommended or supplementary textbooks (O'Reilly online)
No other web sites are allowed. Please follow the C99 language standard, and use proper coding style and standards as you would for an assignment. Worth 2% of your grade. Marked out of 10 marks.

In this quiz you'll write a program that demonstrates data streaming using a queue. In data streaming we can "transmit" information by placing it into a network queue. At the receiver the received information can be obtained by dequeueing it from a network queue. The code for a queue ADT that uses pointers (equivalent to linked list) is posted in SLATE under Week 9. Download Queue.c and Queue.h from SLATE. Use this queue implementation for your program – do not use any other queue implementation or you will receive a mark of zero.

1. The supplied queue implementation stores integers ('int' data type). Your data streaming program needs to work with a stream of characters, not a stream of integers, so first convert the queue implementation (Queue.h and Queue.c) so that it holds individual characters instead of integers. Do not make any other changes to Queue.c/h or marks will be deducted.

2. Write a new source file called main.c that contains a queue declared as a module variable (static global). Then implement the following three functions which use the queue, declared as indicated.
   1) `void transmit(char data[])`
      This function should enqueue all the characters in the string 'data'. Use a loop to enqueue the characters, and remember that a null character ('\0' or just 0) marks the end of a C string. The null is not considered part of the string and should **not** be enqueued.

   2) `void receive(int length)`
      This function should attempt to dequeue 'length' characters and print them. If there are not enough characters in the queue then stop when the queue is empty. Characters printed by one call to receive should appear on the same line.

3) Write a main function that does the following:
   a) Initialize the queue.
   b) Call ‘transmit’ with the following strings:
      “The quick brown fox jumps over the lazy dog”
      “Don’t cut corners”
      “Spot barked up the wrong tree”
   c) Call ‘receive’ with the following lengths: 30, 40, 25
      Note: There should be three calls to ‘transmit’ and three calls to ‘receive’.

The output from your program should look exactly like this. The output from each receive call should be printed on a new line.

```
The quick brown fox jumps over
 the lazy dogDon't cut cornersSpot barke
d up the wrong tree
```

Please see the start of the first page for submission instructions.