CS212 – Lab 4 – Implementing a queue with an array

A. Implement the following functions for manipulating a queue (without actually declaring the queue), based on an array of characters of size QUEUESZ, where QUEUESZ is a "*const int*" or a #include value, with a value of 20 (for now). The name of the queue is your choice. Pick a name. It will be declared in step B.

   a. Push – adds one element to the end of the queue (accepts a character as input), returns an int:

      i. -1 if the queue is already full,

      ii. 0 if the item was successfully added

   b. Pull – removes the front element (if it exists) and returns the value:

      i. NULL if there were no elements (queue was empty)

      ii. The Front element itself, if there was one

   c. Front – does NOT remove anything. Returns the value:

      i. NULL if there are no elements (queue is empty)

      ii. The front char element if there is one

   d. isFull – does NOT remove anything. Returns an int:

      i. 0 if there are no elements (the queue is empty)

      ii. 1 if the queue is full

      iii. -1 for all other cases

B. Using the above functions, write a program that:

   a. Declares the queue with a size of QUEUESZ characters

   b. Uses your functions above to do the following:

      i. Accept input characters, 1 at a time until the symbol ^ is input

      ii. Push each character onto the queue

      iii. Output the queue (front to end) after all inputs have been received

Notes:

1. Your main program should have NO access to the queue except through the above functions. The queue array & value of the index to the front and end ARE global, so all your functions can get at them, but the main code must never look at or use them.
2. If the queue is full and more data keeps coming, your program should ignore the data and reply,

   "The queue is full. Enter the '^' character to stop.".
3. Repeat the above message if more input (other than "^" keeps coming).