Name: Elijah Andrushenko

Date: 9/28/16

Student ID: 011476324

TA: Hunter Wilkes

1. Write a function enqueue() for a queue that is implemented with a singly-linked list, and that has the following declaration and precondition:

int enqueue (Queue \*pQueue, Data newData);

The function should allocate space for a node and space for the string pointed to by pStr. The amount of space required for the string is based on the length of the string found in newData. This will require two calls to malloc(). The function should attempt to insert the node at the end of the queue (pTail). If the node is successfully inserted into the list, then the function must return 1. If memory could not be allocated for the node, then the function must return 0. All structures are defined below:

typedef struct data { char \*pStr; // you will need to allocate space // for the string separately! } Data;

typedef struct node { Data item; struct node \*pNext; } Node;

typedef struct queue { Node \*pHead; Node \*pTail; } Queue

int enqueue(Queue \*pQueue, Data newData)

{

Node \*pMem = makeNode(newData);

int success = 0;

if (pMem != NULL)

{

success = 1;

if (pQueue->pTail != NULL)

{

pQueue->pTail->pNext = pMem;

pQueue->pTail = pMem;

}

else

{

pQueue->pHead = pQueue->pTail = pMem;

}

}

return success;

}