Problem 2 PDF

1. The problem states that we must use a FIFO (first in, first out) data structure. This means that the data structure is a queue. This data structure must also support O(1) time for removal(deletion) and minimally O(N) time for addition (insertion). This means I must implement a linked list. Specifically I used a doubly linked list.

**public** getvalue (**int** Index){

**for** (**int** i=0, i<index, i++){

tail=tail.next;

}

**return** tail.data;

1. This data structure achieves O(N) time for add because at must we will have to traverse N-1 elements to insert a new element. If we are adding to the back of the queue (Enqueue) this takes O(1) time since we always have a pointer to it.

**public** **void** add(**int** a){

**if**(totalnodes==0,){

a= head.data;

totalNodes++;

}

tail=tail.prev;

a=tail.data

totalNodes++;

}