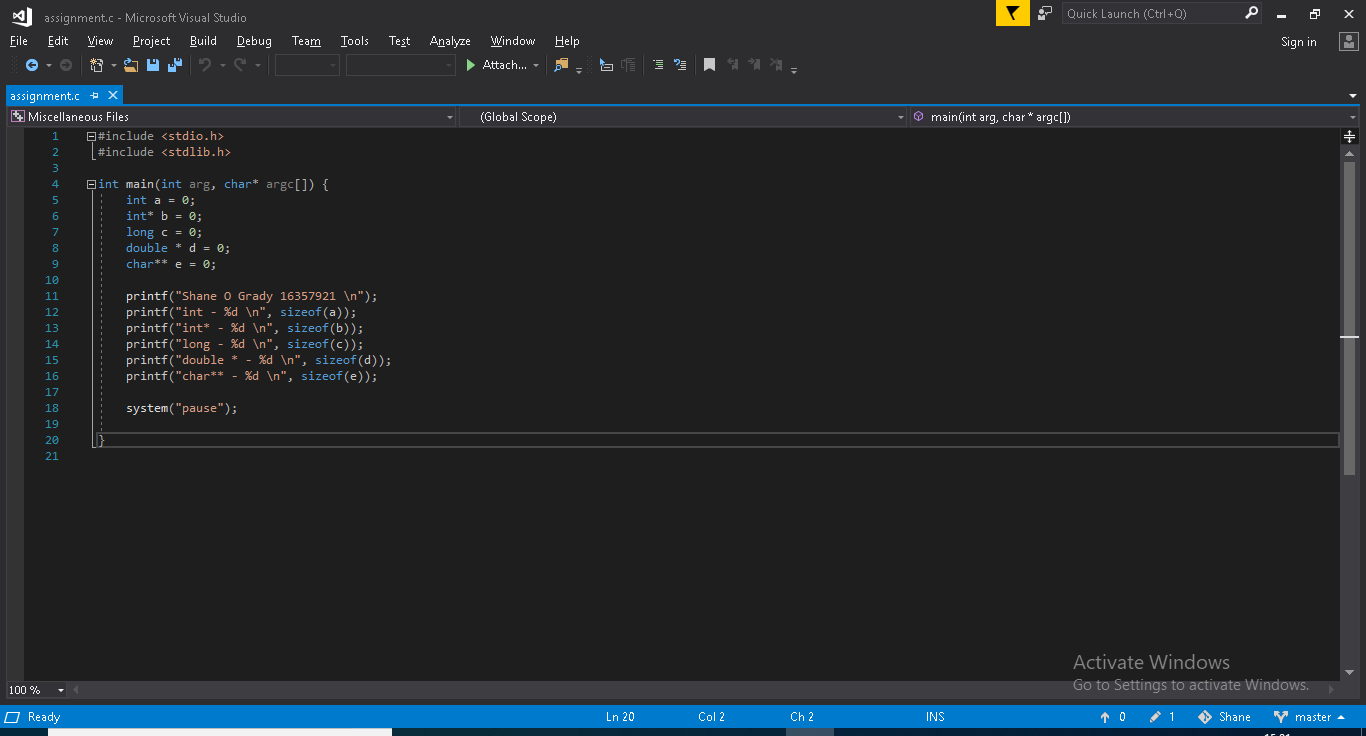
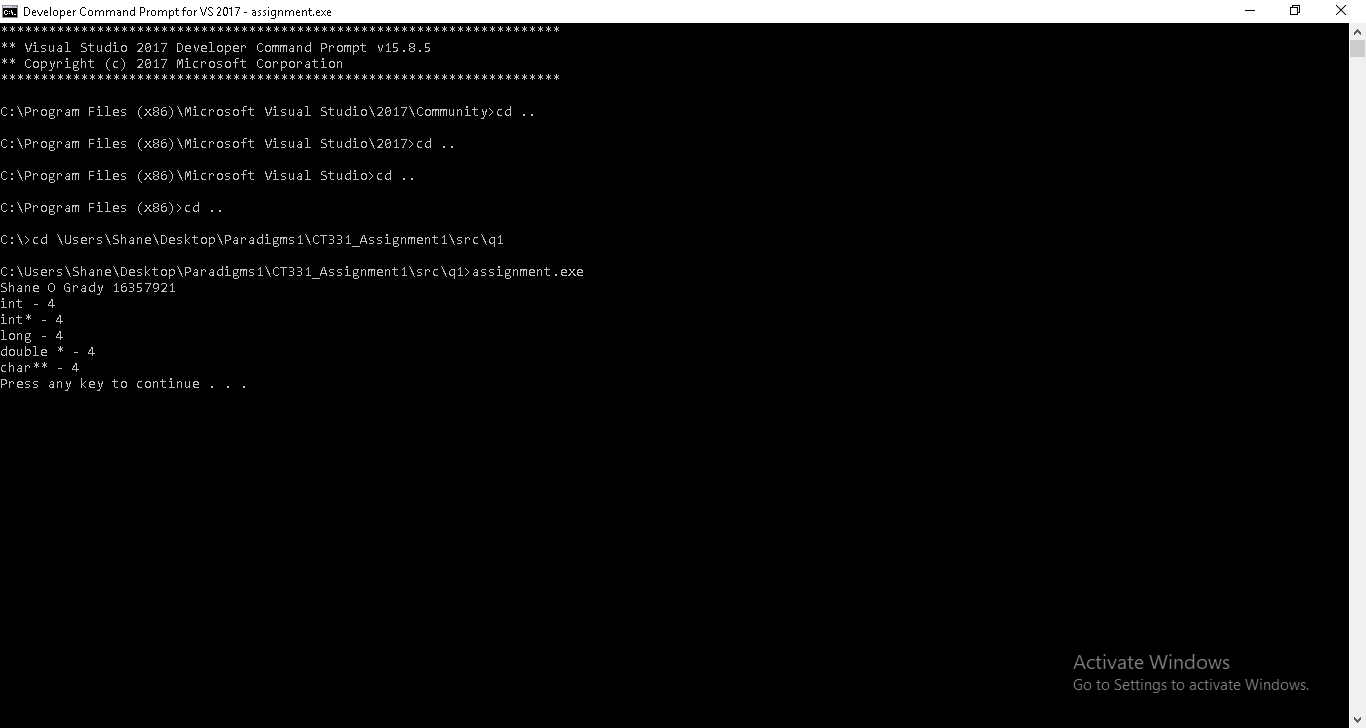
**Shane O Grady 16357921**

**CT331 Assignment 1**

Question 1(A) Screenshot:

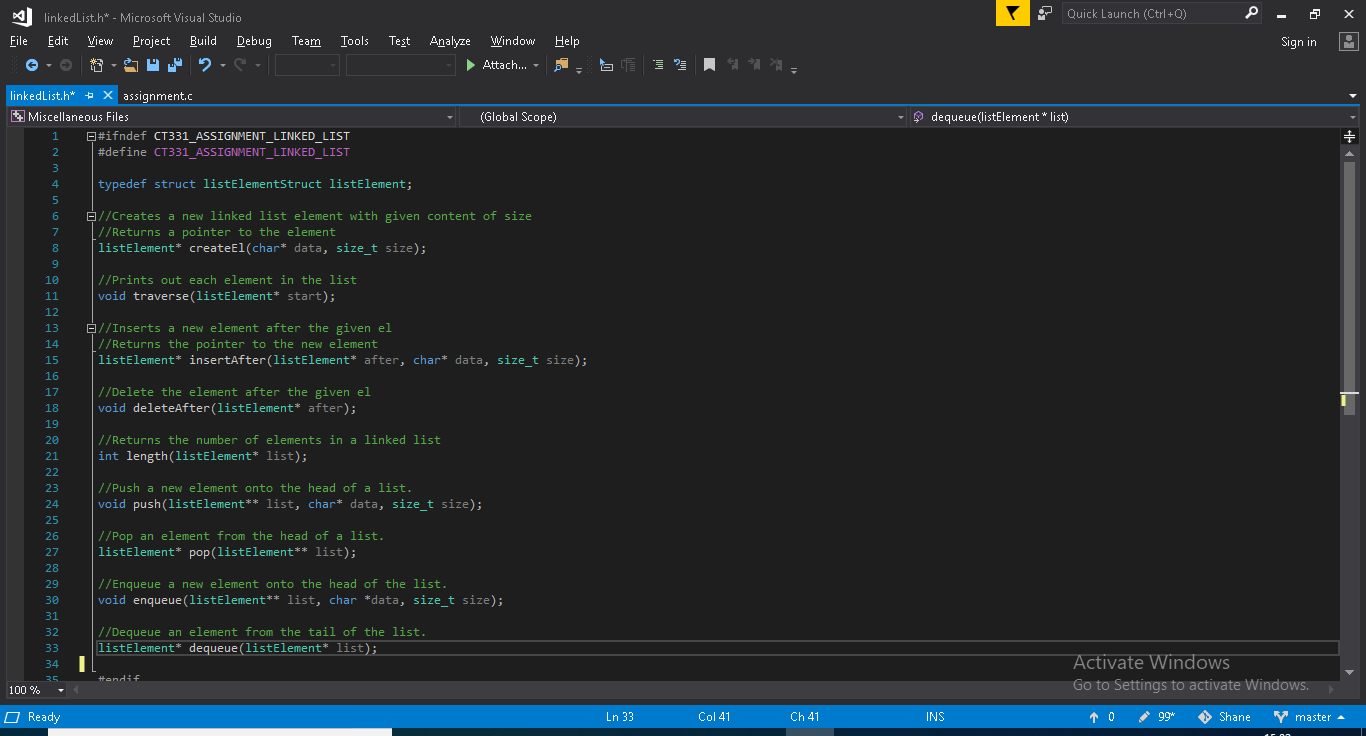


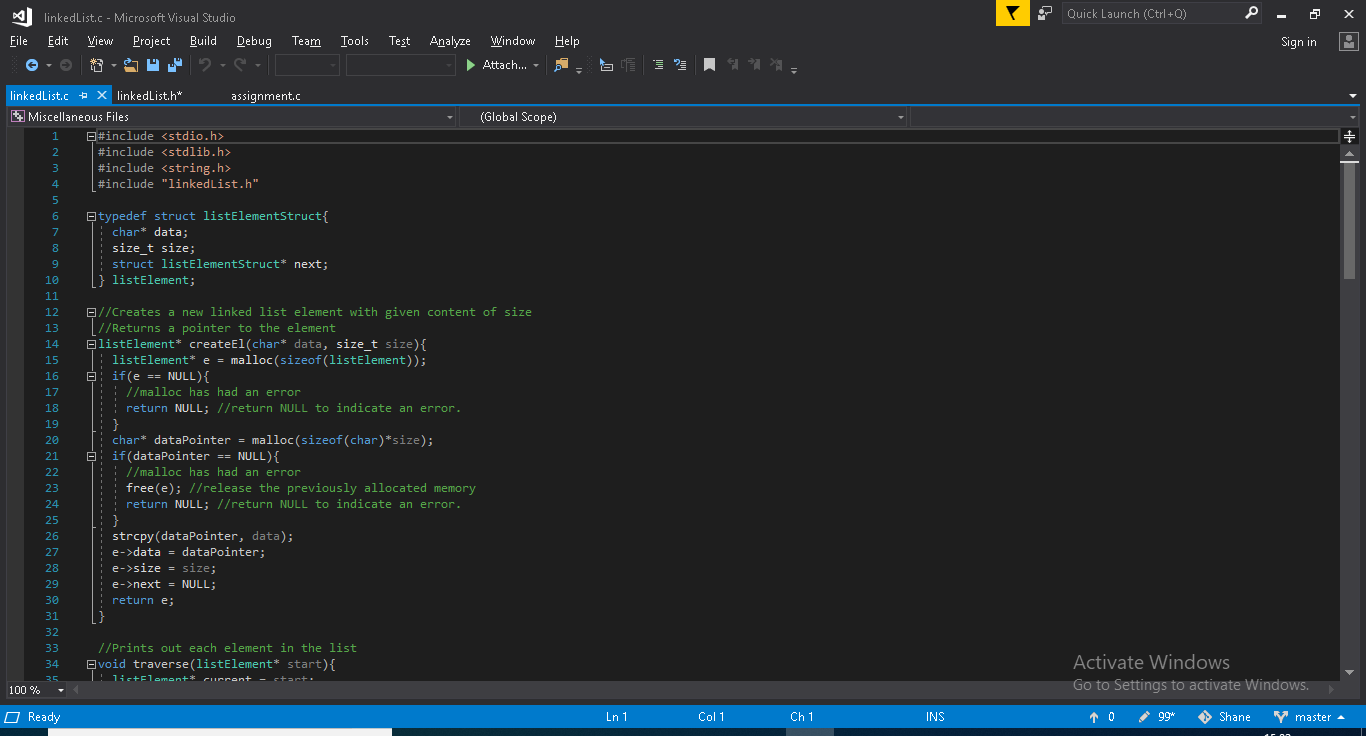


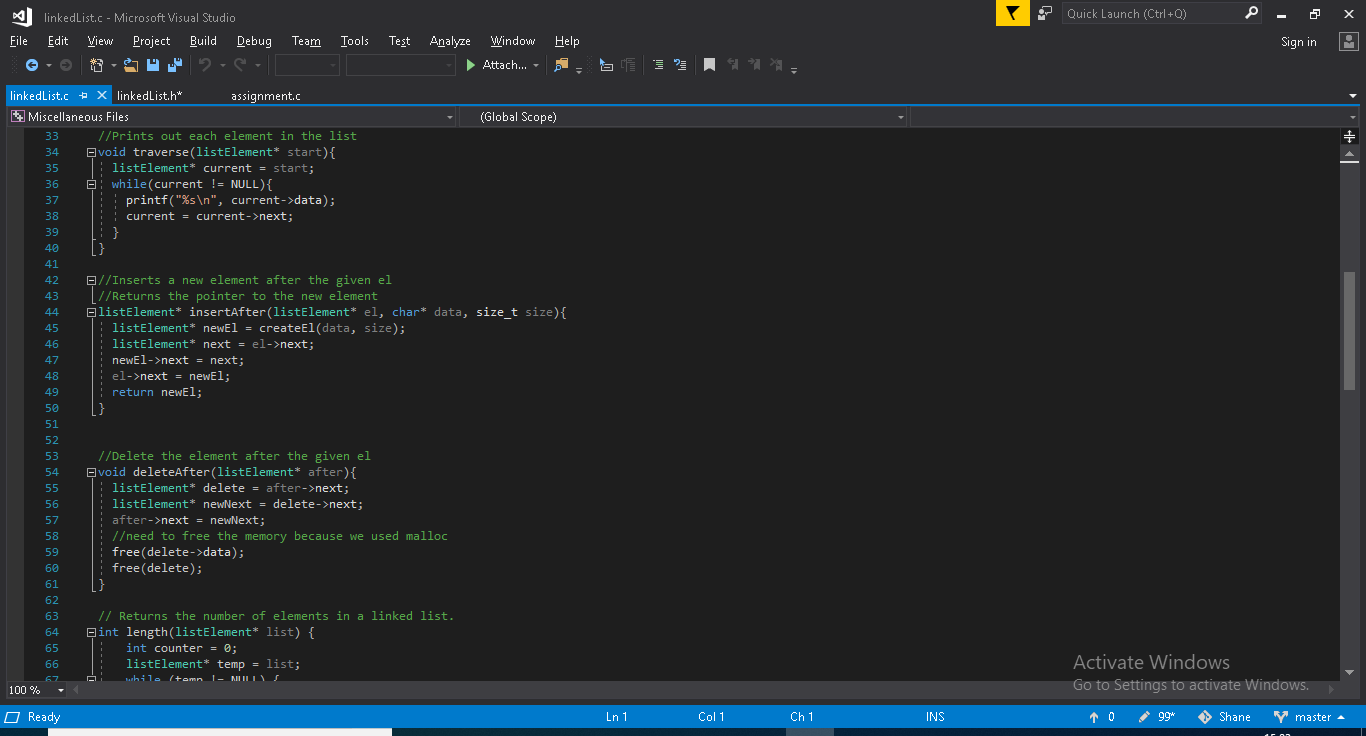
Question 1(B) :

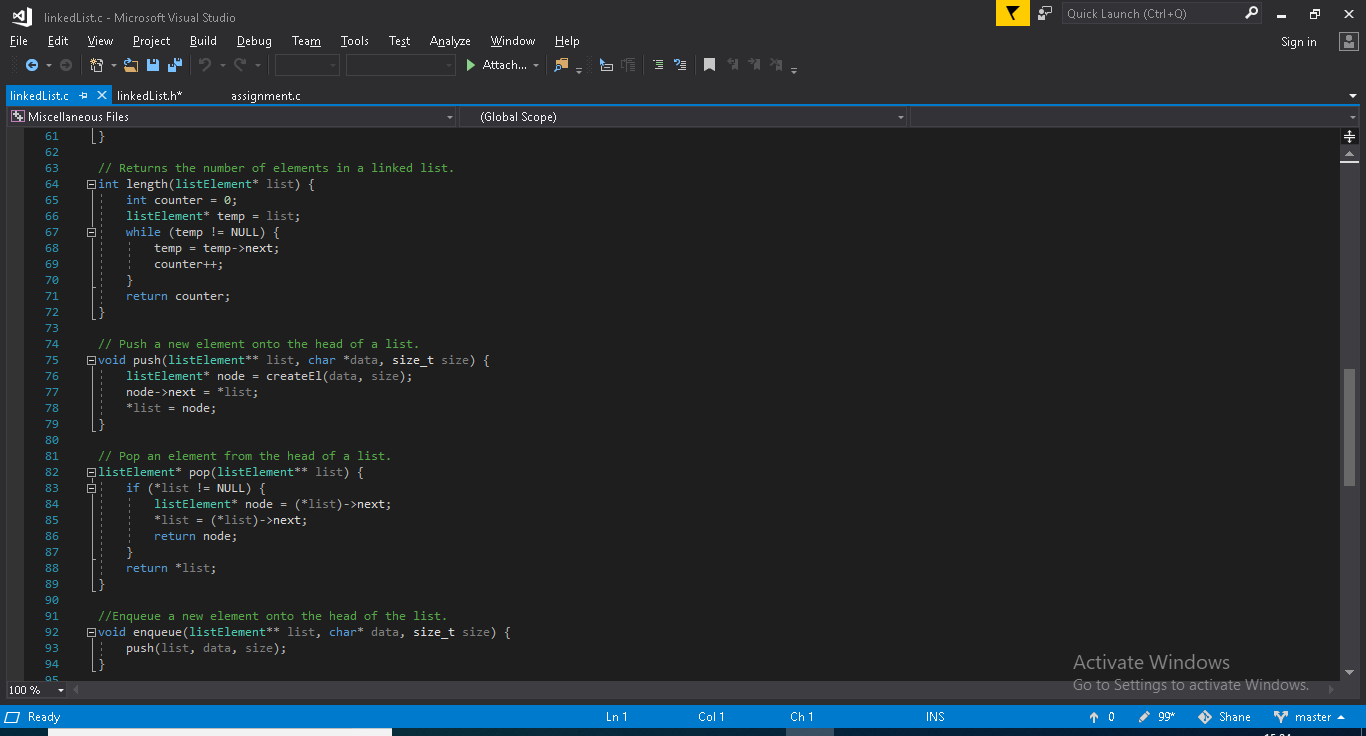
The size of every primitive value turned out to be 4 which surprised me. However, I discovered that a pointer has 4 bytes which explains why the char and double have a result of 4. As well as this I discovered that both the int and long have a size of 4 bytes through research carried out online

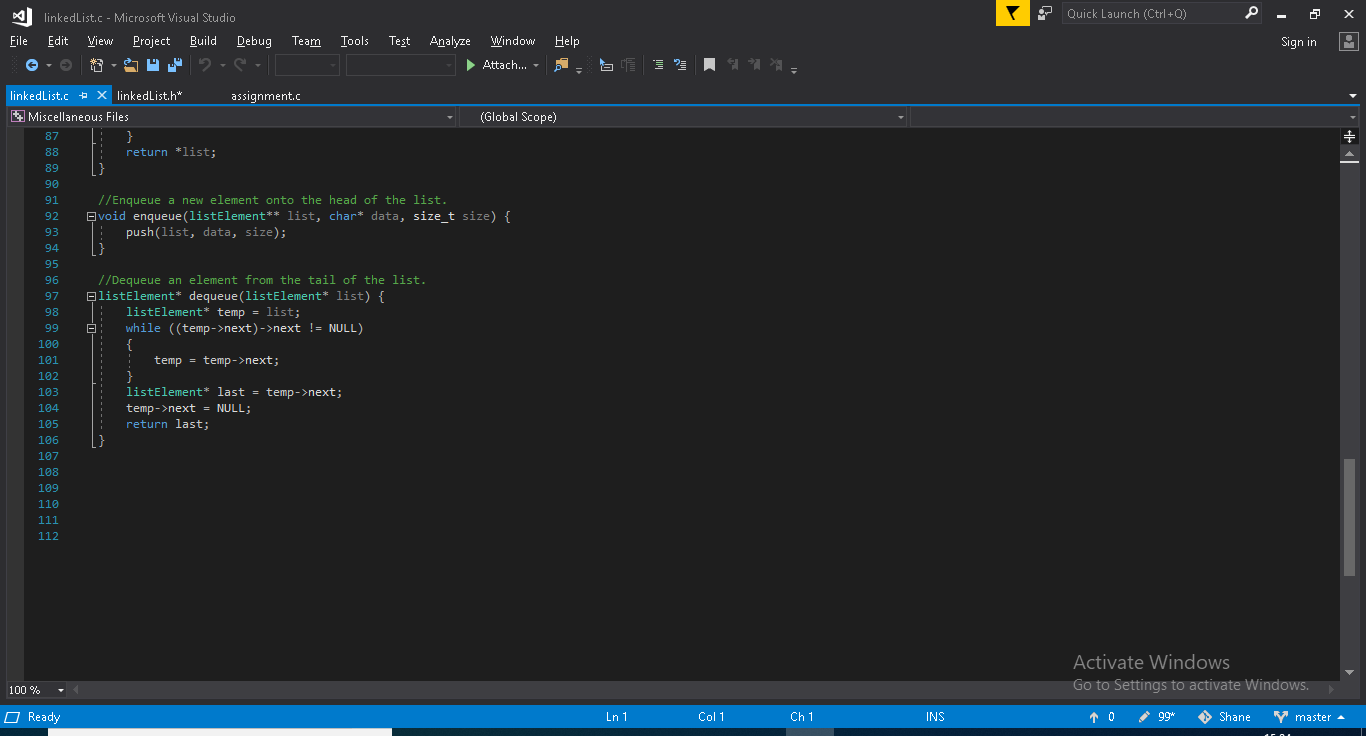
Question 2 Screenshot:

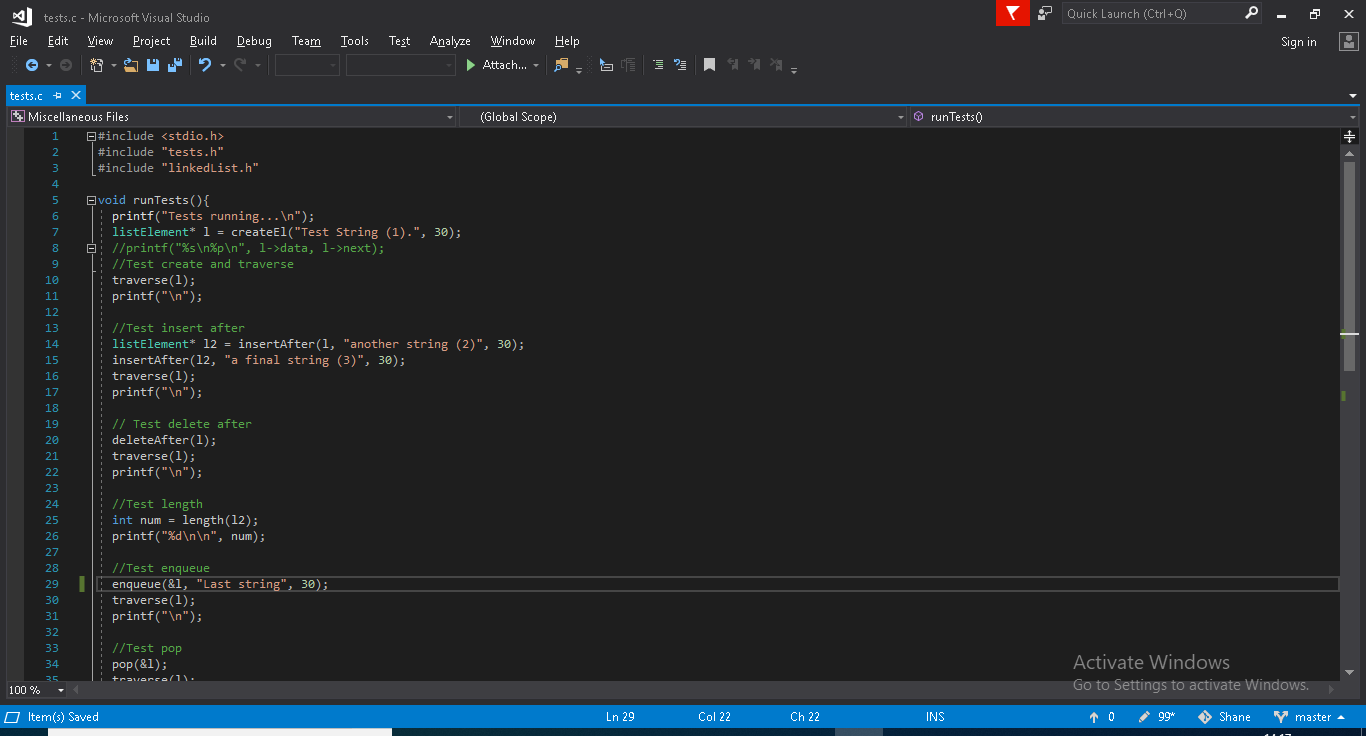


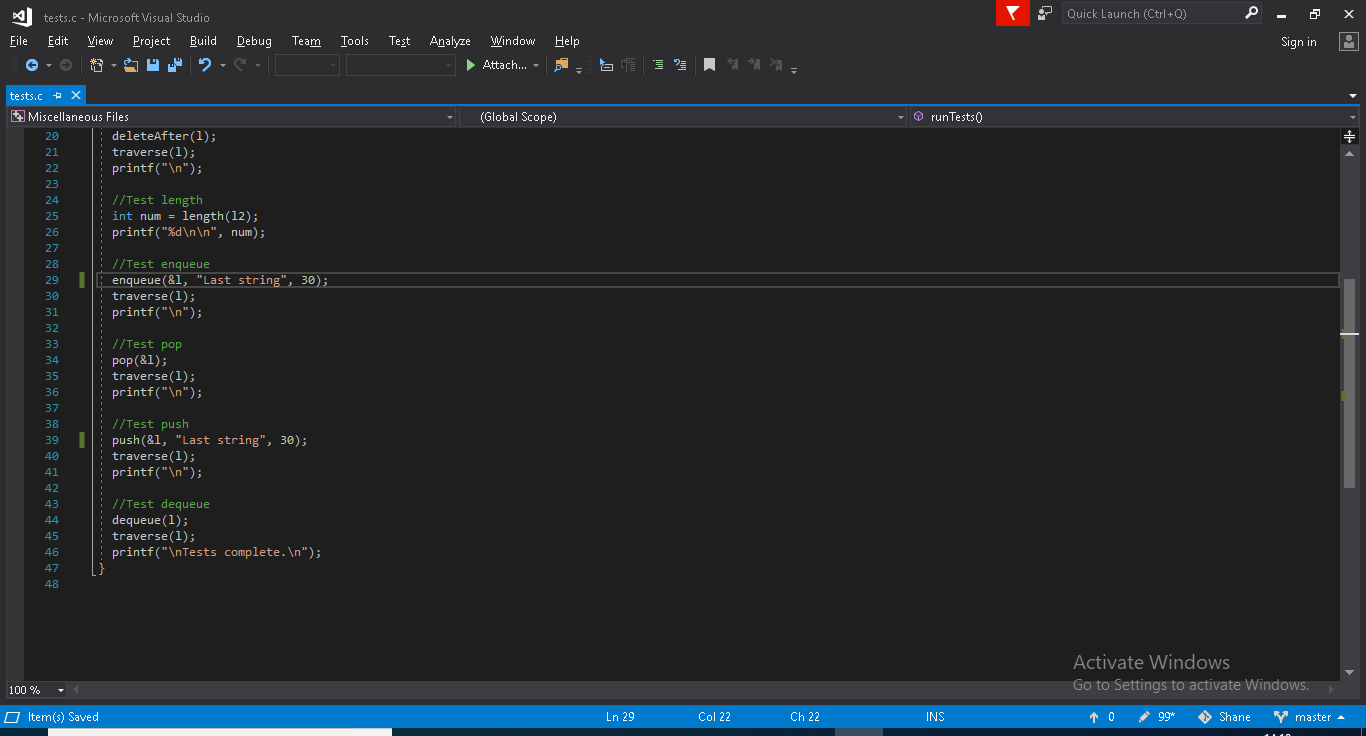


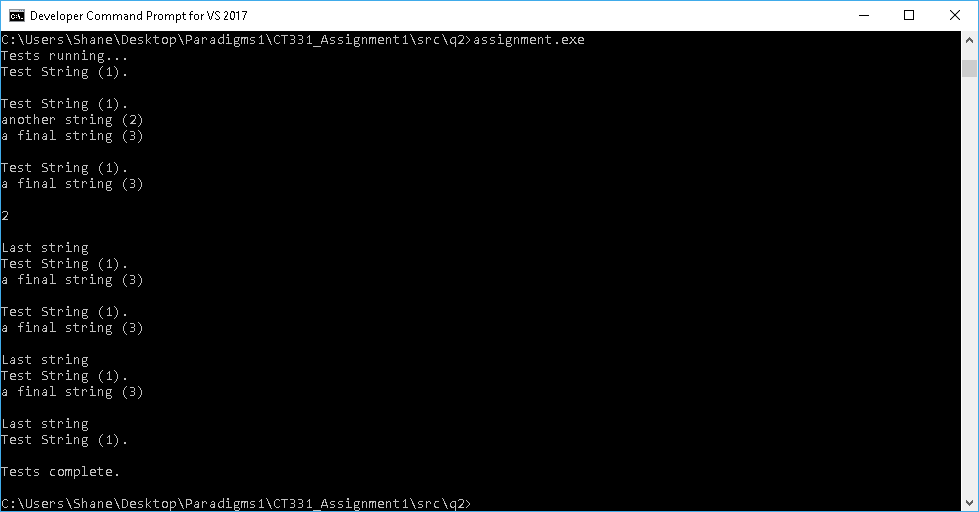




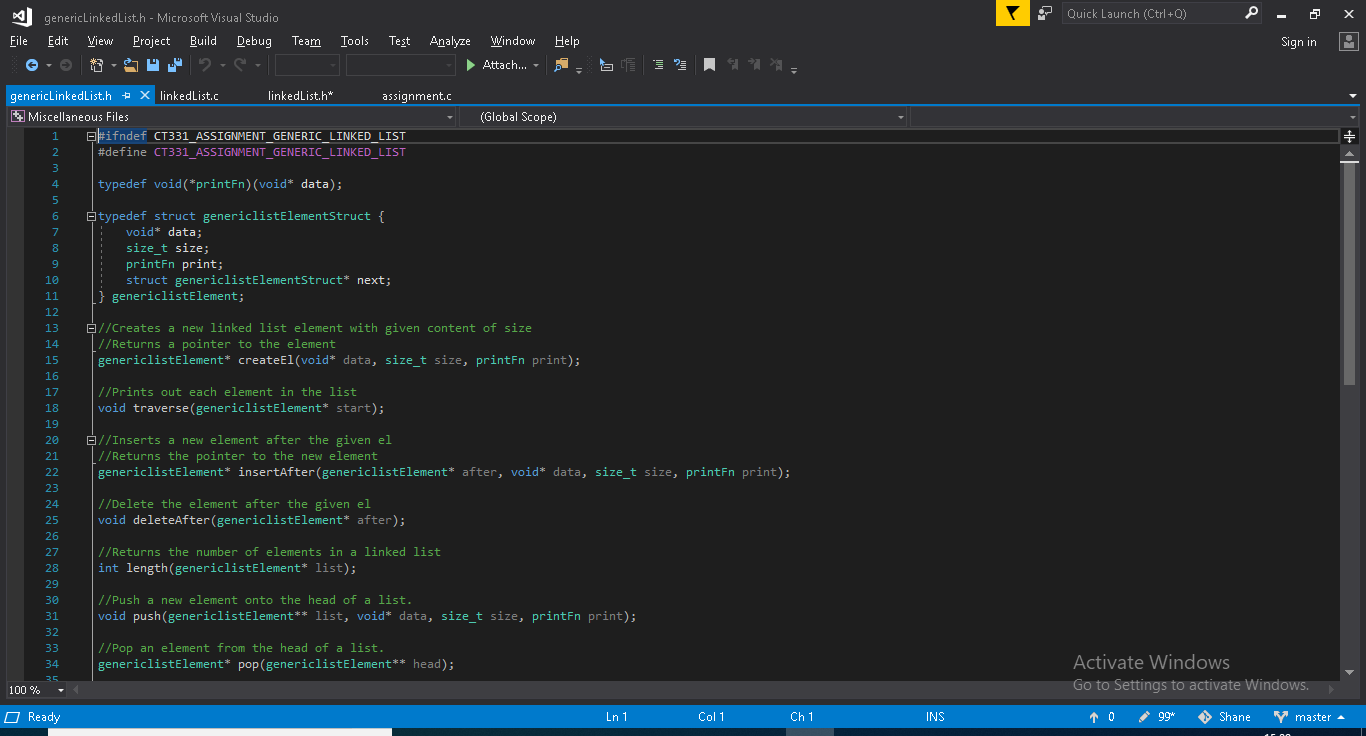


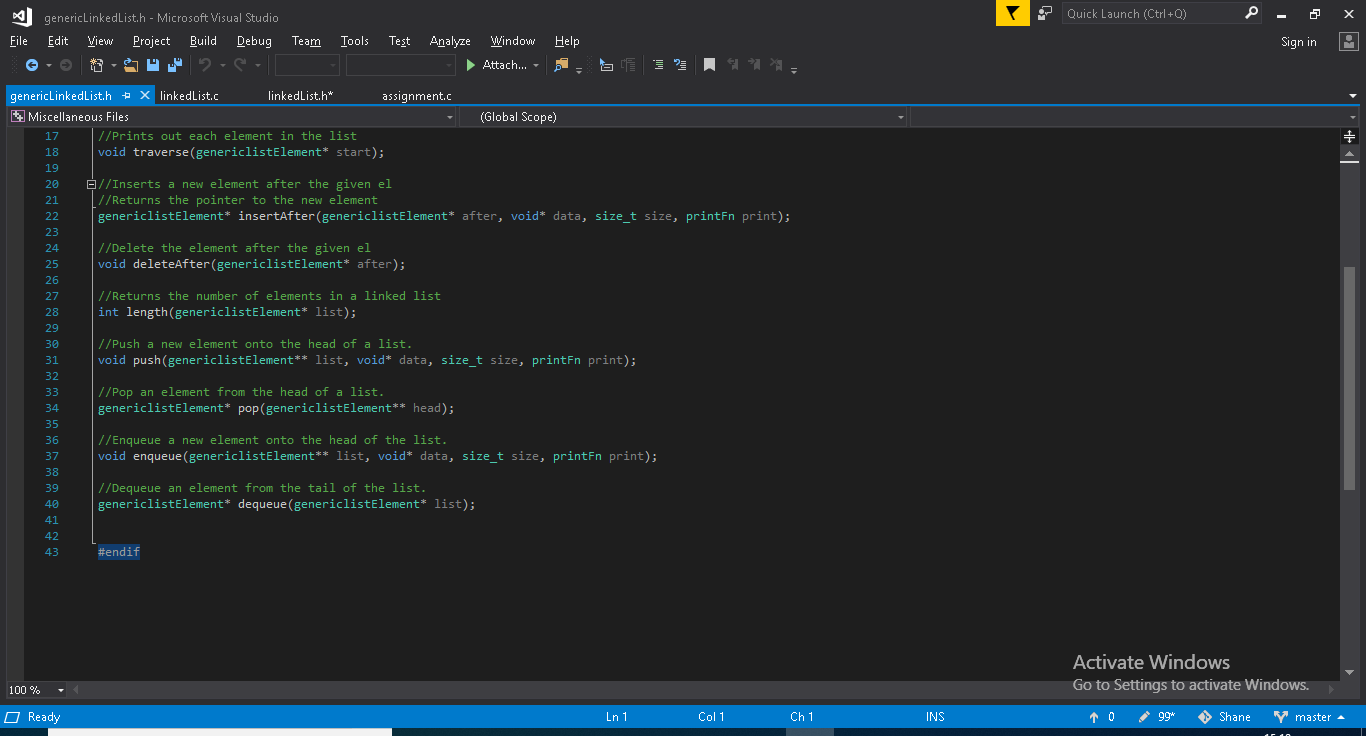


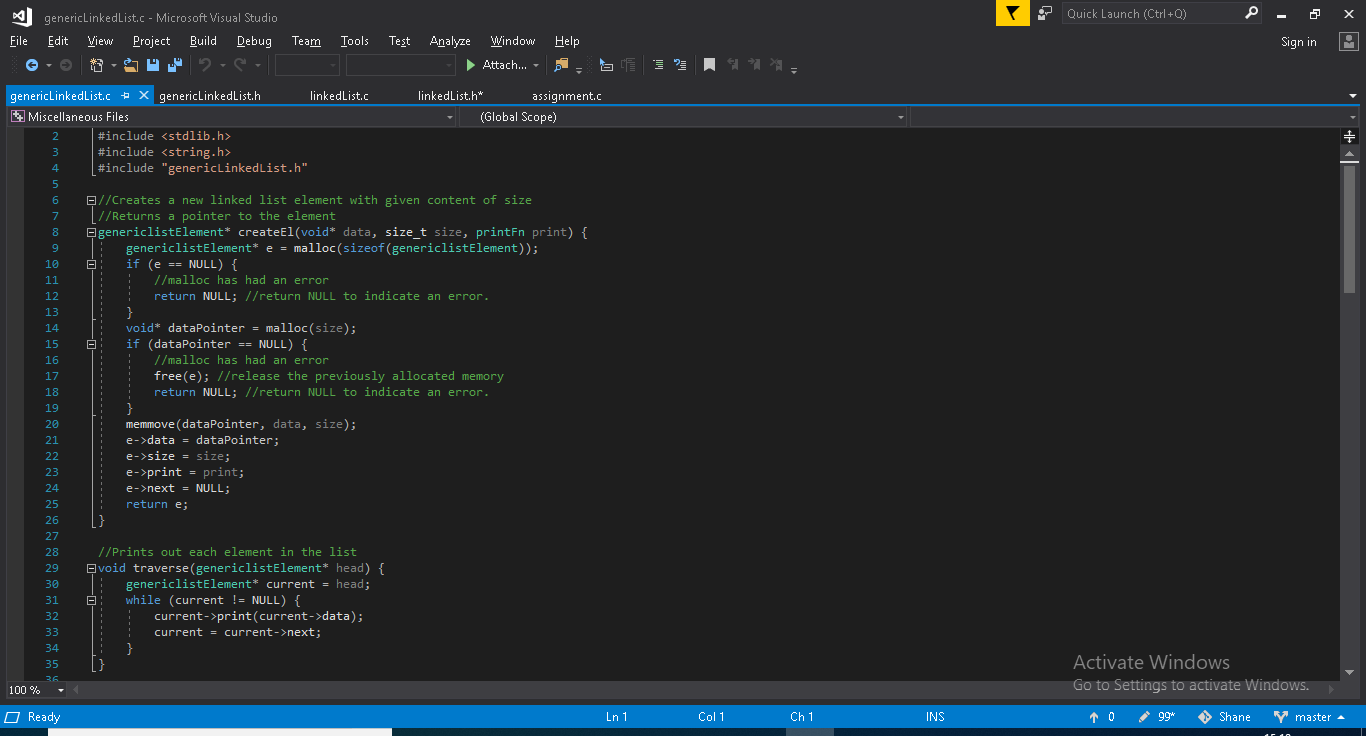


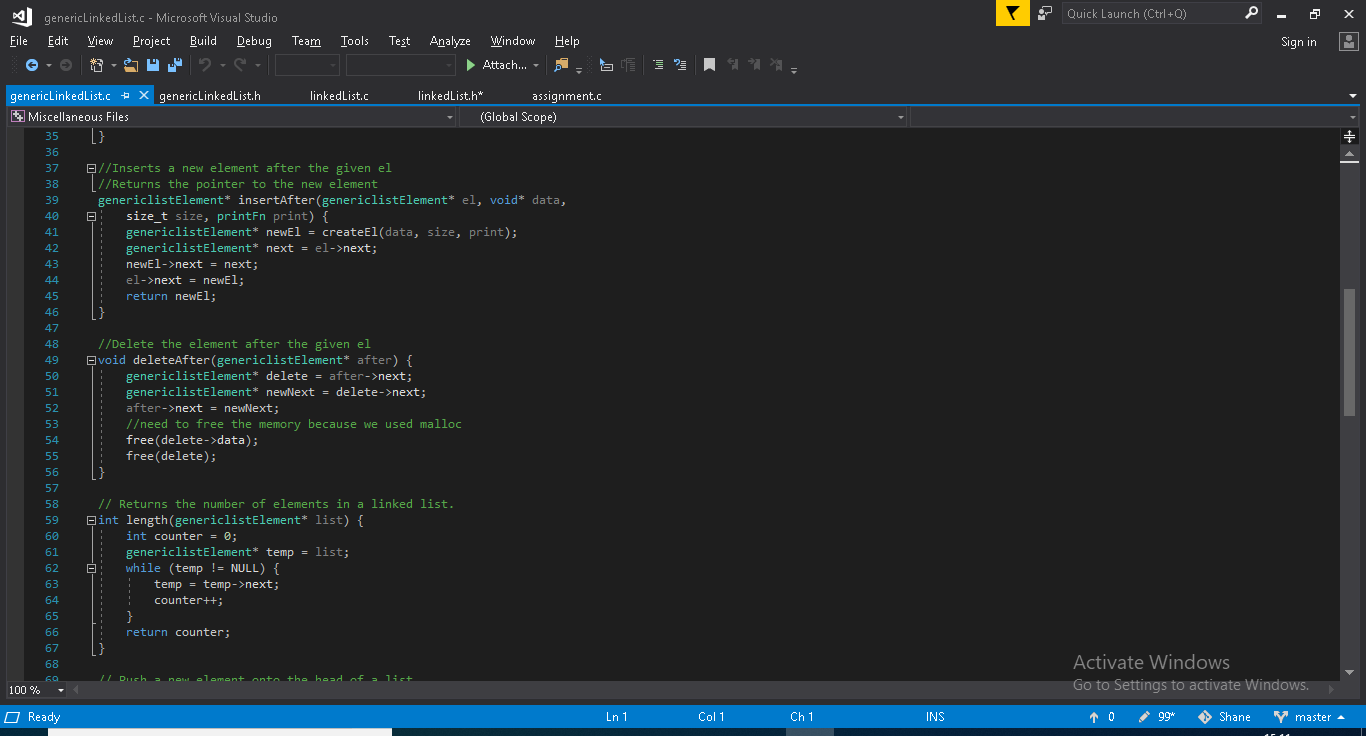


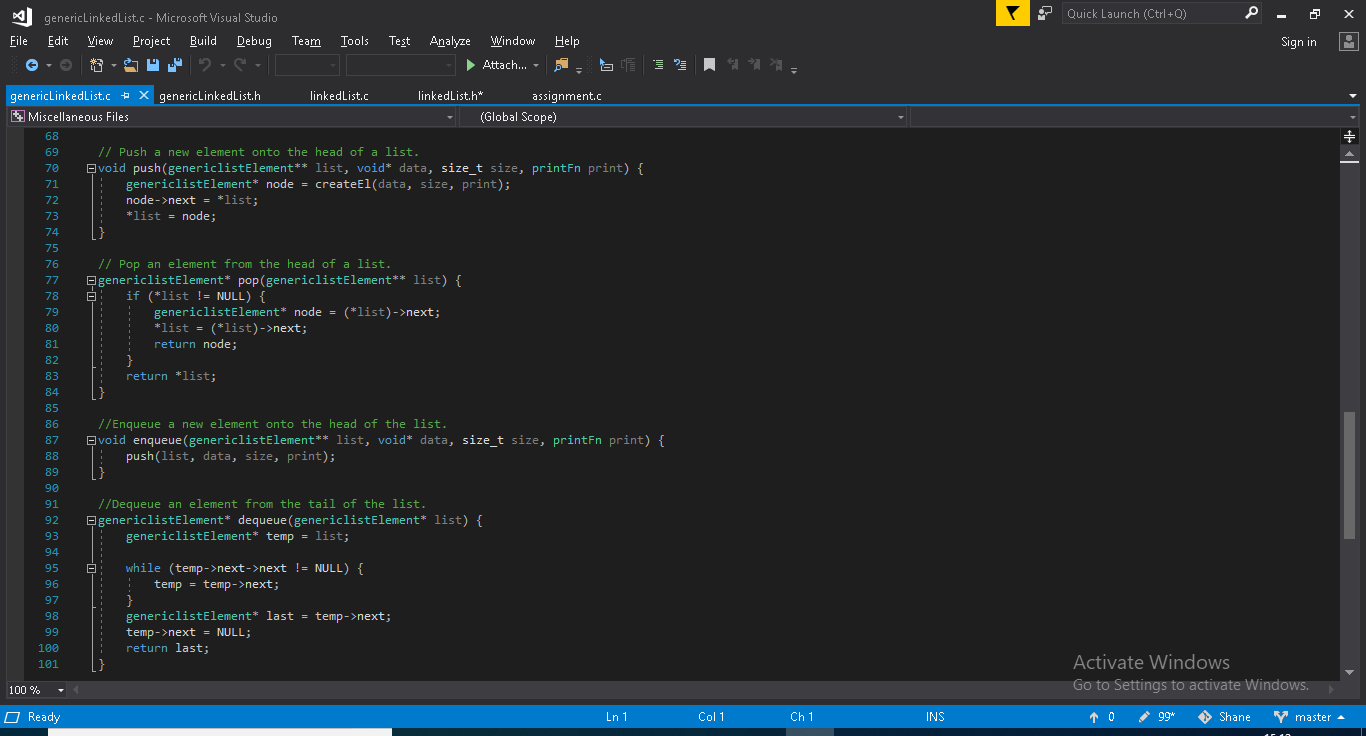
Question 3 Screenshot:

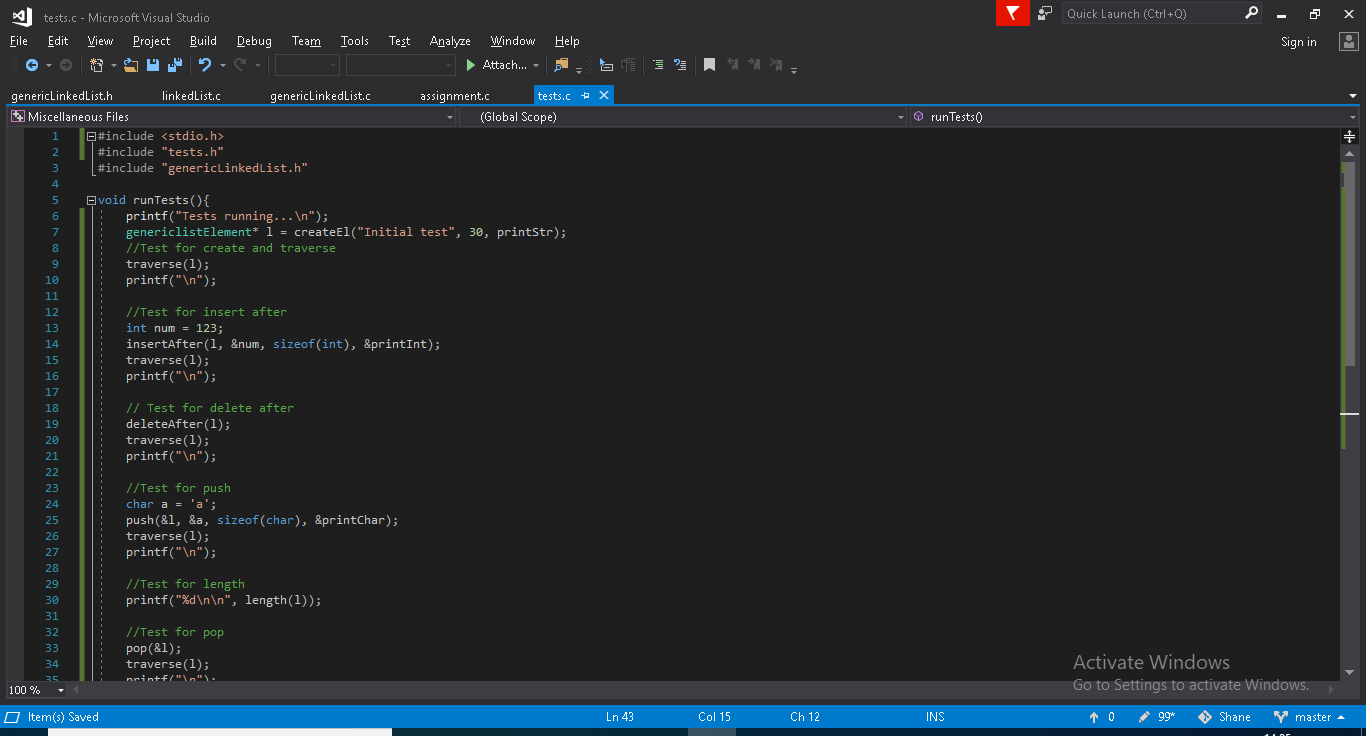


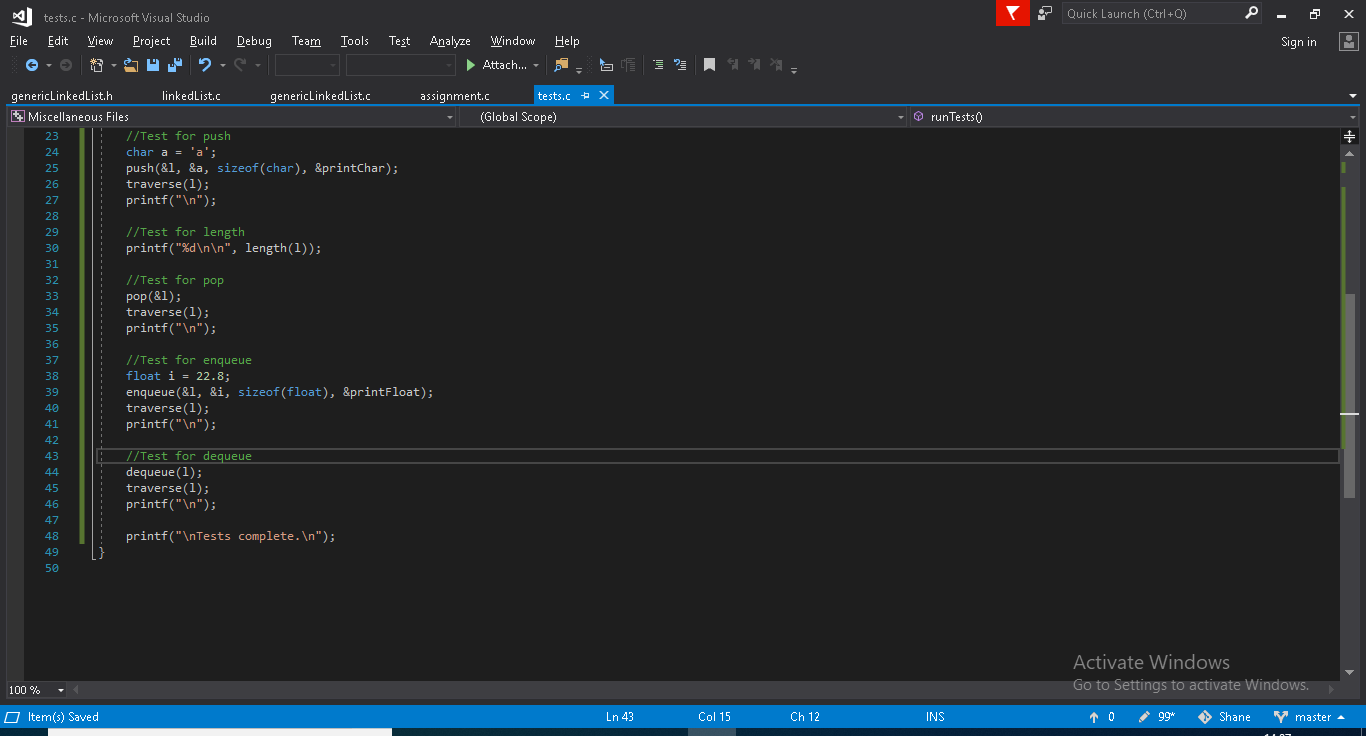


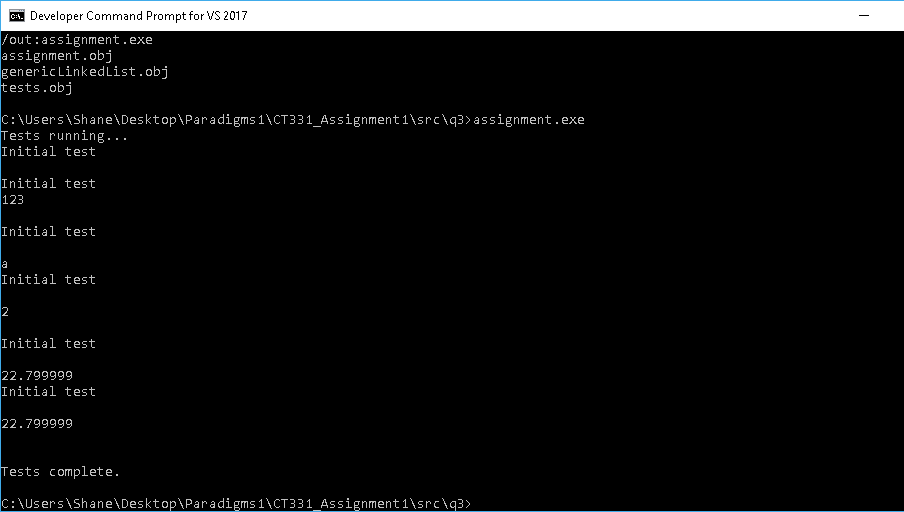












Question 4:

i)

Traversing a linked list in reverse will have high memory intensity as singly linked lists cannot be reversed directly. Once you reach the last node which is now the head you can traverse backwards. In order to traverse the linked list in reverse it is required to iterate one position back each time. This requires a lot of memory as a result.

ii)

To reduce memory intensity, the structure should be changed to a doubly linked list. Unlike the singly linked list, it can travel in two directions, next and previous. Depending on the size of the list, this can make a major impact when traversing in reverse as it would require much more memory to do so in a singly linked list.