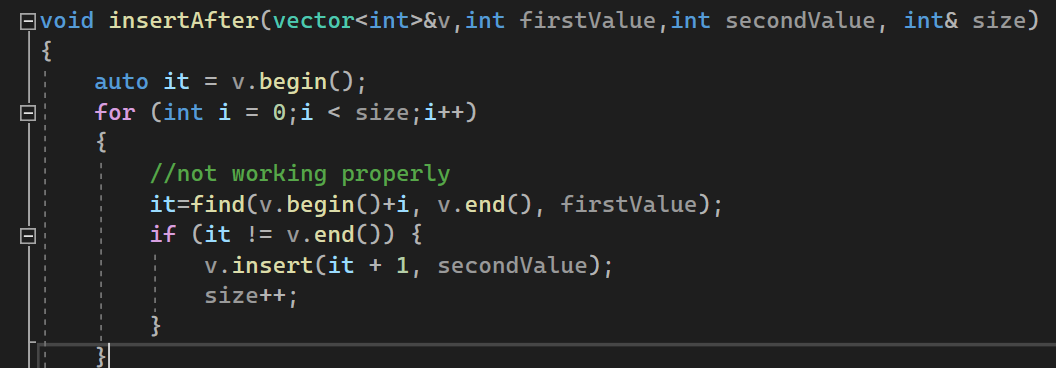
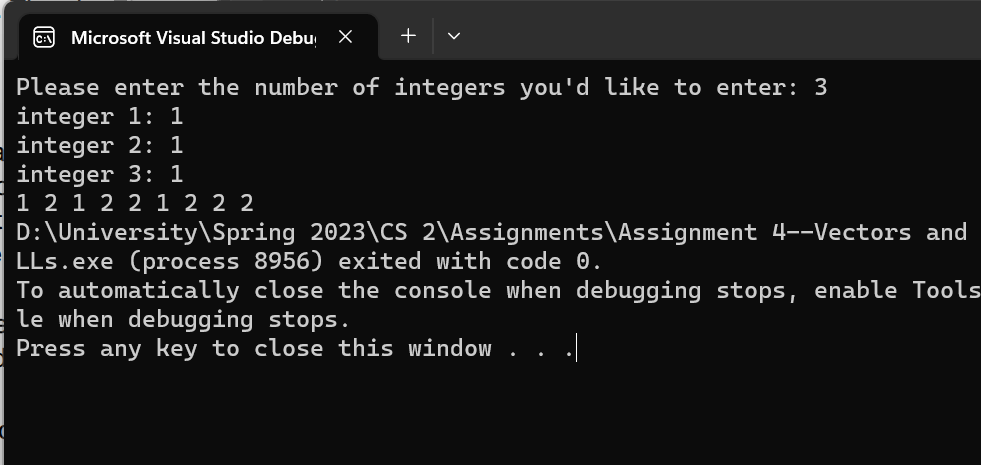
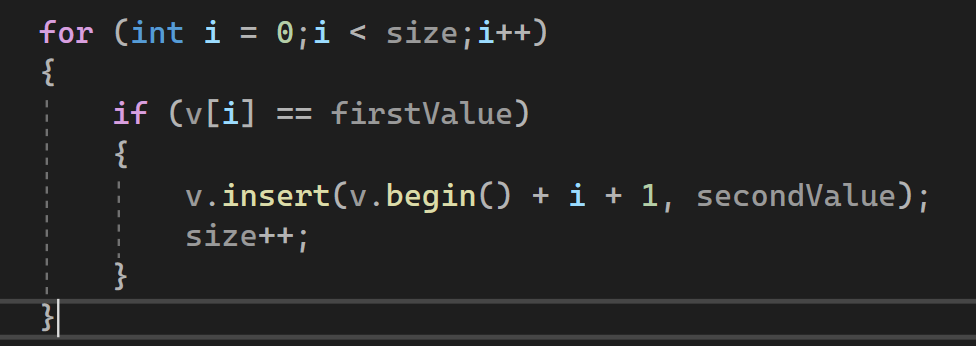
1. At first I tried to implement the insertAfter function using an iterator:



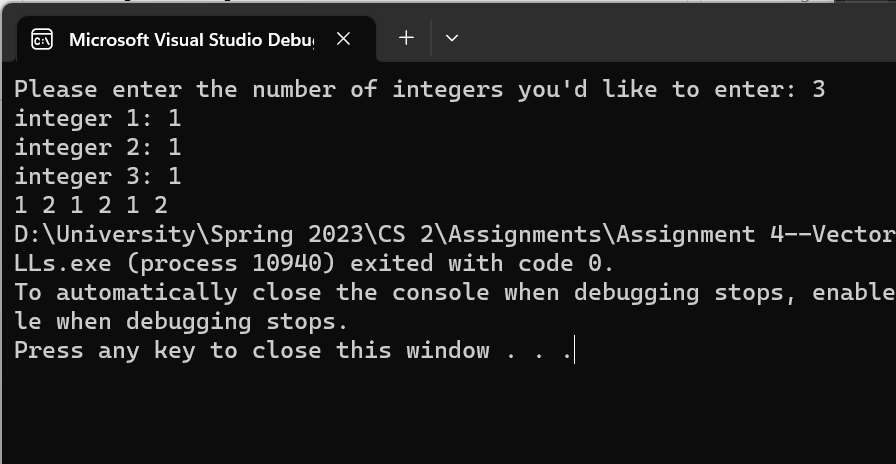
But for some reason it would insert the second value multiple times (I believe )



So I implemented it using the vector []operator:



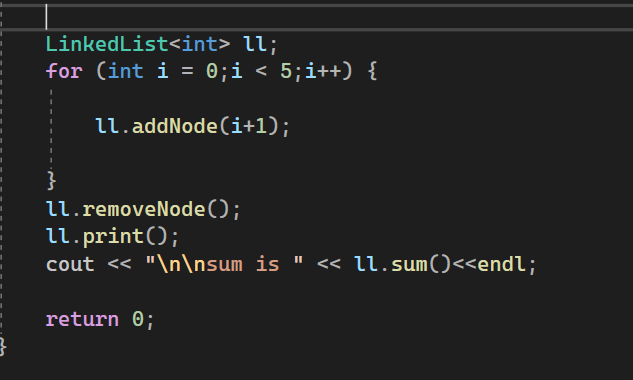
And it worked:



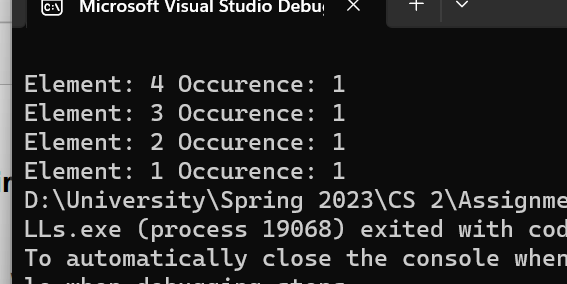
For the add and remove functions I did so from the beginning of the list to lower the time complexity (since the question did not specify that it had to be displayed in a specific order and I did not see the need to add a “last element” pointer)

I also decided to create a template class LinkedList and to send the data instead of sending the nodes. That is because in the “vector to linked list” constructor I needed to deal with dynamic allocation, so I needed to deallocate at the end. In order to avoid the issue of later adding static nodes to that list and being unable to deallocate it (apparently you cannot deallocate a static data member), I decided to allocate all nodes dynamically.

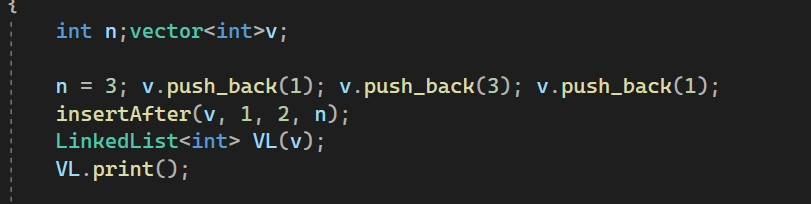
**Call function for add, remove and print**



**Output:**

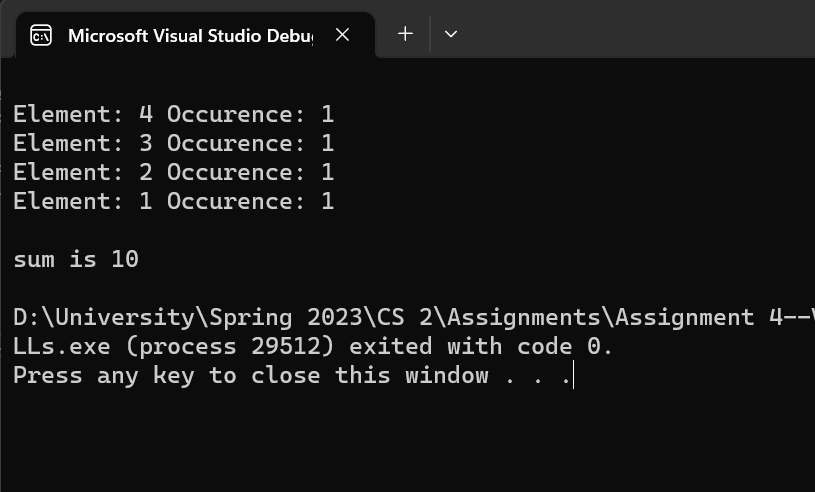


**Call function for vector to list constructor:**

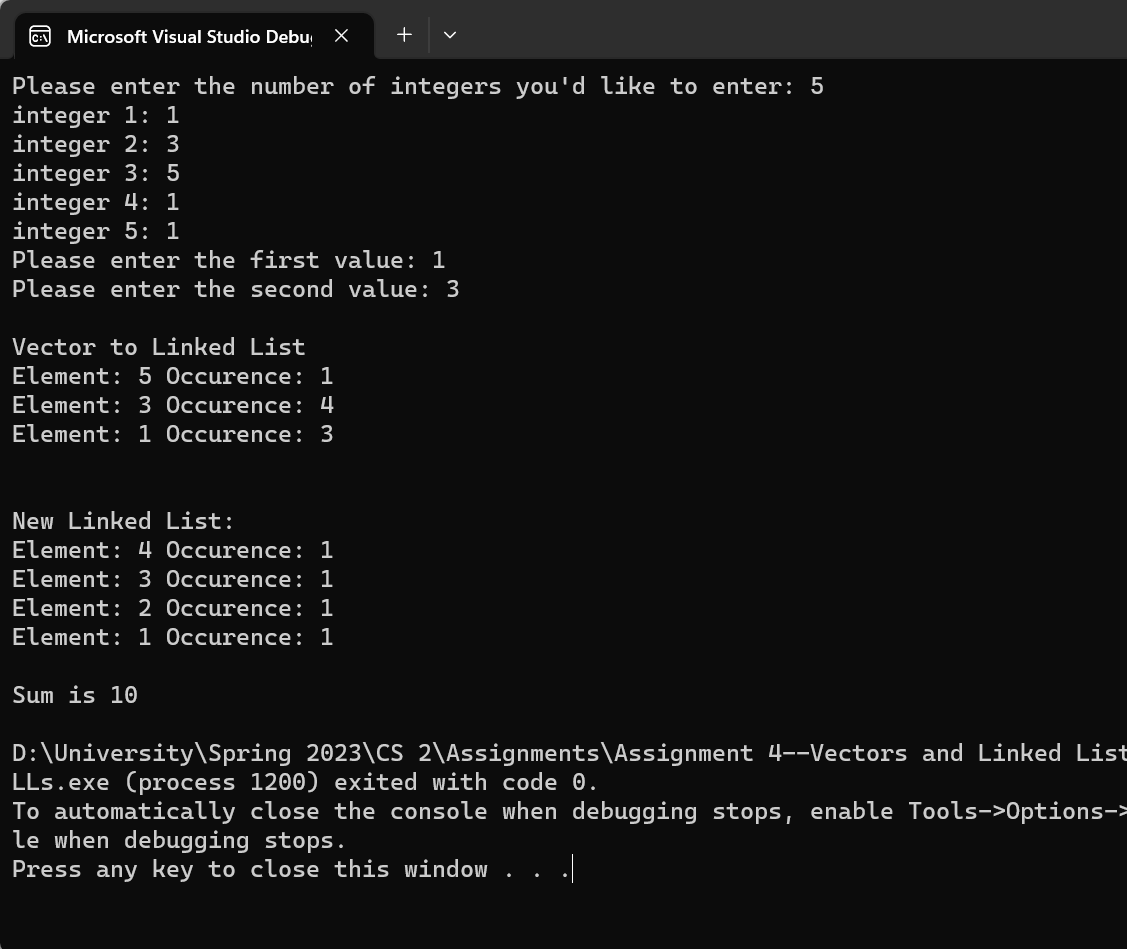
****

**Output:**

**Sum function output**



**Full Program Running:**

****