Problem Solving Steps

Program Assignment #5

**Problem #1**

* Make a linked list after making multiple nodes using my object Node from Node class which you can send the data and the address for the next element in your linked list by using the default constructor or using get data function which will give you the value of the data that been sent to the Node or get Next which will give you the address of the next element of that Node(where it points at), als nextNode function which you can set the address later if you didn’t set an address from the beggining(when you made an object)
* In my Code I made three differents nodes and each one hold a value
  + - Node 1 : Value 3
    - Node 2 : Value 5
    - Node 3 : Value 4
  + Then later I called the nextNode function to set an address for each node and by default it should be None/Null which means points at nothing.
  + Then I printed a list of the nodes values and addresss and as we can see in my code I made first node points at the second and the second one points at the third one and third one is still nodes
* Now we I finished the linked lists part as required and now I created up there a Stack class which do as required empty size top push and pop.
* Now I created a stack.
* Then I checked whather the stack is empty or not by using empty function
* Then I push my nodes values to the stack by using push
* Then I checked again if the stack is empty or not but now is not empty anymore.
* I did used the pop function which will return the value of the top of the stack.

**Problem #2**

* Reading the user input and store it into a var. called usrInput.
* Declared the left side parenthesis and started it from zero, same for the right side.
* Making a for loop to read through the string input.
  + If we catch a left side parenthesis we add one to our counter.
  + Same apply to our left.
* We see both total counter left and right and if the left is even and right is even by using a module then we have a duplication, as asked in the instructions.
* Last print the result as True or False and tested the given cases.

**Problem #3**

* What I did here I created a list as we can see in all cases we have the same list used over and over and the list is 1,10,3,5 and we have to get the window size which I asked the user to input and as explained is very clear that we need to use queue which works the best for me.
* I created a queue class and it has all the functions that required and needs enqueue, dequeue, rear, front, isEmpty, showQueue, size
* After I created a list and took the window size from the user and let’s say is 3 I created tempQueue to store my values from that previous list and also it will help me to dequeue so elements later and get the total. I created a total queue to store the total of each loop then I created a result queue to store my result.
* Then I created a for loop go through the given list 1, 10, 3 ,5 and it has two condition
  + If the size of my temporary queue is NOT equal to the window size
  + Then we’re going to add the given element[i] and enqueue to the tempQueue
  + Then add the value to my variable total(which now equal to 0) and first round or loop the total it will be 1 and the total will be enqueue to the totalsqueue
  + Second loop will do the same thing
    - The size of my temporary queue is 1 so we gonna go through the first condition
    - Value(10) enqueue into the tempQueue and then value(10)+=total which will be now (11) and it will enqueue in the totals queue again
  + Third loop will be in the same condition
    - Value 3 is added to the temp queue and then added the total and total is equal to (14)
  + Fourth round the size of the tempQueue passed the window size so we have to dequeu the first element and then remove the first value and subtract it from the total which total will be (18) and tempQueue will be size 3 again and it will containts (10,3,5)
* Now the we got the left side and now we gonna start calculate the devision which it will start from 1 then 2 then 3 and then 3 because is not supposed to cross the windowSize again which is 3
* List of the totals [ 1, 11, 14, 18]
* Now I created another for loop into the totals list and counter = 1 and loop has two conditions
  + Loop one if the counter is NOT bigger than window size then do the following
  + Result queue we have to enqueue x which is 1 in the first round and devide it by counter which is equal 1 and so on
  + So 1/1 = 1 > enque to resultsQueue
  + Then 11/2 > enque to resultsQueue
  + Then 14/3 > enque to resultsQueue
* Fourth round it will apply the second condition which the counter is bigger than window size which will do the following:
  + Then the counter is equal = to windowSize(3) again
  + And now x is equal to 18 so
  + 18/3 > enque to resultsQueue
* Last thing just printed out the results and that’s about it.