NATIONAL UNIVERSITY
OF COMPUTER & EMERGING SCIENCES
PESHAWAR CAMPUS

Problem Set:	Assignment: A06	**Semester**:	Spring 2019
Points:	*10*		
Date Set:	*See SLATE*	**Due Date**:	*See SLATE*
Course:	CS217 - OOP	**Instructor**:	Dr. Nauman

1   Summing Over a List

We've already created a **List** class. We are going to modify it slightly. The objective of this assignment is to create a list of integers and then write a function that can return the sum of the values in all the nodes of the list.

However, an important requirement for this assignment is that you can not use a loop to perform this summation! So, how do we do it? Here's a method for doing this:
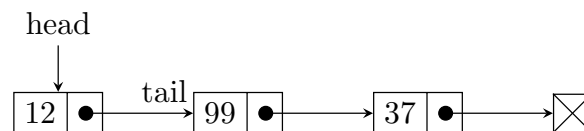
See the list above? Imagine that it is composed of two parts: the *head* and the *tail*. The head is the first node of the list and the tail is everything that comes after the head. Notice that the "rest of the list" i.e. the *tail* is also a list in itself!

Now, to sum over the list is very simple: take the value of the head and add to it *the sum of the tail*. One final point to note is that the sum of an empty list is simply 0.

1.1   Task

Once you understand this algorithm, you need to take the List class we created during lectures and add a public method to it called `sum_list`. This function will receive no parameters and will return the sum of the elements in the list. However, this function will not have the algorithm in its body. Instead, it should call a private function (`do_sum`) of the class and simply return the result produced by `do_sum`. This `do_sum` function should have the following signature:

```
int do_sum(node *current);
```

This private function do_sum should have the algorithm we described above.

2   Submission

After you're done with your class design, write down (**read: assignment must be hand-written**) the full code you've created and submit that on or before the deadline. Late assignments (or soft copies) will not be accepted.