Singly Linked List [CO3]

[Each method carries 5 marks]

Instructions for students:

- This assignment is worth **5*5 = 25 marks**. Viva on the Bonus task will be taken if it is done.

- Complete the following methods on Singly Linked List.

- You may use any language to complete the tasks.

- All your methods must be written in one single .java or .py or .pynb file. DO NOT CREATE separate files for each task.

- If you are using JAVA, you must include the main method as well which should test your other methods and print the outputs according to the tasks.

- If you are using PYTHON, then follow the coding templates shared in this folder.

NOTE:

- **YOU CANNOT USE ANY OTHER DATA STRUCTURE OTHER THAN LINKED LIST**

- **YOUR CODE SHOULD WORK FOR ANY VALID INPUTS. [Make changes to the Sample Inputs and check whether your program works correctly]**

- **LOOK OUT FOR 0th NODE Condition**

Tasks on Linked List

1. Number Beads

Amy Santiago and her son Mac were playing with number beads. Amy decided to make a chain with those beads. Being the way she is, Amy decided to place the numbers in the necklace in a descending order. For example, she placed the numbers in such a way: 20, 17, 13, 10, 6. But her spouse Jake wished to tease her. And so, he right rotated the number beads a number of times without showing Amy. Then he showed the rearranged necklace and told her to guess the number of times he rotated the necklace. Can you help Amy by writing a method that takes the rearranged number beads and returns the number of times the necklace was right rotated?

Sample Input	Sample Output	Explanation
13→10→6→20→17→None	3	The original sequence: 20→17→13→10→6 After Right Rotation 1: 6→20→17→13→10 After Right Rotation 2: 10→6→20→17→13 After Right Rotation 3: 13→10→6→20→17 Therefore, the necklace was rotated 3 times.
6→20→17→13→10→None	1	

2. Building Blocks

Your twin and you are under an experiment where the amount of thinking similarities you two have is being observed. As per the experiment, you are given the same number of building blocks of different colors and are told to make a building using those blocks in two different rooms.
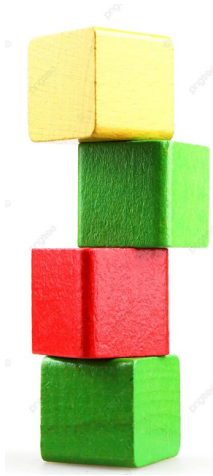
After the buildings are finished, the observers check whether the two buildings are the same based on the block colors. Now, you are the tech guy of that team and you are instructed to write a program that will output "Similar" or "Not Similar" given the two buildings. For fun, you decided to represent those buildings as a linked list!

NB: Red means a red block
Blue means a blue block
Yellow means a yellow block
Green means a green block.

Sample Input	Sample Output
building_1 = Red→Green→Yellow→Red→Blue→Green→None building_2 = Red→Green→Yellow→Red→Blue→Green→None	Similar
building_1 = Red→Green→Yellow→Red→Yellow→Green→None building_2 = Red→Green→Yellow→Red→Blue→Green→None	Not Similar
building_1 = Red→Green→Yellow→Red→Blue→Green→None building_2 = Red→Green→Yellow→Red→Blue→Green→Blue→None	Not Similar

3. Remove Compartment

Sheldon is a train maniac. He loves attaching each compartment of a train to the next with a linking chain as his hobby. Each train compartment is associated with a number. Now he is removing the last occurrence of a particular train compartment from the train. Can you model the scenario by writing a method that takes the compartment sequence and a number; the method returns the changed (or unchanged) train compartment sequence.

Constraint:

   a. You **cannot** create any new linked list. You have to change the given one.

Sample Input	Sample Output
14→10→15→10→41→10→72 number = 10	14→10→15→10→41→72
10→15→34→41→56→72 number = 7	10→15→34→41→56→72
33→15→34→41→56→72 number = 33	15→34→41→56→72

4. Capture the Flag

There is a competition called [CTF](#) (Capture the flag). It's basically a competition about cyber security where you will be provided a range of questions from different categories (from easy level to level extreme) and you have to find out the answers (aka flags) of those questions. [If you are interested, you can solve interesting problems from [here](#)] Suppose, you are playing a CTF with your friends where you are given a series of numbers as clues. If all the numbers are a multiple of its corresponding position **[1 based]**, then the flag will be **a newly created** linked list consisting of the factors of the numbers. If not, then the flag will be a string 'Linkwise'.

Sample Input:	Sample Output:	Explanation
11→8→21→20→5→42 →None	11→4→7→5→1→7 →None	11 is at position 1. Since 1*11 = 11, the node in resulting linked list has 11 in position 1 8 is at position 2. Since 2*4 = 8, the node in resulting linked list has 4 in position 2 21 is at position 3. Since 3*7 = 21, the node in resulting linked list has 7 in position 3
11→8→28→20→5→42 →None	Linkwise	28 is at position 3. Since 28%3 != 0, the result is a string Linkwise.

5. Shuffle on Song

In a Music player , the next button gives you the next song and the previous button moves back to the previous one. For simplicity, suppose our designed music player only has a next button. However, our music player has a special button named 'shuffle on song'. The button creates another playlist from the given playlist where all the songs with even ascii value will be linked in the beginning and the odd ascii valued songs are linked in the end. The relative position of the song remains unchanged. Implement the 'shuffle on song' button that takes a playlist and returns a **newly created** playlist. Given playlist will contain at least two songs.

Sample Input:	Sample Output:
S→E→N→P→A→I	N→P→S→E→A→I
N→I→S→H→I→N→O→Y→A	N→H→N→I→S→I→O→Y→A

BONUS

Assemble Conga Line

Have you ever heard the term conga line? Basically, it's a carnival dance where the dancers form a long line. Everyone holds the waist of the person in front of them and their waists are held in turn by the person to their rear, excepting only those in the front and the back. It kind of looks like this-

By now, you can quite understand the suitable data structure to represent a conga line. Now you are the choreographer of the Conga Dance in a Summer Festival. You have arranged an ascending conga line based on the participants' age. But you feel like this year's conga line is a bit short. You want to add one more person in a certain position in the conga line (maintaining the ascending order). There are other candidates willing to join the line. You have to pick a candidate from that pool to add in your conga line in the specified position **[0 based]**. Now as technical you are, can you write a method that will take the conga line, the candidate line and the insertion index and return the resulting conga line?

Constraint:

b. You **cannot** create any new linked list. You have to change the given one.
c. Pick a candidate whose age is closer to the person in the inserting index in the conga line

Sample Input	Sample Output	Explanation:
conga_line = 10→15→34→41→56→72 candidate_line = 16→2→36→52→40→77 →None insertion_idx = 3	conga_line = 10→15→34→40→41→56→72	To insert a candidate in the 3rd index in conga line, you need to pick someone within 34-41. The possible candidates are 36 and 40. Since, 40 is closer to 41, we choose 40 to be inserted in the 3rd index.