# **Department of Computer Science and Engineering, BRAC University**

**CSE 220 ( Data Structure ) for Summer 2023 Semester**

**Quiz 2**

**Set: A**

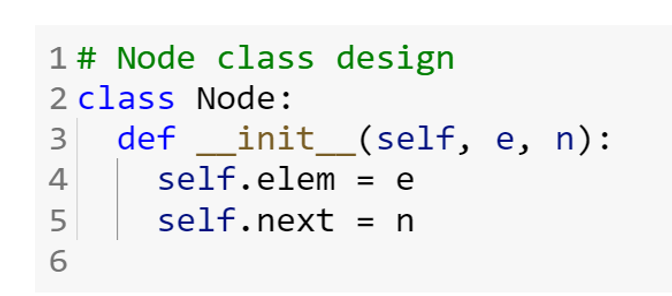
**Student ID: Full Marks:20**

**Name: Duration: 30 minutes**

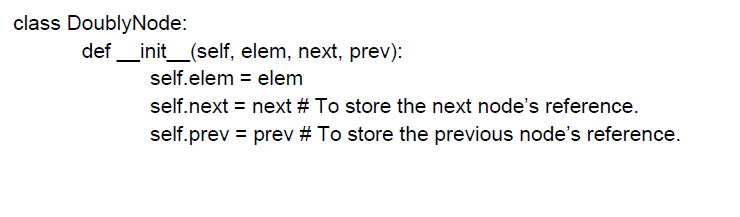
**Section:**

1. Write a function that converts a **Dummy-headed Doubly Circula**r linked list into a **Non Dummy-headed Singly Linear** linked list and returns the resultant list.

The singly node:



The doubly node:



The function looks like this:

def convert ( dh ):

#your code here **(10)**

1. Given an expression in infix format, prepare the corresponding **expression tree**. Then from there find out the **postfix** format. After that, **evaluate** the expression using a stack step by step.   
   ( 1 ✕ ( 2 + 3 ) - 4 ) / ( 2 ✕ 3 - (2 + 3✕1) ) **(3+3+4=10)**

# **Department of Computer Science and Engineering, BRAC University**

**CSE 220 ( Data Structure ) for Summer 2023 Semester**

**Quiz 2**

**Set: B**

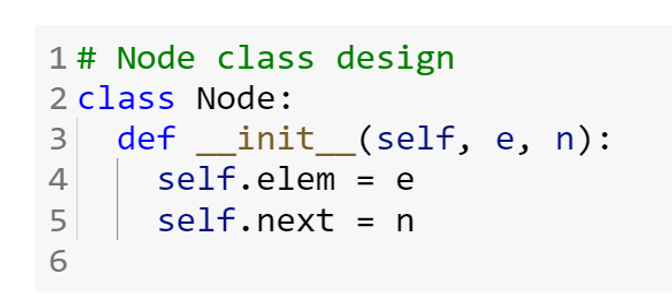
**Student ID: Full Marks:20**

**Name: Duration: 30 minutes**

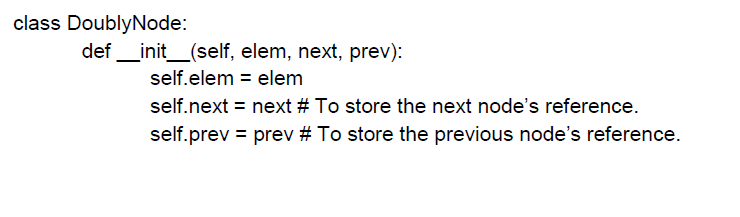
**Section:**

1. Write a function that converts a **Non Dummy-headed Singly Linear** linked list into a **Dummy-headed Doubly Circular** linked list and returns the resultant list.

The singly node:



The doubly node:



The function looks like this:

def convert ( head ):

#your code here **(10)**

1. Given an expression in infix format, prepare the corresponding **expression tree**. Then from there find out the **postfix** format. After that, **evaluate** the expression using a stack step by step.   
   ( 1 ✕ ( 2 + 4 ) - 3 ) / ( 3 ✕ 2 - (3 + 2✕1) ) **(3+3+4=10)**