Data Structure Lab (CSE 207)
Lab 8: Doubly Linked Lists

Submission Guidelines:

1. You cannot use any **readymade function** from the java data structure library for the lab assignment. The purpose of the course is to learn how to build the function provided in the library.
2. You are supposed to do this assignment on your own. While you may discuss the problem with other students, you are not allowed to copy any part of the code from other students or to copy from any other source. Any form of **plagiarism** will not be tolerated. If there is substantial overlap between the codes submitted by two students, both will get reduction in the course grade.
3. The project name should be your enrolment ID, e.g. '**E16CSEXXXlab8.java**'. Only submit **single java** file on LMS. Input file name should be 'input.txt' and output file name should be 'output.txt'.
4. Your code should be commented properly. Every method should be indicated.
5. The assignment should be **shown to lab instructor** in the next lab session and **must be submitted** on LMS by **Friday, Oct. 04, 2019.**

Every assignment submitted by the student would carry

- name of the student
- Enrolment number
- Batch number
- Mobile number
- Home address

Lab Assignment

In this assignment you will required to construct a sorted doubly linked list of integer elements with the **following operations**:

- Insertion: Insert an element in the linked list.
- Deletion: Delete an element
- Printing: Print the complete current list with left node address, current node address, current node value and right node address.
- For printing the list, the code will be 1. Your program should print the list in the format: left node address, current node address, current node value and right node address. (see sample output below).
- For insertion, the code will be 2, followed by number of data values to be inserted, followed by integer data to be inserted. Your program should insert the data in its proper place so that the list remains sorted.

- The code for deletion will be 3 followed by the integer to be deleted from the list. In case the integer is found in the list, your program should delete the node containing the integer. Otherwise, it should not do anything.

Sample Input

```
5
2  2 36  67
2 2 44  56
3  44
3 23
1
```

Sample output

The doubly linked list should be in the given format.

Previous Addr	Node	Curr Addr	Next Addr
Null	36	1008	1100
1008	56	1100	1200
1100	67	1200	Null

Syntax of class:

Class DLinkedlist

```
{
        Int data;
        DLinkedlist prev, next;
        node getprevlink(){}
        node getnextlink(){}
        setprevlink(Node){}
        setnextlink(Node){}
}
```

Ques. 2. Write a simple text editor, which stores and displays a string of characters using linked list, together with a cursor object that highlights a position in the string. The editor must support the following operations:

Left: Move cursor left one character (Do nothing if at the beginning.)

Right: Move cursor right one character (Do nothing if at the end.)

Insert c: Insert the character c just after the cursor.

Delete: delete the character just after the cursor (if not at end.)

Ques. 3. Write a program to multiply two given polynomials using linked list.

List 1 `3 2` → `5 1` → `6 0` → NULL

×

List 2 `6 1` → `8 0` → NULL

↓

Resultant List

`18 3` → `54 2` → `76 1` → `48 0`

↓

NULL

NODE
STRUCTURE

Coefficient	Power	Address of next pointer