Part A - coding

1 . Implement Doubly Linked List **get** method which behaves like the operator[] of array.

- It takes an integer parameter i as the index, it throw an Exception if the index i is illegal, returns the element at given index i, it traverse from the header of the list if index i is in the lower end of the list(less than half of the size), and traverse from the trailer of the list if index i is in the higher end of the list. Part of the code is given, you need to write the loops so that ansNode is referencing the correct node.

2 . Implement Polynomial Linked List **multiply** method which multiply two polynomial and returns the product as a new polynomial.

- It's an instance method which called by a polynomial and takes second polynomial as the parameter pl. It should correctly multiply two polynomials together and returns this product polynomial.

Extra Credit:

Change method **print**, so it print out the polynomial in standard form – highest degree to the lowest.

Submit

Please download the given doubly linked list https://venus.cs.qc.cuny.edu/~yang/fa19/cs313/Code/Review/DLinkedList.java and polynomial linked list https://venus.cs.qc.cuny.edu/~yang/fa19/cs313/Code/Review/PolynomialLinkedlist.java.

Change the file and class name to include your initials at the end. eg, if your name is John Smith, and submitting first method in class DLinkedList, please change the file to DLinkedListSJ and include your name at the very top of the program as comments.

Email both java file to csqc.yang@hotmail.com.

Write CS313 A1-DLL/PLL, your name, and section number (37-Tue/Thur 5:00 class, 38-Tue/Thur 6:30 class, 24-Fri 1:40 class) in the subject line of your email.