**Week 2**

1. **Read the degree of two polynomials and their coefficients, all integers, from the standard input. The polynomial is of the form ( ) = ∗ + ⋯ + 1 ∗ 1 +, where 0 ≠ 0. Write the pseudocode for adding two polynomials.**

Step 1: Start

Step 2: Let m and n be represented as two polynomials by the linked list.

Step 3: If m and n are not null, then repeat Step 2.

Step 4: If the powers of the two given terms are equal and if the terms are not canceled then insert the sum of the given terms into the sum polynomial.

Step 5: Advance m

Step 6: Advance n

Step 7: Else if the power of first given polynomial is greater than the power of second polynomial then,

Step 8: insert the term from first polynomial into sum polynomial

Step 9: Advance m

Step 10: Else insert the term from second polynomial into sum polynomial

Step 11: Advance n

Step 12: copy the remaining terms from non-empty polynomial into the sum polynomial.

Step 13: End

1. **Write the pseudocode and code for a function that determines whether given word is palindrome. What is the time complexity (expressed using BigO notation)?**

Step 1: Start

Step 2: Read word= “Madam”

Step 3: Find the reverse of the given word.

Step 4: If the reverse word is equal to given word then,

Step 5: print “Palindrome”

Step 5.1: else print “Not Palindrome”

Step 6: End

* The time complexity in this function is 0(n).