# Week 1

1. Method of reverse the characters from a string.

char[] rev = input.toCharArray();

for(int i=rev.length-1; i>=0; i--)

{

System.out.println(rev[i]);

}

1. Pseudocode for finding duplicate values in an array.

Step1: Start

Step2: Declare variable num, I and j

Step3: Read the values of num

Step4: Repeat until I is greater or equal to length of num

Step5: Repeat until j is greater or equal to length of num

Step5.1: if num[i]=num[j]

Output num

Step6: Stop

# Week 2

1. Pseudocode of adding two polynomials

Step 1: Start

Step 2: Declare the arrays a, b and sum

Step 3: Declare variables x, y and size

Step 4: Set the values of a to {10,12,15,30} and the values of b to {2,4,5,6,7}

Step 6: Display the first polynomial is

Step 7: for (int j=0;j<x;j++)

7.1: Output x[j]

Display x^ + j

Step 8: Display the second polynomial is

Step 9: for (int j=0;j<n;j++)

9.1: Output b[j]

9.2: if j!=0

Display x^ + 1

9.3: if j!=y-1

Display x^+1

Step 10: if x>y or x=y

Set the value of size to x.

10.1: else

Step 11: Set the length of array sum to size

Step 12: for (int j=0;j<x;j++)

12.1: sum[j]<-a[j]

Step 13: for (int j=0;j<n;j++)

13.1: sum[j] <-sum[j] + b[j]

Step 14: Display the sum is

Step 15: for (int j=0; j<size;j++)

15.1: Output sum [j]

15.2: if j!=0

Display x^ + i

15.3: if j!=size-1

Display +

Step 16: Stop

1. Pseudocode and code for a function that determines whether given word is palindrome.

* Pseudocode of determines whether given word is palindrome.

Step 1: Start

Step 1: data string, reverse as string

Step 2: data input as array of length

Step 3: data i as integer

Step 4: Read str

Step 5: char input [] = str.tocharArray()

Step 6: reverse = “ ”

Step 7: for (int i = (input.length - 1); i >= 0; i- -)

Step 7.1: reverse = reverse + input [i]

Step 8: if (reverse.equals(str))

Step 8.1: output “The string is Palindrome”

Step 8.2: else

Step 8.3: output “The string is not Palindrome”

Step 9: Stop

* Function for determining whether given word is palindrome.

String input, b = "";

Scanner s = new Scanner(System.in);

System.out.print("Enter the string: ");

input = s.nextLine();

int n = input.length();

for(int i = n - 1; i >= 0; i--)

{

b = b + input.charAt(i);

}

if(input.equalsIgnoreCase(b))

{

System.out.println("The string is palindrome.");

}

else

{

System.out.println("The string is not palindrome.");

}

* Time Complexity:

Time complexity is a way of showing how the runtime of a function increase as the size of the input increase.

# Week 3