Name : Yash Jain
Enrolment No. : 0818CSS221230
CSE – 3, III Year.

Indore Institute of Science & Technology
Department of Computer Science & Allied Department

LIST OF Programs
Week -1
Algolution

1. Write a program to count total number of negative numbers in an array.

```
public class Main
{
   static int countNegativeNum(int[] arr){
      int count = 0;
      for(int i=0; i<arr.length; i++){
         if(arr[i] < 0){
            count++;
         }
      }

      return count;
   }

         public static void main(String[] args) {
                  int[] arr = {1,4,7,-8,-4,8,3,-9,6,-4,-2,-8,9,6,5,-2};
                  System.out.println("Total Count of Negative Numbers in Array is :
"+countNegativeNum(arr));
         }
}
```

2. Write a program that takes 5 integers as input from the user and print Max of all the numbers.

```
import java.util.Scanner;
```

```
public class Main
{
   static int maxInArray(int[] arr){
      int max = Integer.MIN_VALUE;

      for(int i=0; i<arr.length; i++){
         max = Math.max(max, arr[i]);
      }

      return max;
   }

         public static void main(String[] args) {
                  Scanner sc = new Scanner(System.in);
                  System.out.print("Enter Size of Array : ");
                  int n = sc.nextInt();

                  int arr[] = new int[n];

                  for(int i=0; i<n; i++){
                     System.out.print("\nEnter "+(i + 1)+" element : ");
                     arr[i] = sc.nextInt();
                  }

                  System.out.print("\nMaximum Element in an array is :
"+maxInArray(arr));
         }
}
```

3. Write a program to merge two sorted arrays to a third array.

```
import java.util.Scanner;

public class Main
{
   static int[] mergeArray(int[] arr1, int[] arr2){
      int idx1 = 0, idx2 = 0;
```

```
    int mergedArray[] = new int[arr1.length + arr2.length];
    int idx = 0;

    while(idx1 < arr1.length && idx2 < arr2.length){
       if(arr1[idx1] < arr2[idx2]){
          mergedArray[idx++] = arr1[idx1++];
       }
       else{
          mergedArray[idx++] = arr2[idx2++];
       }
    }

    while(idx1 < arr1.length){
       mergedArray[idx++] = arr1[idx1++];
    }

    while(idx2 < arr2.length){
       mergedArray[idx++] = arr2[idx2++];
    }

    return mergedArray;
  }

      public static void main(String[] args) {
            int[] arr1 = {1,3,5};
            int[] arr2 = {2,4,6};

            int arr[] = mergeArray(arr1, arr2);

            System.out.print("Final Array After Merging : ");

            for(int i=0; i< arr.length; i++){
               System.out.print(arr[i]+" ");
            }
      }
}
```

4. Write a program that takes 7 integers as input from the user then Reverse the order of numbers in the array, then print the numbers.

```
import java.util.Scanner;

public class Main
{
   static int[] reverseArray(int[] arr){
      for(int i=0; i<arr.length/2; i++){
         int temp = arr[i];
         arr[i] = arr[arr.length - 1 - i];
         arr[arr.length - 1 - i] = temp;
      }

      return arr;
   }

         public static void main(String[] args) {
                  int arr[] = new int[7];

                  Scanner sc = new Scanner(System.in);

                  for(int i=0; i<arr.length; i++){
                     System.out.print("Enter "+(i + 1)+" element : ");
                     arr[i] = sc.nextInt();
                  }

                  int reverseArray[] = reverseArray(arr);

                  System.out.print("\nArray after reversing : ");

                  for(int i=0; i<reverseArray.length; i++){
                     System.out.print(arr[i]+" ");
                  }
         }
}
```

5. Write a program that takes 5 integers as input from the user and finds out if the order of numbers in array is palindrome
   e.g.
   if input numbers are 44 56 56 44 then the array is in palindrome order

if input numbers are 44 53 156 44 then the array is not in palindrome order.

```
import java.util.Scanner;

public class Main
{
   static boolean isPalindrome(int[] arr){
      for(int i=0; i<arr.length/2; i++){
         if(arr[i] != arr[arr.length - i - 1]) return false;
      }

      return true;
   }

        public static void main(String[] args) {
                int arr[] = new int[5];

                Scanner sc = new Scanner(System.in);

                for(int i=0; i<arr.length; i++){
                   System.out.print("Enter "+(i + 1)+" element : ");
                   arr[i] = sc.nextInt();
                }

                if(isPalindrome(arr)) System.out.print("\nArray is palindrome");
                else System.out.print("\nNot a palindrome");
        }
}
```

6. Write a program to search a number (taken as input from user) in an array then print the index of the first occurrence of the input number in the array
   If the input number is not present in the array then print -1

```
import java.util.Scanner;
```

```
public class Main
{
   static int firstOccurence(int[] arr, int target){
      for(int i=0; i<arr.length; i++){
         if(arr[i] == target){
            return i;
         }
      }

      return -1;
   }

         public static void main(String[] args) {
                  int arr[] = {4,6,5,8,7,6,3,9,1,2,5,4,3,6,5,7,8,4};

                  Scanner sc = new Scanner(System.in);

                  System.out.print("Enter number to find : ");
                  int numToFind = sc.nextInt();

                  int idx = firstOccurence(arr, numToFind);

                  if(idx > -1 ) System.out.print("Element found at "+idx+" index");
                  else System.out.print("Element Not Found!!");
         }
}
```

7. Write a program to print all unique numbers in an array.

```
import java.util.Scanner;

public class Main
{
   static int uniqueElements(int[] arr){
      int max = Integer.MIN_VALUE;
      for(int i=0; i<arr.length; i++){
         max = Math.max(max, arr[i]);
      }
```

```
      int count[] = new int[max+1];

      for(int i=0; i<arr.length; i++){
         count[arr[i]]++;
      }

      System.out.print("Unique Elements in array : ");
      for(int i=0; i<count.length; i++){
         if(count[i] == 1){
            System.out.print(i+" ");
         }
      }

      return -1;
   }

         public static void main(String[] args) {
                  int arr[] = {4,6,5,8,7,6,3,9,1,2,5,4,3,6,5,7,8,4};

                  uniqueElements(arr);
         }
}
```

8. Write a program to count frequency of each number in an array.

```
import java.util.Scanner;

public class Main
{
   static int frequencyOfElements(int[] arr){
      int max = Integer.MIN_VALUE;
      for(int i=0; i<arr.length; i++){
         max = Math.max(max, arr[i]);
      }

      int count[] = new int[max+1];
```

```
      for(int i=0; i<arr.length; i++){
         count[arr[i]]++;
      }

      for(int i=0; i<count.length; i++){
         System.out.println("Frequency of element "+i+" in an array is :
"+count[i]);
      }

      return -1;
   }

         public static void main(String[] args) {
                  int arr[] = {4,6,5,8,7,6,3,9,1,2,5,4,3,6,5,7,8,4};

                  frequencyOfElements(arr);
         }
}
```

9. Write a program to count total number of duplicate numbers in an array.

```
import java.util.Scanner;

public class Main
{
   static int countOfDuplicateElements(int[] arr){
      int max = Integer.MIN_VALUE;
      for(int i=0; i<arr.length; i++){
         max = Math.max(max, arr[i]);
      }

      int count[] = new int[max+1];

      for(int i=0; i<arr.length; i++){
         count[arr[i]]++;
      }

      int duplicateCount = 0;
```

```
      for(int i=0; i<count.length; i++){
         if(count[i] > 1) duplicateCount++;
      }

      return duplicateCount;
   }

         public static void main(String[] args) {
                int arr[] = {4,6,5,8,7,6,3,9,1,2,5,4,3,6,5,7,8,4};

                System.out.println("Count of duplicate element in array : "+
countOfDuplicateElements(arr));
         }
}
```

10. write a function in C to print following pattern

    Note: Number of rows and columns shall be the input parameters to the function

    Parallelogram

```
    *****
   *****
  *****
 *****
*****
```

```
import java.util.Scanner;

public class Main
{
   static void printParallelogram(int size){
      for(int i = 0; i < size; i++){
         for(int j=size - 1 - i; j > 0; j--){
            System.out.print(" ");
         }
         for(int j = 0; j < size; j++){
            System.out.print("*");
```

```
        }
        System.out.println();
      }
   }

        public static void main(String[] args) {
                printParallelogram(5);
        }
}
```

11. write a function in C to print following pattern

    Note: Only number of rows shall be the input parameter to the function

    Right triangle

    *
    **
    ***
    ****
    *****

```
import java.util.Scanner;

public class Main
{
   static void printTriangle(int size){
      for(int i = 0; i < size; i++){
         for(int j = i; j >= 0; j--){
            System.out.print("*");
         }
         System.out.println();
      }
   }

        public static void main(String[] args) {
                printTriangle(5);
        }
```

```
}
```

12. write a function in C to print following pattern

    Note: Only number of columns will be the input parameter to the function

    Half diamond

```
*
**
***
****
*****
****
***
**
*
```

```
import java.util.Scanner;

public class Main
{
   static void printHalfDiamond (int size){
      for(int i = 0; i < size; i++){
         for(int j = i; j >= 0; j--){
            System.out.print("*");
         }
         System.out.println();
      }
      for(int i = 1; i < size; i++){
         for(int j = i; j < size; j++){
            System.out.print("*");
         }
         System.out.println();
      }
   }

         public static void main(String[] args) {
                  printHalfDiamond(5);
```

```
        }
}
```

13. write a function in C to print following pattern

    Note: Only odd number of rows will be the input parameter to the function. If the number of rows is even then print an error and return false else return true.

    Diamond

```
      *
     ***
    *****
   *******
  *********
   *******
    *****
     ***
      *
```

```
import java.util.Scanner;

public class Main
{
   static void printDiamond (int size){
      for(int i = 1; i <= size; i++){
         for(int j = size - i; j > 0; j--){
            System.out.print(" ");
         }

         for(int j = 0; j < 2 * i - 1; j++){
            System.out.print("*");
         }

         System.out.println();
      }

      for(int i = size-1; i > 0; i--){
```

```
        for(int j = i; j < size; j++){
           System.out.print(" ");
        }
        for(int j = 2 * i - 1; j > 0 ; j--){
           System.out.print("*");
        }
        System.out.println();
     }
  }

        public static void main(String[] args) {
                printDiamond(5);
        }
}
```

14. write a function in C to print following pattern

    Note: Number of rows and columns shall be the input parameters to the function

    12345
    54321
    12345
    54321
    12345

```
import java.util.Scanner;

public class Main
{
   static void printPattern(int row, int col){
      for(int i = 0; i < row; i++){
         if(i % 2 == 0){
            for(int j = 1; j <= col; j++){
               System.out.print(j);
            }
            System.out.println();
         }
```

```
        else{
           for(int j = col; j > 0; j--){
              System.out.print(j);
           }
           System.out.println();
        }
     }
  }

        public static void main(String[] args) {
                printPattern(5,5);
        }
}
```

15. write a function in C to print following pattern

    Note: Number of rows and columns shall be the input parameters to the function

    11111
    10001
    10001
    10001
    11111

```
import java.util.Scanner;

public class Main
{
   static void printPattern(int row, int col){
      for(int i = 0; i < row; i++){
         for(int j = 0; j < col; j++){
            if(i == 0 || i == row-1 || j == 0 || j == col-1){
               System.out.print(1);
            }
            else{
               System.out.print(0);
            }
```

```
        }
        System.out.println();
      }
   }

        public static void main(String[] args) {
                printPattern(5,5);
        }
}
```

16. write a function in C to print following pattern

    Note: number of rows shall be the input parameter to the function.

    12345
    23455
    34555
    45555
    55555

```
import java.util.Scanner;

public class Main
{
   static void printPattern(int row, int col){
      for(int i = 1; i <= row; i++){
         int count = 0;
         for(int j = i; j <= col; j++){
            System.out.print(j);
            count++;
         }
         while(count < col){
            System.out.print(col);
            count++;
         }
         System.out.println();
      }
   }

        public static void main(String[] args) {
```

```
                printPattern(5,5);
        }
}
```
