04_Rajadip_Chavada

MCA 2

Subject : JAVA

Assignment 2

```
import java.util.*;
import java.util.Scanner;
import java.util.Arrays;
import java.util.StringTokenizer;
import java.lang.StringBuffer;

/*
```

1) Write application that creates an array of double, to provide following functionality.

- display the length of the array and its elements.

- Display an array. (Use for each version of loop for display).

- compute the sume of the squares of these numbers.

- Determine Mean and Median of an array.

- Sort an array – Ascending and Descending. Use any two sorting algorithm. User can also select the sorting method.

- Search an element from the array, i.e. returns the location of the element of an array that matches an indicated value.

- Copy of an array.

- Reverse of an array.

```
*/

class p1
{
   static void sumOfSquares(double[] arr,int n)
   {
      double avg=0,sum=0;
      for(double i : arr)
      {
         avg +=i;
      }
      avg = avg /n;
      //System.out.println(avg);
      for(double j : arr)
      {
                        //System.out.println(Math.pow((j-avg),2));
         sum +=Math.pow((j-avg),2);
      }
      System.out.println("Sum of Squares = "+sum);
      System.out.println("Mean of Array : "+avg);

      bubbleSort(arr,n);
      //Median of an Array
      double m=0;
```

```
    if(n%2==1)
    {
        m=arr[(n+1)/2-1];
    }
    else
    {
        m=(arr[n/2-1]+arr[n/2])/2;
    }

    System.out.println("Median :"+m);

}

static void bubbleSort(double[] arr,int n)
{
    double temp;
    for (int i = 0; i < n-1; i++)
    {   for (int j = 0; j < n-i-1; j++)
        {
            if (arr[j] > arr[j+1])
            {
                temp = arr[j];
                arr[j] = arr[j+1];
                arr[j+1] = temp;
```

```
        }
      }
   }

}
    static void selectionSort(double[] arr,int n)
{
   int min ;
   for(int i=0;i< n-1;i++)
   {
      min = i;
      for(int j=i+1;j<n;j++)
      {
         if(arr[j] < arr[min])
            min = j;
      }

      double temp= arr[min];
      arr[min] = arr[i];
      arr[i] =temp;
   }
   printArray(arr);
}
```

```
static void printArray(double[] arr)
{
   System.out.println("Array Elements are ::  ");

   for(double i : arr)
   {
      System.out.println(i);
   }
}

static int searchElement(double[] arr,int n,double key)
{
   for(int i=0;i<n;i++)
   {
      if(arr[i] == key)
      {
         return i;
      }
   }
   return 0;
}
static void copyArray(double[] arr,int n)
{
   System.out.println("Copy Array using 3 Method : ");
```

```
    double[] arrB = new double[n];
    double[] arrC = new double[n];
    //copy location of array to another array
    arrB = arr;
    printArray(arrB);

    for (int i = 0; i < n; i++)
        arrC[i] = arr[i];

    printArray(arrC);

    //array copy using clone
    double[] arrD = arr.clone();
    printArray(arrD);
}

static void reverseArray(double[] arr,int n)
{
    System.out.println("Reverse Array using ");
    double temp;
    for(int i=0;i<n/2;i++)
    {
        temp = arr[i];
        arr[i] = arr[n-i-1];
```

```
            arr[n-i-1] = temp;
        }
        printArray(arr);
    }
    public static void main(String[] args)
    {
        int n;
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the number of elements u want to
store : ");
        n = sc.nextInt();

        double arr[] = new double[n];

        System.out.println("Enter Element of the array");

        for(int i=0;i<n;i++)
        {
            arr[i]=sc.nextDouble();
        }

        System.out.println("Length of the array : "+ n);

        printArray(arr);
```

```
sumOfSquares(arr,n);
System.out.println();

int choice;
System.out.println("Choose Any one algorithm for sorting");
System.out.println("1.Bubble sort");
System.out.println("2.Selection Sort");
choice = sc.nextInt();

if(choice == 1)
{
   bubbleSort(arr,n);
   printArray(arr);
}
else if(choice == 2)
{
   selectionSort(arr,n);
}
else{
   System.out.println("select between 1 | 2 ");
}
```

```
        //Searching Elemetn
        int loc;
        double key=0;
        System.out.println("Enter Key Value you want to find : ");
        key = sc.nextDouble();
        loc = searchElement(arr,n,key);
        System.out.println("Location is : "+ loc + " value is : " +
arr[loc]+"\n\n");


        //Copy elements of array from one array  to another array
        copyArray(arr,n);


        System.out.println();


        reverseArray(arr,n);
    }
}


/*
```

2) Write a program in Java to create variable size array. Data can be numeric.

- Flush an array
- add a number at a specified location in an anay.
- display the array

```
*/

class p2
{
   static void printArray(int[] arr,int n)
   {
      System.out.println("Array is : ");
      for(int i=0;i<n;i++)
      {
         System.out.print(arr[i]+ " ");
      }
   }
   public static void main(String[] args)
   {
      Scanner sc = new Scanner(System.in);
      int n;
      System.out.println("Enter how many numbers add in array : ");
      n = sc.nextInt();
      int[] arr = new int[n];
      for(int i=0;i<n;i++)
      {
         arr[i] = sc.nextInt();
      }
      printArray(arr,n);
```

```
        int key,val;
        System.out.println("\nIndex No: ");
        key = sc.nextInt();
        System.out.println("\nEnter value for Index : ");
        val = sc.nextInt();
        if(key < n)
        {
           arr[key] = val;
        }
        else{
           System.out.println("Index is greater than range");
        }
        printArray(arr,n);
        Arrays.fill(arr,0);
        System.out.println("\nArray flushed suceessfully \n" );
    }
}

/*
3) Program in Java to find A+B, A-B, A*B and transpose of A, where A is a matrix of 3*3 and B is a matrix of 3*4.
Take the values in matrixes A and B from the user.
*/
```

```
class p3
{
   static void matrixMulti(int[][] A,int[][] B,int row1,int row2,int col2)
   {
      int c[][] = new int[row1][col2];
      System.out.println("MAtrix Multiplication : ");
       for(int i=0;i<row1;i++){
         for(int j=0;j<col2;j++)
         {
            c[i][j]=0;
            for(int k=0;k<row2;k++)
            {
               c[i][j]+=A[i][k]*B[k][j];
            }
            System.out.print(c[i][j]+" ");
         }
         System.out.println();
      }
   }
   static void summationMatrix(int[][] A,int[][] B ,int row1,int col1)
   {
      int c[][] = new int[row1][col1];
       for(int i=0;i<row1;i++)
```

```
    {
      for(int j=0;j<col1;j++)
      {
         c[i][j] +=A[i][j] + B[i][j];
      }
    }
    System.out.println("Matrix Addition A+B :");
    printMatrix(c,row1,col1);
}
static void substractionMatrix(int[][] A,int[][] B ,int row1,int col1)
{
   int c[][] = new int[row1][col1];
    for(int i=0;i<row1;i++)
    {
      for(int j=0;j<col1;j++)
      {
         c[i][j] +=A[i][j] - B[i][j];
      }
    }
    System.out.println("Matrix Substraction A-B :");
    printMatrix(c,row1,col1);
}
static void printMatrix(int[][] arr,int row1,int col2)
{
```

```
    System.out.println("MAtrix ::  ");
    for(int i=0;i<row1;i++)
    {
        for(int j=0;j<col2;j++)
        {
            System.out.print(arr[i][j] + " ");
        }
        System.out.println();
    }
}
public static void main(String[] args)
{
    /* A+B A-B A*B*/
    Scanner sc = new Scanner(System.in);
    int row1,col1,row2, col2;
    System.out.println("Enter First Matrix Size Row & column");
    row1 = sc.nextInt();
    col1 = sc.nextInt();
    System.out.println("Enter Second Matrix Size Row & column");
    row2 = sc.nextInt();
    col2 = sc.nextInt();

    int[][] A = new int[row1][col1];
    int[][] B = new int[row2][col2];
```

```
System.out.println("Enter First Matrix Element Row Wise : ");
for(int i=0;i<row1;i++)
{
   for(int j=0;j<col1;j++)
   {
      A[i][j] = sc.nextInt();
   }
}
System.out.println("Enter Second Matrix Element Row wise : ");
for(int i=0;i<row2;i++)
{
   for(int j=0;j<col2;j++)
   {
      B[i][j] = sc.nextInt();
   }
}
//printMatrix(A,row1,col1);
//printMatrix(B,row2,col2);

System.out.println();
if(row1 == col1 && row2 == col2)
{
   summationMatrix(A,B,row1,col1);
```

```
            substractionMatrix(A,B,row1,col1);
            System.out.println("Matrix multiplication is not possible bcoz its a Square matrix");
        }
        else
        {
            System.out.println("Addtion matrix and substraction of matrix Requires Square Matrix ");
            if(row2 == col1)
                matrixMulti(A,B,row1,row2,col2);
            else
                System.out.println("Matrix multiplication is not possible");
        }
    }
}
```

/*

4) Write and run a JAVA program that reads a string from the user and perform the following.

- counts number of occurance of a given character (for example, " a") in a string.

- searches the last occurance of a character in a string.

- removes the unneccessary spaces from a string : leading and trailing spaces.

```
- displays the substring formed by the last ten characters of a string
*/


class p4
{
   public static void main(String[] args)
   {
      Scanner sc = new Scanner(System.in);
      System.out.println("Enter String : ");
      String str;
      str = sc.nextLine();
      char key;
       System.out.println("Enter key char for searching : ");
      key = sc.next().charAt(0);
      //System.out.println(str);
      char[] ch = str.toCharArray();
      int count=0;
      for(char i: ch)
      {
         if(i == key)
         {
            count++;
         }
      }
```

```
        System.out.println("'"+key +"' number of Occurance is : " +
count);


        //searches last occurance of char in string
        System.out.println("last char = " + str.charAt(str.length()- 1));


        //remove extra space from string
        System.out.println(str.replaceAll("\\s+"," ").trim());


        //substring
        String last10Digit = " ";
        last10Digit =  str.substring(str.length() - 10);


        System.out.println(last10Digit);
    }


}


/*
5) WAP that inputs a line of text, tokenizes the line with
StringTokenizer and outputs the tokens in reverse order.
*/
```

```
class p5
{
   public static void main(String[] args)
   {
      Scanner sc =new Scanner(System.in);

      String str;
      str = sc.nextLine();

      System.out.println("Original String : "+ str);

      StringTokenizer st = new StringTokenizer(str);
      String strRev= " ";
      while(st.hasMoreTokens())
      {
         strRev = st.nextToken() + " " + strRev;
      }

      System.out.println("Reverse String : " + strRev);
   }
}

/*
```

6) Create a StringBuffer and illustrate how to append character. Display capacity, length of the StringBuffer.

*/

```
class p6
{
   public static void main(String[] args)
   {
      Scanner sc = new Scanner(System.in);
      String str = " ";

      str = sc.nextLine();
      StringBuffer sb = new StringBuffer(str);
      System.out.println("Enter character You want to append : ");
      char newStr = sc.next().charAt(0);
      sb.append(newStr);
      System.out.println(sb);
      System.out.println(sb.capacity());
      System.out.println("Length of string : "+sb.length());
   }
}
```

/*

7) Write an application that reads and processes strings from the console. Perform the following functions based on the menu choice selected by the user..

- Reverse the sequence of strings and then display it.

- Reverse the sequence of characters in each string and then display it.

- rearrange the strings according to the length of the string.

- Sorting

- Concatenation

- Change them to uppercase, lowercase depending on user's choice.

*/

```
class p7
{
   static Scanner sc =new Scanner(System.in);

   static void reverseSeqOfString(String str)
   {
      String s[] = str.split(" ");
      String seqStr = " ";

      for(int i=s.length-1;i>=0;i--)
         seqStr += s[i] + " ";
```

```
        System.out.println("Reverse the sequence of strings\n" + seqStr
+"\n");
    }
    static void reverseSeqOfCharEachString(String str)
    {


        String s[] = str.split(" ");
        String seqStr = " ";
        String tmp;
        for(String w:s)
        {
            StringBuilder sb=new StringBuilder(w);
            sb.reverse();
            seqStr += sb.toString() + " ";
        }
        System.out.println("Reverse the sequence of character of each
strings\n" + seqStr+"\n");
     }
    static void rearrangeString(String str)
    {
        String[] ch = str.split(" ");
       /* for(int i=1;i < ch.length;i++)
        {
            String temp = ch[i];
```

```
        int j = i-1;
        while( j >= 0 && temp.length() < ch[j].length() )
        {
            ch[j+1] = ch[j];
            j--;
        }
        ch[j+1] = temp;
    }*/
    System.out.println(ch.toString());
    System.out.println("Rearrange String : "+str);
}
static void sortingString(String str)
{
    char sortStr[] = str.toCharArray();
    Arrays.sort(sortStr);
    System.out.println(sortStr);
}
static void ConcatenationString(String str)
{
    String newStr =" " ;
    System.out.println("Enter string u want to append");
    newStr = sc.nextLine();
    String concateStr = str.concat(newStr);
    System.out.println(concateStr);
```

```
    }
    static void upperCaseString(String str)
    {
        System.out.println("UpperCase : "+str.toUpperCase());
    }
     static void lowerCaseString(String str)
    {
        System.out.println("LowerCase "+str.toLowerCase());
    }

    public static void main(String[] args)
    {
        System.out.println("Enter Your String : ");
        String str=sc.nextLine();
        char ch = ' ';
        int choice ;

        do
        {
        System.out.println("1.Reverse the Sequence of string ");
        System.out.println("2.Reverse the sequence of characters in
each string ");
        System.out.println("3.rearrange the strings according to the
length of the string");
```

```
System.out.println("4.Sorting");
System.out.println("5.Concatenation");
System.out.println("6.String Convert in UpperCase");
System.out.println("7.String convert in lowerCase");

   choice = sc.nextInt();
   switch(choice)
   {
   case 1:
      reverseSeqOfString(str);
      break;
   case 2:
      reverseSeqOfCharEachString(str);
      break;
   case 3:
      rearrangeString(str);
      break;
   case 4:
      sortingString(str);
      break;
   case 5:
      ConcatenationString(str);
      break;
   case 6:
```

```
            upperCaseString(str);
            break;
        case 7:
            lowerCaseString(str);;
            break;
        case 8:
            System.exit(0);
        default:
            System.out.println("Enter Valid choice");
        }
      System.out.println("\n Do you want to continue? (Press y/n)");
      ch = sc.next().charAt(0);
      }while(ch == 'y');
   }
}
```