

17 August 2020

Aayush Shah

19BCE245

Practical 5

OOP Lab

Practical 5 A

That allows you to create an integer array of 18 elements with the following values: $\text{int } A[] = \{3, 2, 4, 5, 6, 4, 5, 7, 3, 2, 3, 4, 7, 1, 2, 0, 0, 0\}$. The program computes the sum of element 0 to 14 and stores it at element 15, computes the average and stores it at element 16 and identifies the smallest value from the array and stores it at element 17.

CODE

```
import java.util.Scanner;
class Prac5a {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter the number of elements : ");
        int n = sc.nextInt();          //user input starts
        int arr[] = new int[n+3];
        int sum = 0;
        int min=0;
        for(int i=0;i<n;i++){          //user input along with calculation for the final
answer to reduce the number of for loops
            System.out.print("element no." + (i+1) + " : ");
            arr[i] = sc.nextInt();
            sum+=arr[i];
            if(i==0)                  //assigning the first element of array to min variable
to compare it after with remaining element and finding the smallest number among
all
                min=arr[i];
            if(min>arr[i])            //min checking condition
                min=arr[i];
        }
        arr[n] = sum;                 //sum of all array element
        arr[n+1] = sum/n;             //average of all array element
    }
}
```

```
arr[n+2] = min;          //smallest number among all
System.out.println("Sum of all elements is : " + arr[n] + ".");
System.out.println("Average of all elements is : " + arr[n+1] + "." +
" (actual float value : " + ((float)sum/(float)n) + ")");
System.out.println("Smallest value from the array is : " + arr[n+2] + ".");//
as there is int type of array , we cannot get actual value for average if the answer of
average if in float so, we can give it as separate value without changing the problem
statement.

    }
}
```

INPUT :

```
15
3
2
4
5
6
4
5
7
3
2
3
4
7
1
2
```

OUTPUT :

```
Enter the number of elements : 15
element no.1 : 3
element no.2 : 2
element no.3 : 4
element no.4 : 5
element no.5 : 6
element no.6 : 4
element no.7 : 5
element no.8 : 7
element no.9 : 3
element no.10 : 2
element no.11 : 3
element no.12 : 4
element no.13 : 7
element no.14 : 1
element no.15 : 2
Sum of all elements is : 58.
Average of all elements is : 3. (actual float value : 3.8666666)
Smallest value from the array is : 1.
```

✓ Run Succeeded | Time 234 ms | Symbol ◊ | Tabs: 4 ◊ | 27 lines, 775 characters

CONCLUSION :

From the practical 5 A, we revised the concept of for loop along with System.out.println and its functionalities .

Practical 5 B

Sort given n numbers and display them in ascending and descending order.

CODE

```
import java.util.Scanner;
class Prac5b {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the number of elements : ");
        int n = sc.nextInt();          //user input starts
        int arr[] = new int[n];
        for(int i=0;i<n;i++){
            System.out.print("element no." + (i+1) + " : ");
            arr[i] = sc.nextInt();
        }
        for(int pick=0;pick<n;pick++){          //actual calculation part
            for(int comp=pick+1;comp<n;comp++){
                if(arr[pick]>arr[comp]){          //exchange the picked element
                    with comperision element if element picked is greater than the comperision element
                    in second for loop which is right side of it.
                    int temp = arr[pick];
                    arr[pick] = arr[comp];
                    arr[comp] = temp;
                }
            }
        }
    }
}
```

```
        System.out.println("Ascending order : | Descending order : "); //printing
the result of ascending and descending order at same time to reduce one for loop
(also including in it with a line to seperate it)
        for(int i=0;i<n;i++){
            System.out.println(arr[i] + "          | " + arr[n-i-1]);
        }
    }
}
```

INPUT :

```
5
2
3
5
1
4
```

OUTPUT :

```
Enter the number of elements :
5
element no.1 : 2
element no.2 : 3
element no.3 : 5
element no.4 : 1
element no.5 : 4
Ascending order : | Descending order :
1                | 5
2                | 4
3                | 3
4                | 2
5                | 1
```

✓ Run Succeeded Time 215 ms M main ▾ Tabs: 4 ▾ Line 18, Column 18

CONCLUSION :

From the practical 5 B, we revised the logic of sorting in java using for loops .

Practical 5 C

To add two given matrices and to multiply two given matrices. courses from user. Calculate the percentage and display

CODE :

```
import java.util.Scanner;
class Prac5c {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Select one option : \n[1.] Matrix Addition    (press
1)\n[2.] Matrix Multiplication (press 2)");          //giving a choice to user to chose btw
addition and multiplication
        int choice = sc.nextInt();
        if(choice == 1){          //if block for addition
            System.out.print("Enter the number of rows for matrix 1 : ");//user
input starts
            int r1 = sc.nextInt();
            System.out.print("Enter the number of columns for matrix 1 :");
            int c1 = sc.nextInt();
            int M1[][] = new int[r1][c1];
            for(int i=0;i<r1;i++){
                for(int j=0;j<c1;j++){
                    System.out.print("M1[" + i + "][" + j + "] = ");
                    M1[i][j] = sc.nextInt();
                }
            }
            System.out.print("Enter the number of rows for matrix 2 : ");
            int r2 = sc.nextInt();
            System.out.print("Enter the number of columns for matrix 2 :");
            int c2 = sc.nextInt();
            int M2[][] = new int[r2][c2];
            for(int i=0;i<r2;i++){
                for(int j=0;j<c2;j++){
                    System.out.print("M2[" + i + "][" + j + "] = ");
                    M2[i][j] = sc.nextInt();
                }
            }
            if(r1==r2 && c1==c2){          //Actual calculation part starts
                int Madd[][] = new int[r1][c1];
                System.out.println("Addition of given two matrices : ");
                for(int i=0;i<r1;i++){          //printing matrix 1
                    for(int j=0;j<c1;j++){
```

```

        System.out.print(M1[i][j] + " ");
    }
    System.out.println();
}
System.out.println(" + ");
for(int i=0;i<r2;i++){          //printing matrix 2
    for(int j=0;j<c2;j++){
        System.out.print(M2[i][j] + " ");
    }
    System.out.println();
}
System.out.println(" = ");
for(int i=0;i<r1;i++){          //printing answer
    for(int j=0;j<c1;j++){
        Madd[i][j] = M1[i][j] + M2[i][j];
        System.out.print(Madd[i][j] + " ");
    }
    System.out.println();
}
}
else{          //rule for addition showed when wrong input entered
    System.out.println("For Addition row and column of both
matrices must be equal !");
}
}
else if(choice == 2){          //If block for Multiplication
    System.out.print("Enter the number of rows for matrix 1 : ");//user
input starts
    int r1 = sc.nextInt();
    System.out.print("Enter the number of columns for matrix 1 :");
    int c1 = sc.nextInt();
    int M1[][] = new int[r1][c1];
    for(int i=0;i<r1;i++){
        for(int j=0;j<c1;j++){
            System.out.print("M1[" + i + "][" + j + "] = ");
            M1[i][j] = sc.nextInt();
        }
    }
    System.out.print("Enter the number of rows for matrix 2 : ");
    int r2 = sc.nextInt();
    System.out.print("Enter the number of columns for matrix 2 : ");
    int c2 = sc.nextInt();
    int M2[][] = new int[r2][c2];
    for(int i=0;i<r2;i++){
        for(int j=0;j<c2;j++){
            System.out.print("M2[" + i + "][" + j + "] = ");
            M2[i][j] = sc.nextInt();
        }
    }
}
}

```

```

        if(c1 == r2){          //actual multiplication starts
            int sum =0;
            System.out.println("Multiplication of given two matrices : ");
            for(int i=0;i<r1;i++){          //printing matrix 1
                for(int j=0;j<c1;j++){
                    System.out.print(M1[i][j] + " ");
                }
                System.out.println();
            }
            System.out.println(" * ");
            for(int i=0;i<r2;i++){          //printing matrix 2
                for(int j=0;j<c2;j++){
                    System.out.print(M2[i][j] + " ");
                }
                System.out.println();
            }
            System.out.println(" = ");
            for(int i=0;i<r1;i++){          //calculating the multiplication and
printing it at the same time
                for(int j=0;j<c2;j++){
                    sum = 0;
                    for(int k=0;k<c1;k++){
                        sum += M1[i][k] * M2[k][j];
                    }
                    System.out.print(sum + " ");
                }
                System.out.println();
            }
        }
        else{          //rule for multiplication showed when wrong input entered
            System.out.println("Multiplication cannot be performed as c1 is
not equal to r2!");
        }
    }
    else{          //shows error message of invalid input if user enters choice
other than 1 and 2
        System.out.println("Invalid Input :(");
    }
}
}
}

```

INPUT :

```

2
2
3
1
2
3

```

4
5
6
3
2
7
8
9
10
11
12

OUTPUT :

```
Select one option :  
[1.] Matrix Addition      (press 1)  
[2.] Matrix Multiplication (press 2)  
2  
Enter the number of rows for matrix 1 : 2  
Enter the number of columns for matrix 1 : 3  
M1[0][0] = 1  
M1[0][1] = 2  
M1[0][2] = 3  
M1[1][0] = 4  
M1[1][1] = 5  
M1[1][2] = 6  
Enter the number of rows for matrix 2 : 3  
Enter the number of columns for matrix 2 : 2  
M2[0][0] = 7  
M2[0][1] = 8  
M2[1][0] = 9  
M2[1][1] = 10  
M2[2][0] = 11  
M2[2][1] = 12  
Multiplication of given two matrices :  
1 2 3  
4 5 6  
*  
7 8  
9 10  
11 12  
=  
58 64  
139 154
```

✓ Run Succeeded | Time 239 ms

CONCLUSION :

From the practical 5 C, we revised the concept of matrix multiplication and addition in java using nested for loops .