6.

Shambhu wants the magic board, which will display a character for the corresponding number in his science project. Now help him to develop such an application?

For Example: when the digits like 65, 66, 67, 68 are entered then the alphabet A B C and D will be displayed. Assume the no of inputs should be always 4

Sample Input 1:
Enter the digits:
65 66 67 68

Sample Output 1:
65-A 66-B 67-C 68-D

Sample Input 2:
Enter the digits:
115 116 101 112

Sample Output 2:
115-s 116-t 101-e 112-p

Capgemini

1. Problem Statement – Raj wants to know the maximum marks scored by him in each semester. The mark should be between 0 to 100, if it goes beyond the range display "You have entered invalid mark."

   Sample Input 1:
   - Enter no of semester:

   3
   - Enter no of subjects in 1 semester:

   3
   - Enter no of subjects in 2 semester:

   4
   - Enter no of subjects in 3 semester:

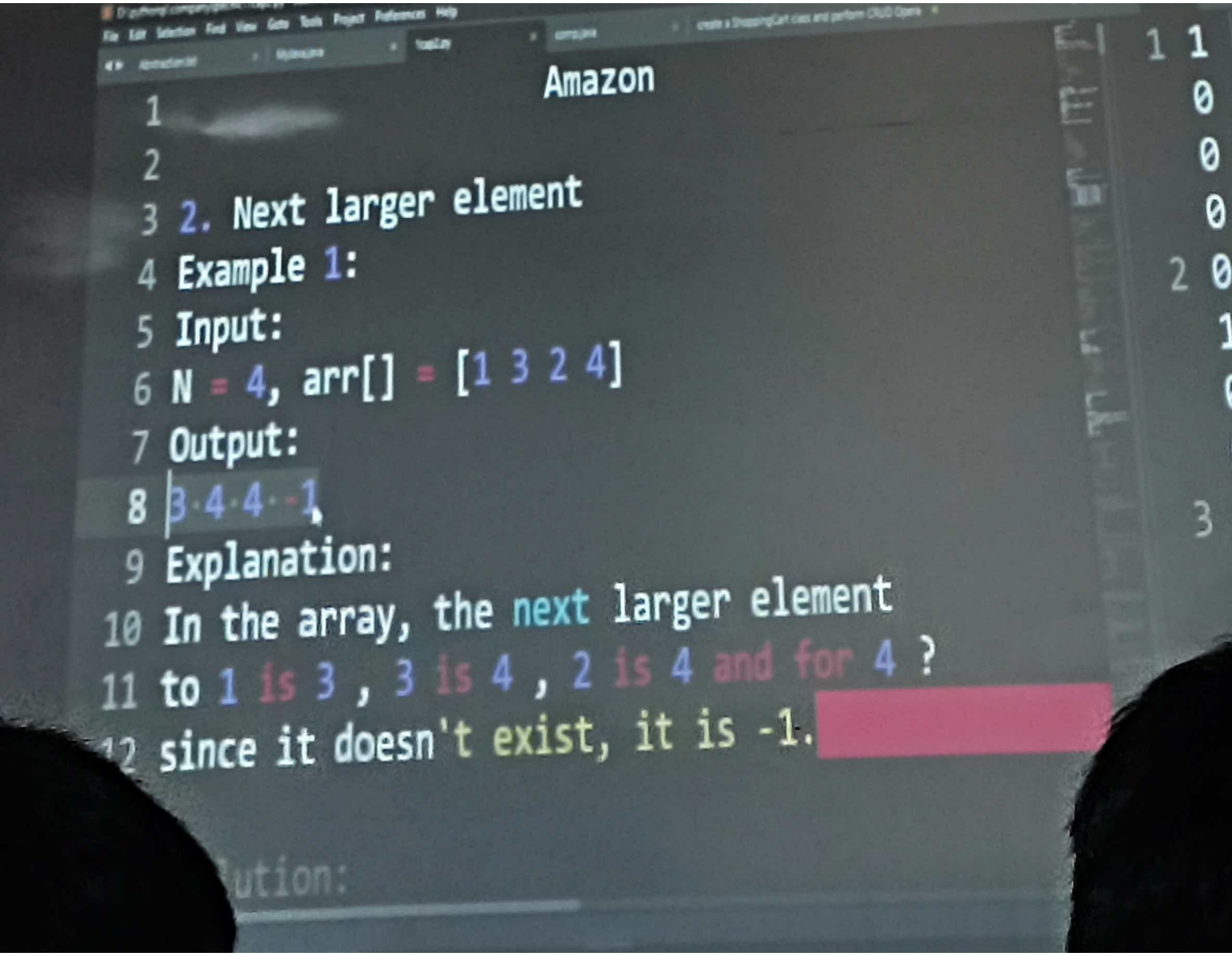
   2
   - Marks obtained in semester 1:

   50
   60
   70
   - Marks obtained in semester 2:

   90
   98
   76
   67
   - Marks obtained in semester 3:

   89
   76

   Sample Output 1:
   - Maximum mark in 1 semester:70
   - Maximum mark in 2 semester:98
   - Maximum mark in 3 semester:89


Amazon
2. Next larger element
Example 1:
Input:
N = 4, arr[] = [1 3 2 4]
Output:
3 4 4 -1
Explanation:
In the array, the next larger element
to 1 is 3 , 3 is 4 , 2 is 4 and for 4 ?
since it doesn't exist, it is -1.

Compare two Arrays

```
import java.util.*;
class Test
{
   public static void main(String args[])
   {
       int[] num1 = { 1, 2, 3 };
       int[] num2 = { 4, 5, 6 };
       int[] num3 = { 1, 2, 3 };
       System.out.println (Arrays.equals (num1, num2));
       System.out.println (Arrays.equals (num1, num3));
   }
}
```

Fill element in an array

```
import java.util.*;
class Test
{
   public static void main(String args[])
   {
        int arr[] = new int[6];
        Arrays.fill(arr, 100);
        for (int i = 0;i<arr.length; i++) {
                System.out.print(" "+arr[i]);
        }
        System.out.println();
        Arrays.fill(arr, 3, 6, 44);
        System.out.print(Arrays.toString(arr));
        }
}
```

Output

```
 100 100 100 100 100 100
[100, 100, 100, 44, 44, 44]
```

Insert an element at the specific position in the array

```
import java.util.*;
class Test
{
   public static void main(String args[])
   {
        int[] arr = {23, 42, 52, 35, 69, 55, 99, 83, 28, 19};
        int indexPosition = 2;
        int newValue = 5;
        System.out.println("Original Array : "+Arrays.toString(arr));
                for(int i=arr.length-1; i > indexPosition; i--)
                {
                        arr[i] = arr[i-1];
                }
        arr[indexPosition] = newValue;
        System.out.println("New Array: "+Arrays.toString(arr));
   }
}
```

Output

```
Original Array : [25, 14, 56, 15, 36, 56, 77, 18, 29, 49]

New Array: [25, 14, 5, 56, 15, 36, 56, 77, 18, 29]
```

Copy an array into another array

```
import java.util.*;
class Test
{
   public static void main(String args[])
   {
        int a1[]=new int[]{55, 52, 33, 48, 68, 26};
        int a2[]=Arrays.copyOf(a1,6);
                System.out.println(Arrays.toString(a2));
                int a3[]=new int[10];
                System.arraycopy(a2,0,a3,0,6);
                System.out.println(Arrays.toString(a3));
   }
}
```

Output

```
[55, 52, 33, 48, 68, 26]
[55, 52, 33, 48, 68, 26, 0, 0, 0, 0]
```

Find duplicate elements in an array

```
import java.util.*;
class Test
{
   public static void main(String args[])
   {
        String[] strArray = {"Ashish", "Prashant", "Sandip", "Ashish", "Rahul", "Manoj",
"Prashant"};
        for (int i = 0; i < strArray.length-1; i++) {
                for (int j = i+1; j < strArray.length; j++) {
                        if( (strArray[i].equals(strArray[j])) && (i != j) ) {
                                System.out.println("Duplicates String : "+strArray[j]);
                        }
                }
        }
   }
}
```

Output

Duplicates String : Ashish
Duplicates String : Prashant

Sorting with Arrays

```
import java.util.*;
class Test
{
   public static void main(String args[])
   {
      int marks[] = {72, 55, 37, 49};
      Arrays.sort(marks);
      System.out.println(Arrays.toString(marks));
   }
}
```

Output

```
[37, 49, 55, 72]
```

Print element of the array

```
import java.util.*;
class Test
{
   public static void main(String args[])
   {
       int marks [] = {55, 66, 77, 88, 99};

       System.out.println (Arrays.toString(marks));
   }
}
```

Copy an array into another array

```
import java.util.*;
class Test
{
   public static void main(String args[])
   {
        int a1[]=new int[]{55, 52, 33, 48, 68, 26};
        int a2[]=Arrays.copyOf(a1,6);
                System.out.println(Arrays.toString(a2));
                int a3[]=new int[10];
                System.arraycopy(a2,0,a3,0,6);
                System.out.println(Arrays.toString(a3));
   }
}
```

Output

```
[55, 52, 33, 48, 68, 26]
[55, 52, 33, 48, 68, 26, 0, 0, 0, 0]
```

Find duplicate elements in an array

```
import java.util.*;
class Test
{
   public static void main(String args[])
   {
        String[] strArray = {"Ashish", "Prashant", "Sandip", "Ashish", "Rahul", "Manoj",
"Prashant"};
        for (int i = 0; i < strArray.length-1; i++) {
                for (int j = i+1; j < strArray.length; j++) {
                        if( (strArray[i].equals(strArray[j])) && (i != j) ) {
                                System.out.println("Duplicates String : "+strArray[j]);
                        }
                }
        }
   }
}
```

Output

Duplicates String : Ashish

Duplicates String : Prashant

Anonymous Array

```
import java.util.*;
class Test
{
  public static void main(String args[])
  {
 System.out.println(new int[]{55,52,33,48,68, 26}.length);
 System.out.println(new int[]{91,44,17,41,11}[3]);
 }
```

Compare two Arrays

```
import java.util.*;
class Test
{
   public static void main(String args[])
   {
      int[] num1 = { 1, 2, 3 };
      int[] num2 = { 4, 5, 6 };
      int[] num3 = { 1, 2, 3 };
      System.out.println (Arrays.equals (num1, num2));
      System.out.println (Arrays.equals (num1, num3));
   }
}
```

Output

```
false
true
```

Sorting with Arrays

```
import java.util.*;
class Test
{
    public static void main(String args[])
    {
        int marks[] = {72, 55, 37, 49};
        Arrays.sort(marks);
        System.out.println(Arrays.toString(marks));
    }
}
```

Output

```
[37, 49, 55, 72]
```

Print element of the array

```
import java.util.*;
class Test
{
   public static void main(String args[])
   {
      int marks [] = {55, 66, 77, 88, 99};

      System.out.println (Arrays.toString(marks));
   }
}
```

Output

```
[55, 66, 77, 88, 99]
```

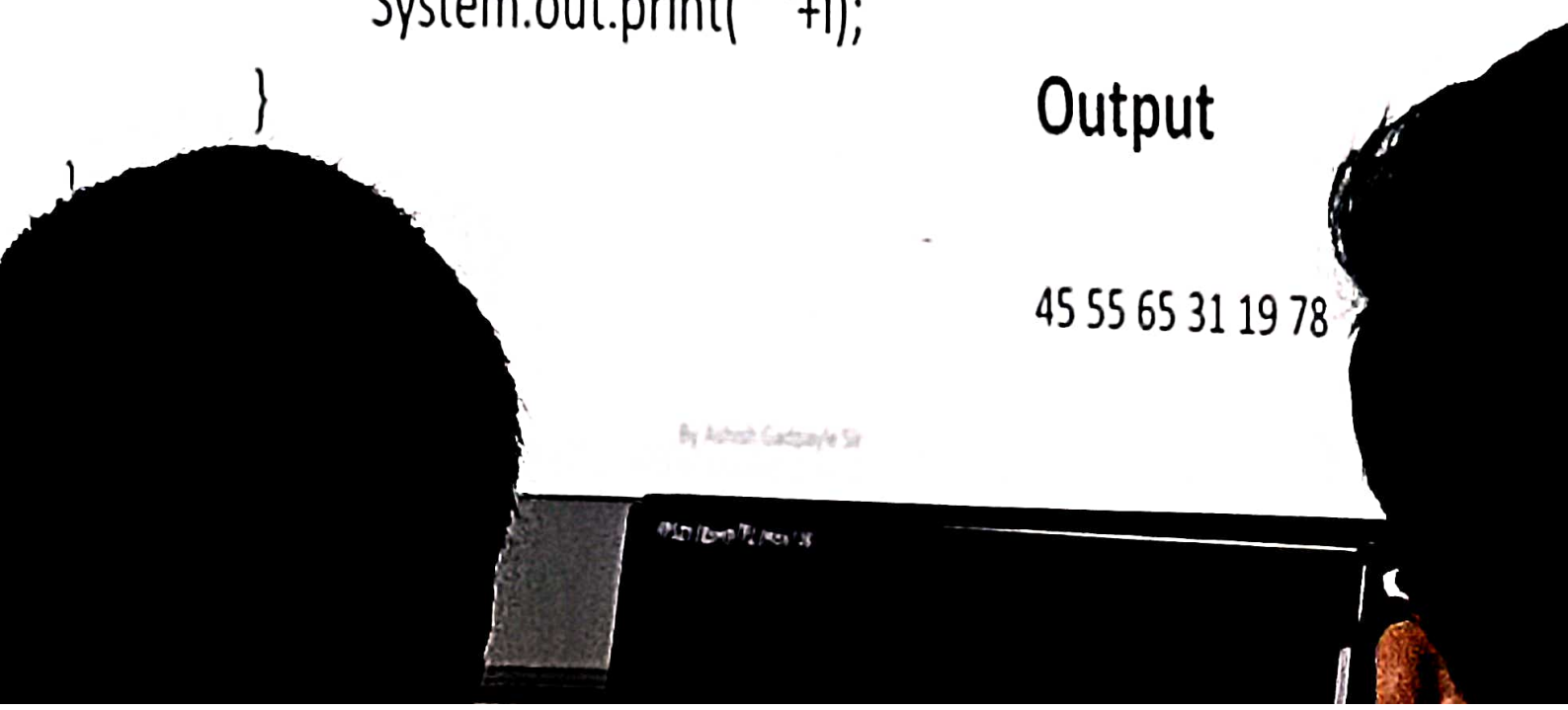
Print array using for each loop

```
class Test
{
   public static void main(String args[])
   {
       int[] arr = new int[]{45, 55, 65, 31, 19, 78};

              for (int i: arr){
                     System.out.print(" "+i);
              }
}
```

Output

```
45 55 65 31 19 78
```

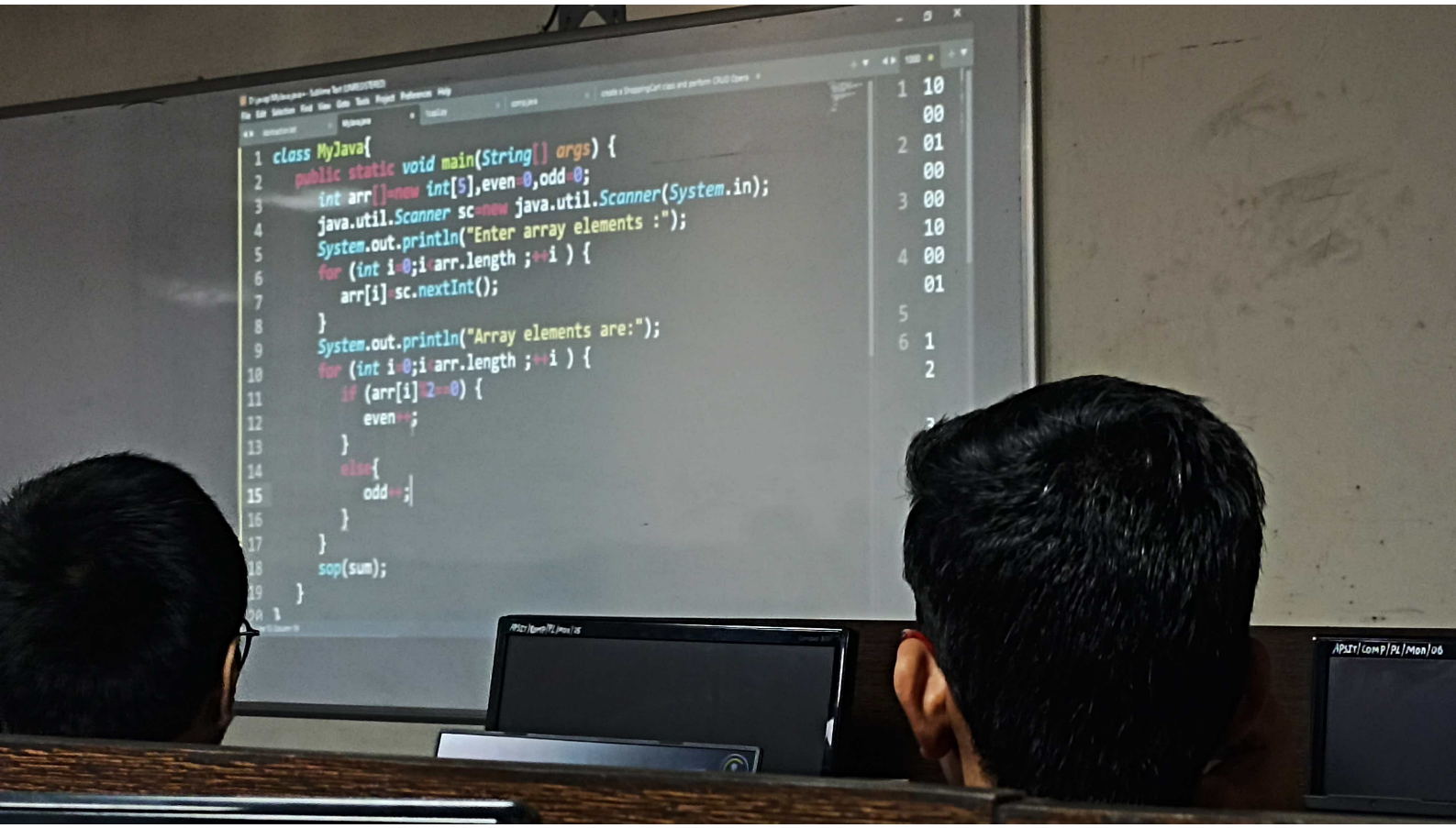


Find smallest and largest number in array

```
class ArrayDemo
{
  public static void main (String[] args)    {
      int arr[]={2,3,1,11,4,10,33,7,2,5};
            int small,large,i;
            small=large=arr[0];
      for(i=1;i<10;i++){
            if(arr[i]<small)
              small=arr[i];
            if(arr[i]>large)
              large=arr[i];
      }
System.out.print("smallest no is "+small+" and largest no is = "+large);
```

Output:

smallest no is 1 and
largest no is = 33


class MyJava{
public static void main(String[] args) {
int arr[]=new int[5],even=0,odd=0;
java.util.Scanner sc=new java.util.Scanner(System.in);
System.out.println("Enter array elements :");
for (int i=0;i<arr.length ;++i) {
arr[i]=sc.nextInt();
}
System.out.println("Array elements are:");
for (int i=0;i<arr.length ;++i) {
if (arr[i]%2==0) {
even++;
}
else{
odd++;
}
}
sop(sum);
}
APSIT/COMP/PL/Mon/06

Input values into array and display that values

```
class ArrayDemoc {
  public static void main (String[] args) {
    // declares an Array of integers.
    int[] arr;
    // allocating memory for 5 integers.
    arr = new int[5];
    // initialize the elements of the array
    arr[0] = 10;
    arr[1] = 20;
    arr[2] = 30;
    arr[3] = 40;
    arr[4] = 50;
    // accessing the elements
    for (int i = 0; i < arr.length; i++)
      System.out.println("Element at index " + i +  " : "+ arr[i]);
  }
```

Output:

```
Element at index 0 : 10
Element at index 1 : 20
Element at index 2 : 30
Element at index 3 : 40
Element at index 4 : 50
```

One dimensional array

An array represent with single subscript variable is known as one dimensional array.

Syntax:

```
datatype arrayname[ ]=new datatype[size];
```

EX:

```
int arr[ ]=new int[5];
    //here arr can hold 5 integer values
```

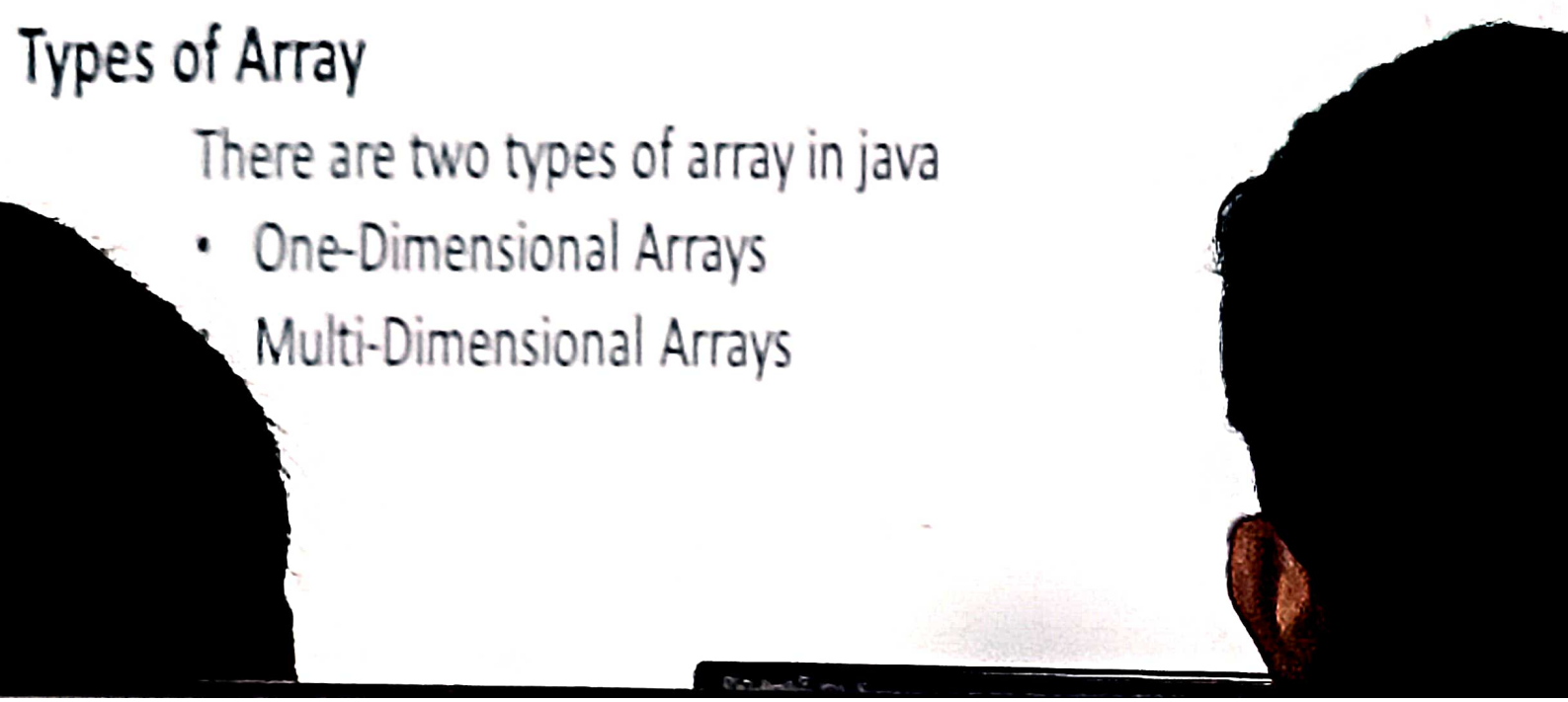
Limitation of Array

Array size is fixed, which means we cannot increase or decrease its size after its creation.

Types of Array

There are two types of array in java

- One-Dimensional Arrays
- Multi-Dimensional Arrays



Following are the important points in java:

- All arrays are dynamically allocated in java.
- The size of an array must be specified by an int.
- The variables in the array are ordered and each element's index is started from 0.
- Every array type implements the interface Cloneable and Serializable.
- Arrays are objects in java, we can find their length using member property **length**.
- Array variable can also be declared like other variables with [] after the data type
- The direct superclass of an array type is Object.

Need of Array

In java, array is used to collect similar type of values or objects to send all those multiple values with single call from one method to another methods either as an argument or as a return value.

What is Array?

- Array is a referenced type of data type in java.
- It is used to create fixed number of multiple variables of same type to store multiple values of similar type in contiguous memory locations with single variable name.


1 1 10
2 2 9
3 4 7
4 5 6
5
1 1000
2 0100
3 0010
4 0001
5
6 1 2 3
4 ->sumo
frow

continue statement

continue statement is opposite to that of break statement, instead of terminating the loop, it forces to execute the next iteration of the loop

```
class Ex
{
   public static void main(String[] arg)    {
      for(int i=1;i<=10;i++)   {
         if(i==5)
            continue;
       System.out.println(i);
       }
   }
}
```

Output:

```
1
2
3
4
6
7
8
9
10
```

break statement

It is used to terminate currently executing looping statement . It is allowed within looping and switch statement .

```
class Ex
{
   public static void main(String[] arg)     {
      for(int i=1;i<=10;i++)   {
          if(i==5)
              break;
       System.out.println(i);
       }
   }
}
```

Output:

```
1
2
3
4
```

Practice: Print following patterns

- 1111
  2222
  3333
  4444

- 1 2 3 4
  5 6 7 8
  9 10 11 12
  13 14 15 16

- ABCD
  EFGH
  IJKL
  MNOP

- 1
  12
  123
  1234

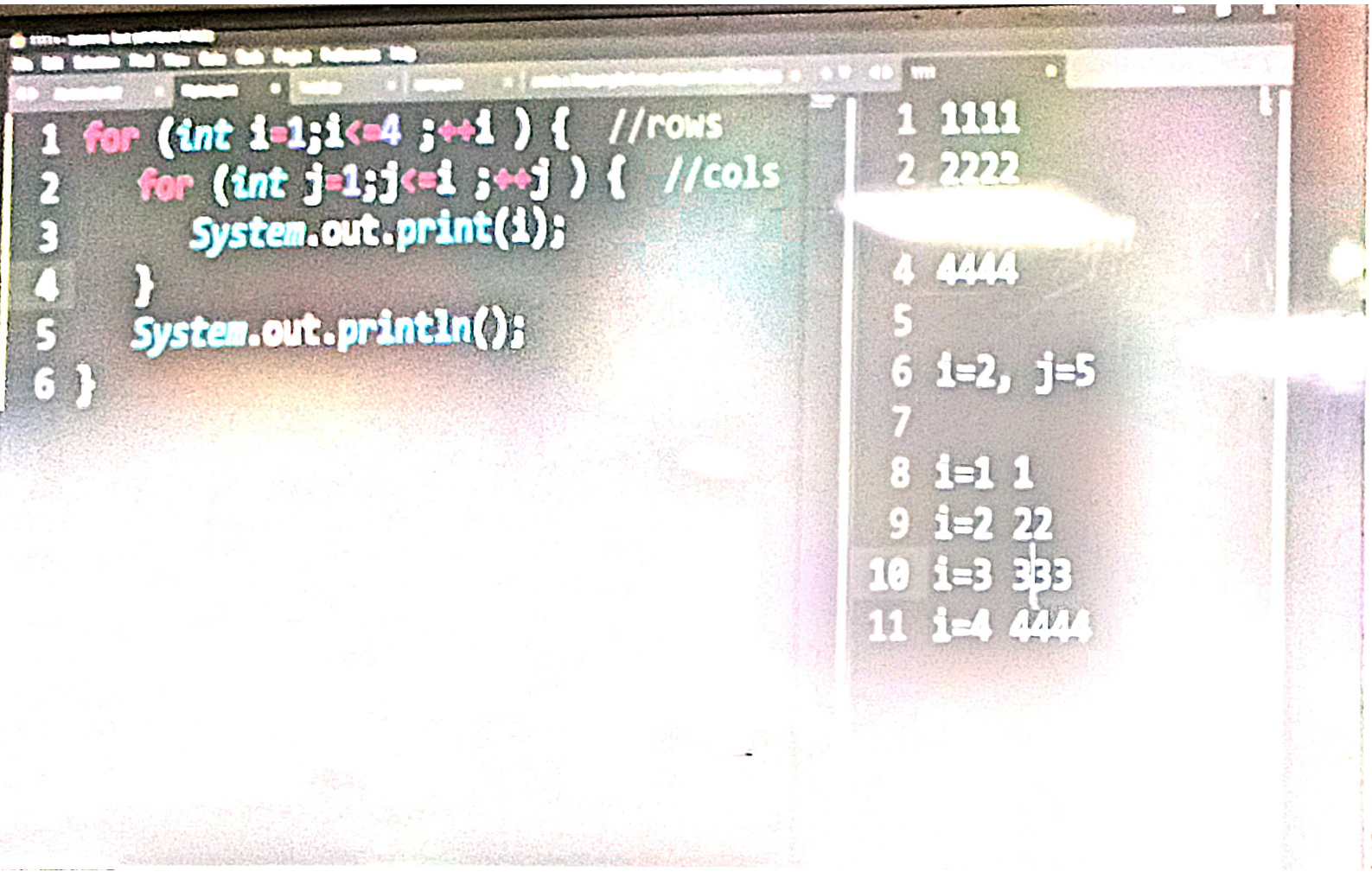
- *
  **
  ***
  ****

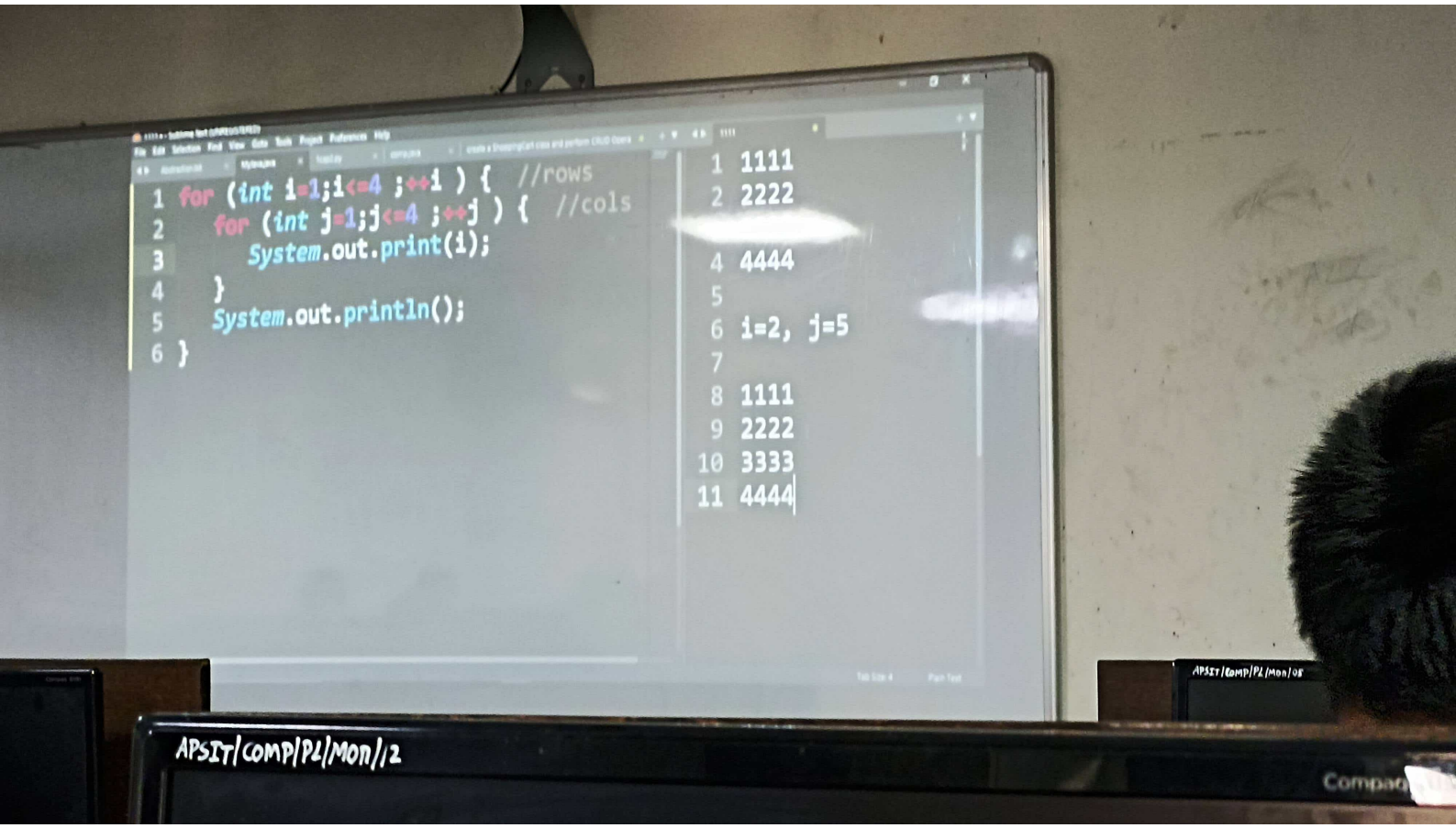
- ****
  ***
  **
  *

- ****
  ***
  **
  *

- *
  **
  ***
  ****

- *
  ***
  *****
  ****


1 for (int i=1;i<=4 ;++i) { //rows
2 for (int j=1;j<=i ;++j) { //cols
3 System.out.print(i);
4 }
5 System.out.println();
6 }
1 1111
2 2222
4 4444
5
6 i=2, j=5
7
8 i=1 1
9 i=2 22
10 i=3 333
11 i=4 4444

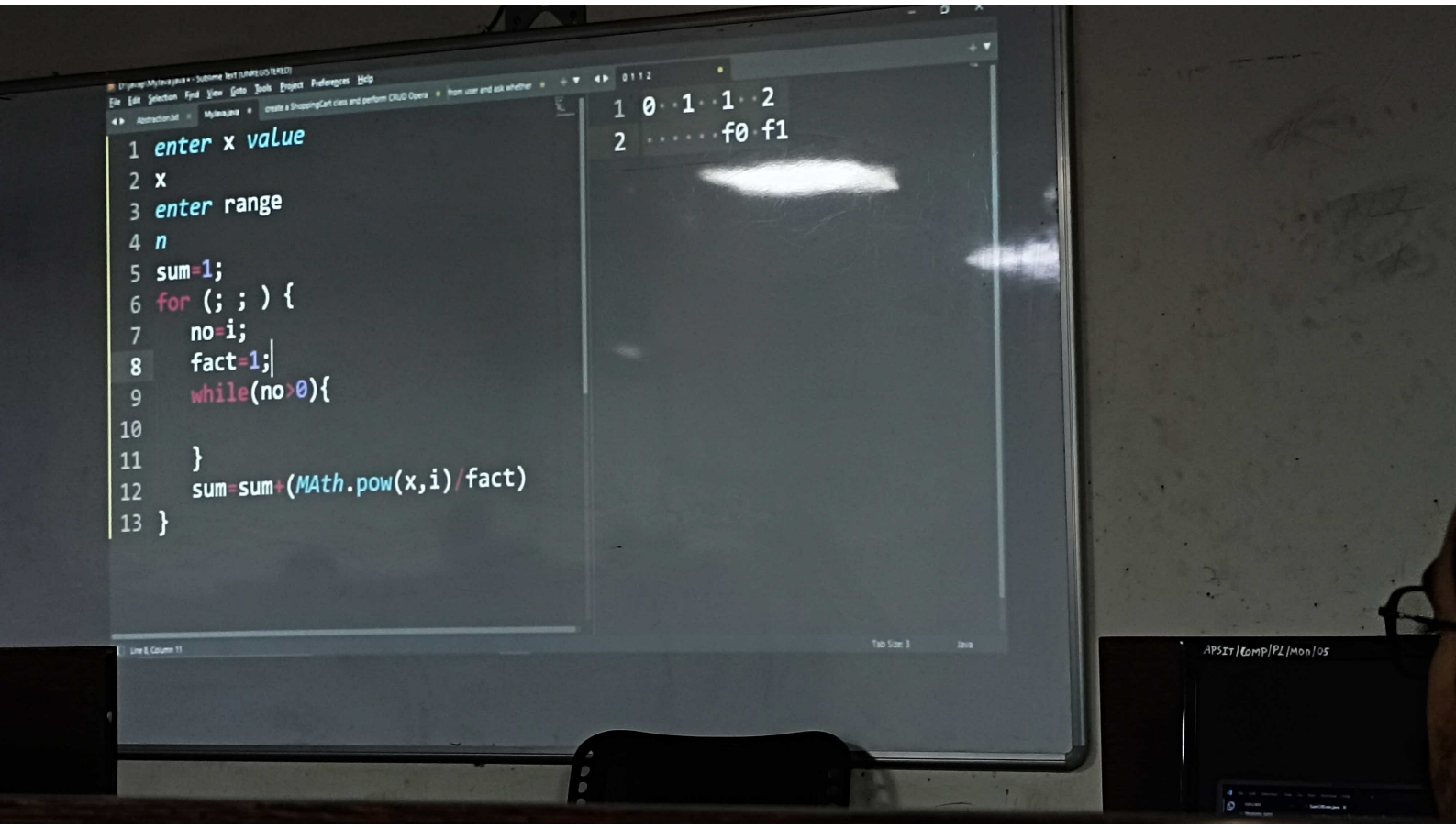

for (int i=1;i<=4 ;++i) { //rows
for (int j=1;j<=4 ;++j) { //cols
System.out.print(i);
}
System.out.println();
}
1111
2222
4444
i=2, j=5
1111
2222
3333
4444
APSIT/COMP/PL/MON/12

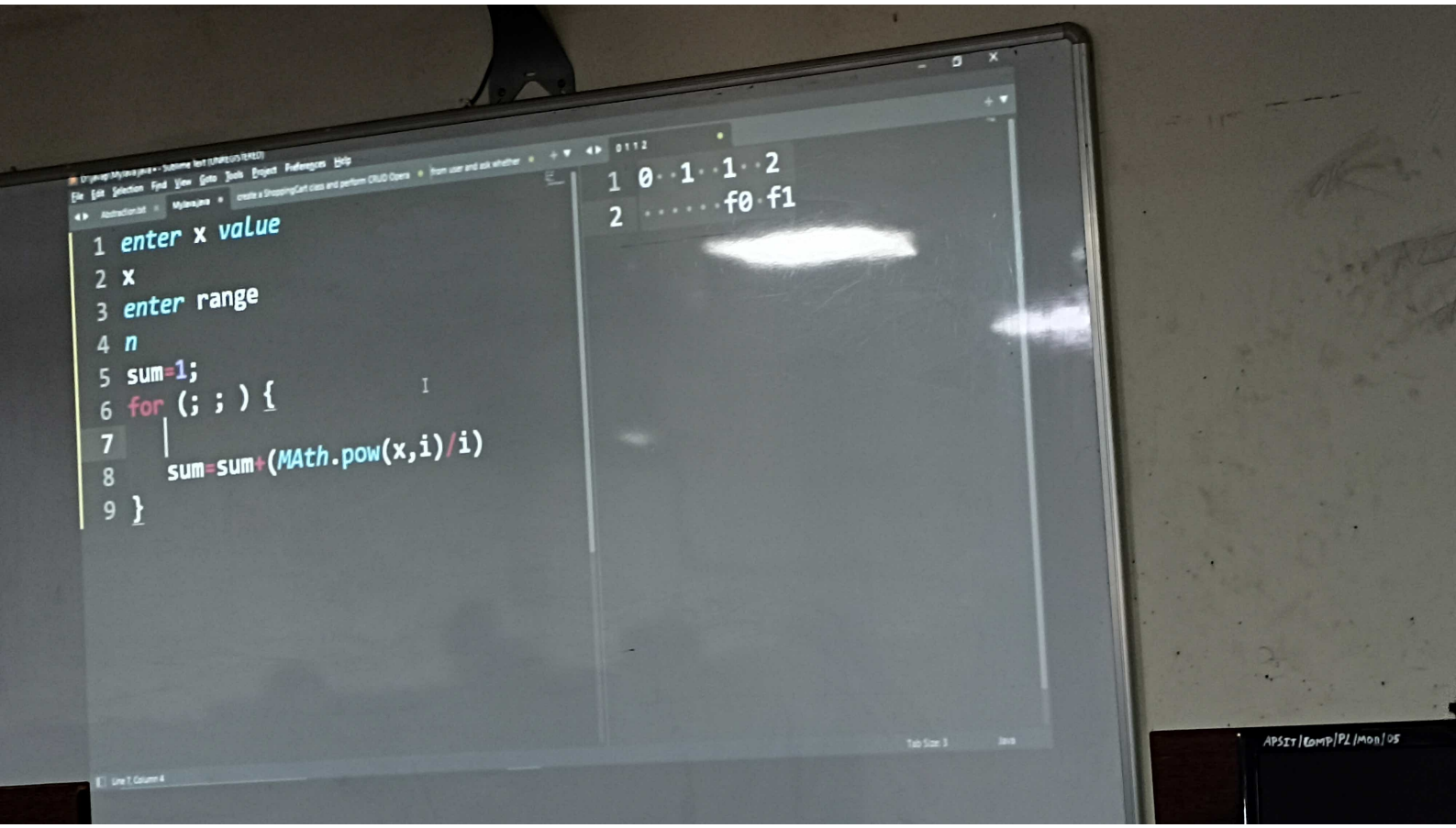
Find max number in array using for loop

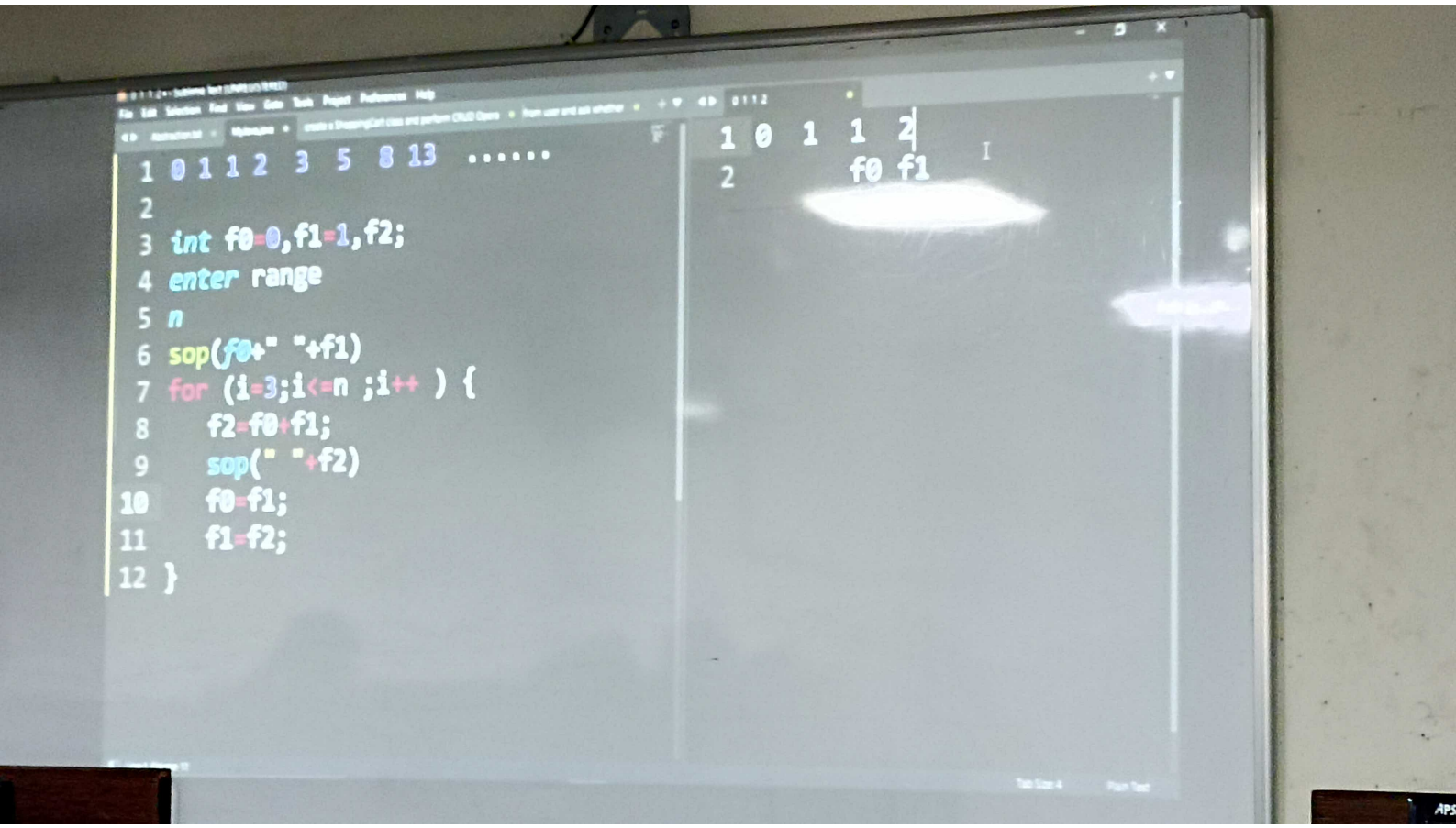
```
class ForEach   {
   public static void main(String[] arg)      {
      int[] marks = { 125, 132, 95, 116, 110 };
      int max = marks[0];
      for (int num : marks)
      {
         if (num > max)  {
            max = num;
         }
      }
      System.out.println("Student highest score is-"+max);
   }
}
```

Output:

Student highest score is 132


enter x value
x
enter range
n
sum=1;
for (; ;) {
no=i;
fact=1;
while(no>0){
}
sum=sum+(MAth.pow(x,i)/fact)
}
0 1 1 2
f0 f1
APSIT/COMP/PL/Mon/05


enter x value
x
enter range
n
sum=1;
for (; ;) {
sum=sum+(MAth.pow(x,i)/i)
}
0 1 1 2
f0 f1
APSIT/COMP/PL/MON/05


1 0 1 1 2 3 5 8 13 .....
int f0=0,f1=1,f2;
enter range
n
sop(f0+" "+f1)
for (i=3;i<=n ;i++) {
f2=f0+f1;
sop(" "+f2)
f0=f1;
f1=f2;
}
1 0 1 1 2
2
f0 f1

Practice:

- Find factorial of any number.
- Print sum of even numbers from 1 to 20.
- Print sum of numbers any number.
- Print fibonacci series.
- Find sum of following series.

  sum= **1 + x/1 + x^2/2 + x^3/3 + .. + x^n/n**
- Find sum of following series.

  sum= **1 + x/1! + x^2/2! + x^3/3! + .. + x^n/n!**
- Print numbers in reverse order with difference two.
- Multiply 2 positive numbers without using * operator

Practice:

- Accept a number and check number is armstrong or not.
- Accept a number and check number is palindrome or not.

Practice:

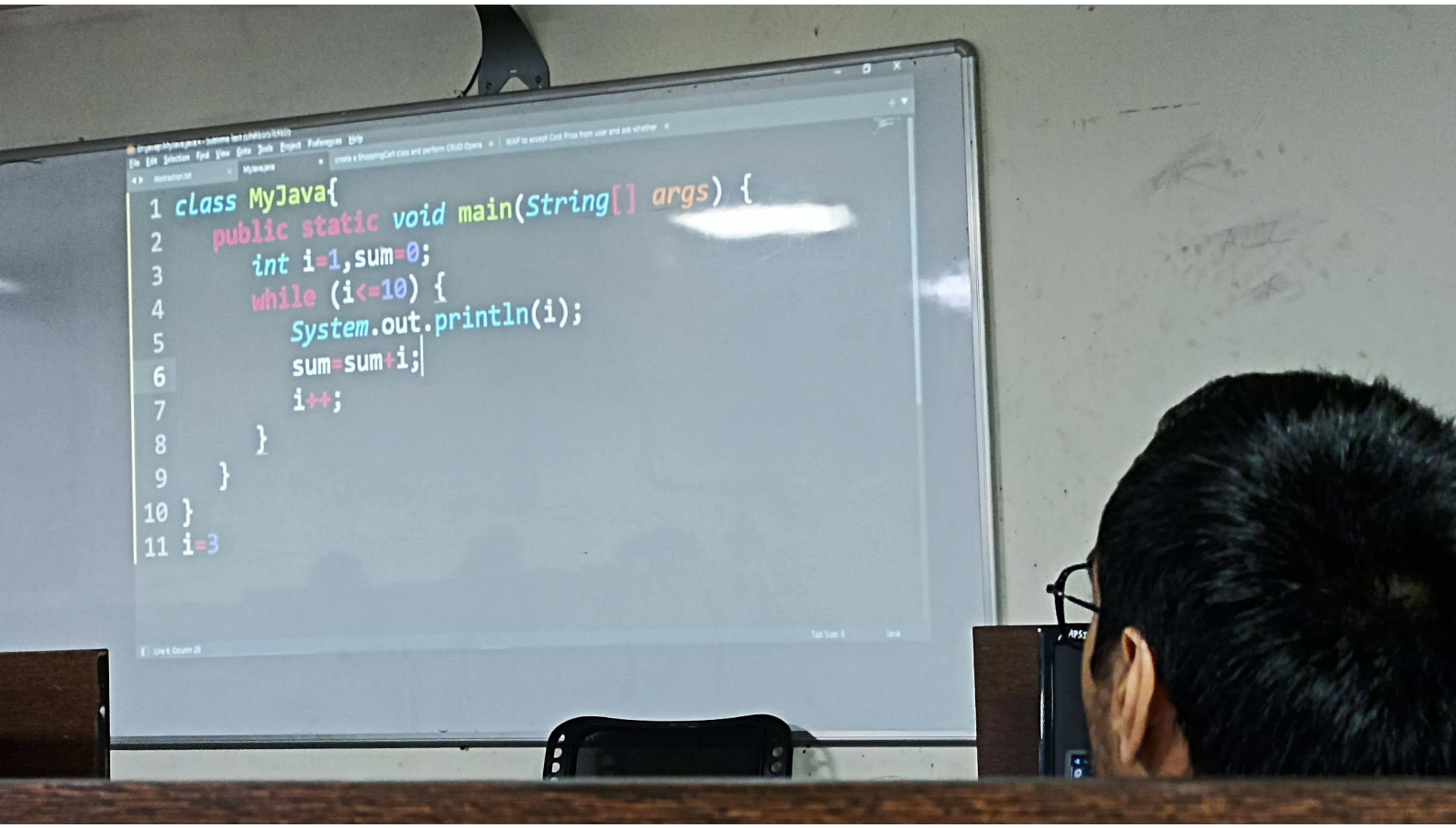
- Accept a number and count number of digits entered by user.
- Accept a number and print sum of digits of any number.
- Accept a number and find multiplication of any number.
- Accept a number and find factorial of any number.

Print reverse of a number

```
import java.util.Scanner;
class Pro {
   public static void main(String[] args) {
            int rev=0,no,rem;
            System.out.println("Enter Any Number : ");
            Scanner sc=new Scanner(System.in);
            no=sc.nextInt();
                  while(no>0)
                  {
                   rem=no%10;
                   rev=rev*10+rem;
                   no=no/10;
                  }
          System.out.println("Reverse number is "+rev);
   }
}
```

Output:

```
Enter Any Number : 1234
Reverse No is : 4321
```


class MyJava{
public static void main(String[] args) {
int i=1,sum=0;
while (i<=10) {
System.out.println(i);
sum=sum+i;
i++;
}
}
}
i=3

It is used to execute group of statement repeatedly till the condition is satisfy.

Java support four types of looping statement.

1. while loop
2. do while loop
3. for loop
4. for each loop

Basically in looping statement three things are included and that are most important .

1. initialization
2. condition
3. updation