

## SECTION A (40 Marks)

*Attempt all questions*

### Question 1

- (a) Define abstraction. [2]
- (b) Differentiate between searching and sorting. [2]
- (c) Write a difference between the functions `isUpperCase( )` and `toUpperCase( )`. [2]
- (d) How are private members of a class different from public members? [2]
- (e) Classify the following as primitive or non-primitive datatypes: [2]
  - (i) `char`
  - (ii) `arrays`
  - (iii) `int`
  - (iv) `classes`

## Comments of Examiners

- (a) Many candidates, instead of writing the definition of abstraction, wrote the definition of encapsulation. Some candidates did not write important keywords like hiding the background details or showing the essential details.
- (b) Some candidates wrote the types of searching and sorting instead of their differences based on Java programming. A few candidates used the same words as given in the question i.e. searching means to search something and sorting means sorting data.
- (c) Majority of the candidates were not clear about the difference between Character class functions and String functions. Most of the candidates were not aware that `isUpperCase()` happens to be a Character class function.
- (d) Majority of the candidates wrote very general answers like- can be seen by everyone or shown to or used by everyone or public premises or cannot be seen by everyone or only user can see or use, or everything is not shown or private premises.
- (e) Most of the candidates were able to answer this question. However, a few candidates got confused between primitive and non-primitive data types.

## Suggestions for teachers

- Explain in detail with relevant examples, the basic principles of OOPS and difference between them.
- Emphasize the importance of the keywords in the definitions and stress on their usage while writing the definitions.
- Explain to the students, the meaning of and the difference between the terms searching and sorting as a concept to be used in programming with illustrations.
- Teach the different Character functions and String functions with examples
- Encourage the students to use different functions in programming.
- Program segments with library functions to write the output or return data type or purpose or questions will improve their learning.
- Ensure that students understand the meaning of all technical terms and their use.
- Train students to write the answers as per Java, rather than writing general meaning of the terms.
- Explain primitive and non-primitive data types with examples. Also give sufficient practice to the students on questions based on data types

## MARKING SCHEME

### Question 1

(a)	abstraction: the act of representing essential features and hide unnecessary or background details is called abstraction.							
(b)	<table border="1"> <tr> <th>Searching</th> <th>Sorting</th> </tr> <tr> <td>To find an element in an array is called searching.</td> <td>To arrange the elements in an order is called sorting.</td> </tr> </table>	Searching	Sorting	To find an element in an array is called searching.	To arrange the elements in an order is called sorting.			
Searching	Sorting							
To find an element in an array is called searching.	To arrange the elements in an order is called sorting.							
(c)	<table border="1"> <tr> <th>isUpperCase</th> <th>toUpperCase</th> </tr> <tr> <td>CHECKS whether a given character is an uppercase letter or not</td> <td>CONVERTS the character to its upper case</td> </tr> <tr> <td>Output of this function is boolean</td> <td>Output of this function is character.</td> </tr> </table>	isUpperCase	toUpperCase	CHECKS whether a given character is an uppercase letter or not	CONVERTS the character to its upper case	Output of this function is boolean	Output of this function is character.	
isUpperCase	toUpperCase							
CHECKS whether a given character is an uppercase letter or not	CONVERTS the character to its upper case							
Output of this function is boolean	Output of this function is character.							
(d)	public members can be accessed from any other class. private members can only be accessed within the declared class.							
(e)	char – primitive      array – non-primitive    int – primitive    class – non primitive							

### Question 2

- (a) (i) `int res = 'A';` [2]  
What is the value of **res**?
- (ii) Name the package that contains wrapper classes.
- (b) State the difference between **while** and **do while** loop. [2]
- (c) `System.out.print("BEST ");` [2]  
`System.out.println("OF LUCK");`  
Choose the correct option for the output of the above statements
- (i) BEST OF LUCK
- (ii) BEST  
OF LUCK
- (d) Write the prototype of a function **check** which takes an integer as an argument and returns a character. [2]

(e) Write the return data type of the following function.

[2]

(i) endsWith()

(ii) log()

## Comments of Examiners

- (a) (i) Majority of the candidates wrote the answer as ‘A’ or “A” or A. Candidates were unable to understand the implicit conversion.
- (ii) Most of the candidates were able to answer this question. However, some candidates wrote io or util as the answer while a few candidates wrote the answer Lang or in capital letters LANG.
- (b) Most of the candidates were able to answer this question. However, some candidates interchanged the answers.
- (c) Majority of the candidates were unable to select the correct option for the output of the given statements as either they seemed to be unclear of the working of the println() method or did not read all the options carefully.
- (d) Several candidates could not write the prototype of the function given in the question.
- (e) (i) Many candidates, instead of writing Boolean, wrote return type as String.
- (ii) Many candidates wrote return type as float or integer. Some candidates mentioned data types with the first letter in uppercase.

## Suggestions for teachers

- Explain type of conversions, explicit and implicit, with examples especially character type to integer type and ASCII code and *vice-versa*.
- Ensure that all concepts of the lesson on wrapper classes are clear to the students.
- Explain the concept of while and teach this concept with output printing questions.
- Clarify the working of println() method explicitly.
- Give practice to the students in the computer lab, of print() and println() methods, with ample examples.
- Explain to the students the purpose of prototype and every segment of the prototype clearly
- Ensure that the students have understood the data types of the parameters
- While teaching in-built functions, ensure that return data type of each function, whether mathematical or character or string, has been explained, with examples, to the students
- Explain the difference between the data types and the wrapper classes with examples.

## MARKING SCHEME

### Question 2

(a)	(i) 65 (ii) lang						
(b)	<table border="1"><thead><tr><th>while loop</th><th>do while loop</th></tr></thead><tbody><tr><td>Entry controlled loop</td><td>Exit controlled loop</td></tr><tr><td>If condition is false in the beginning loop never executes, <i>minimum repetition is 0</i></td><td>Loop will execute at least once even if the condition is false. <i>minimum repetition is 1</i></td></tr></tbody></table>	while loop	do while loop	Entry controlled loop	Exit controlled loop	If condition is false in the beginning loop never executes, <i>minimum repetition is 0</i>	Loop will execute at least once even if the condition is false. <i>minimum repetition is 1</i>
while loop	do while loop						
Entry controlled loop	Exit controlled loop						
If condition is false in the beginning loop never executes, <i>minimum repetition is 0</i>	Loop will execute at least once even if the condition is false. <i>minimum repetition is 1</i>						
(c)	(i) BEST OF LUCK						
(d)	char check (int n)						
(e)	(i) boolean (ii) double						

### Question 3

(a) Write a Java expression for the following: [2]

$$\frac{\sqrt{3x + x^2}}{a + b}$$

(b) What is the value of y after evaluating the expression given below? [2]

y+= ++y + y-- + -- y; when int y=8

(c) Give the output of the following: [2]

(i) Math.floor (-4.7)

(ii) Math.ceil(3.4) + Math.pow(2, 3)

(d) Write two characteristics of a constructor. [2]

(e) Write the output for the following: [2]

System.out.println("Incredible"+"\\n"+"world");

(f) Convert the following **if else if** construct into **switch case** [2]

if( var==1)  
System.out.println("good");

```
else if(var==2)
System.out.println("better");
else if(var==3)
    System.out.println("best");
else
System.out.println("invalid");
```

(g) Give the output of the following string functions: [2]

(i) "ACHIEVEMENT".replace('E', 'A')

(ii) "DEDICATE".compareTo("DEVOTE")

(h) Consider the following String array and give the output [2]

```
String arr[]= {"DELHI", "CHENNAI", "MUMBAI", "LUCKNOW", "JAIPUR"};
System.out.println(arr[0].length()> arr[3].length());
System.out.print(arr[4].substring(0,3));
```

(i) Rewrite the following using ternary operator: [2]

```
if (bill >10000)
discount = bill * 10.0/100;
else
discount = bill * 5.0/100;
```

(j) Give the output of the following program segment and also mention how many times the loop is executed: [2]

```
int i;
for ( i = 5 ; i > 10; i ++ )
System.out.println( i );
System.out.println( i * 4 );
```

## Comments of Examiners

- (a) Majority of the candidates did not put the parenthesis in the denominator. Some candidates were not familiar with the syntax of sqrt function.
- (b) Many candidates seemed to be unaware of the meaning of  $y+=$ . Most candidates made errors while evaluating prefix and post fix operators.  
In many answer scripts, addition of the initial value of the variable to the evaluated value of the expression was incorrect. Some candidates added the decremented value to the evaluated value of the expression.
- (c) (i) Several candidates did not write the negative sign in the answer. Many students did not write the answer in decimals. Some candidates, instead of writing the answer as  $-5.0$ , wrote it as  $-4$  or  $-4.0$  or  $4.0$ .  
(ii) Most of the candidates wrote 12 as the answer instead of  $12.0$ .
- (d) Majority of the candidates were able to answer this question. However, some candidates wrote that constructor is used to construct objects, or it is used to make the program easier.
- (e) Several candidates simply copied the question. Majority of the candidates were not clear of the use of  $\backslash n$ . The common errors made by the candidates were: Incredible  $\backslash n$  world, "Incredible" +  $\backslash n$  + "world".
- (f) Many candidates wrote if with in case.  
The common errors made by majority of the candidates were:  
(i) Case 1:  $\text{if}(\text{var}==1)$  or  $\text{var}==1$ .  
(ii) Switch  $\text{var}()$ ; or switch was missing.  
(iii) Break and default statements were missing.

## Suggestions for teachers

- While explaining the hierarchy of operators to the students, lay stress on the importance of parenthesis, with the help of appropriate examples.
- Emphasize on the importance of short hand notation.
- Guide students how to evaluate prefix/postfix notation.
- Explain the addition of the initial value of the variable to the evaluated value of the expression in such expressions.
- Explain the difference between  $y+=$   $\langle$ expression $\rangle$  and  $y=$   $\langle$ expression $\rangle$ .
- Give adequate practice to the students of expressions with Math functions.
- Encourage students to execute and see the outputs practically.
- Teach, with examples, the definition of constructor and its characteristics.
- Clarify Escape sequence and its working with sufficient number of examples.
- Demonstrate on the computer, the effect of the escape sequence on the output.
- Interpret to the students the difference between switch and  $\text{if}()$  statement.

- (g) (i) Majority of the candidates were able to answer this question. However, in a few answer scripts only the first occurrence of E was converted.
- (ii) Majority of the candidates were unclear about the working of compareTo() function. Several candidates were unable to calculate ASCII difference. In a few answer scripts, the answer calculated was either 18 or -1. Some candidates wrote any random negative value as the answer.
- (h) Some candidates, instead of writing false as the answer, wrote 5>6. Some candidates were unclear whether the position of second argument was included in the output or not. Some candidates extracted the character at third index also.
- (i) The common errors made by majority of the candidates were:
- (i) `if(bill>10000)? bill*10.0/100: bill*5.0/100.`
- (ii) The assignment part of the statement was not being written.
- (iii) `'discount ='` - not being written.
- (j) Majority of the candidates were unable to identify the final limit of the loop. Some candidates wrote error in the condition of the loop.

- Give adequate practice in conversion of if to switch and vice-versa.
- Discuss with the students, examples of various problems with the appropriate conditional statements.
- Explain String functions with examples.
- Theoretically and through practical sessions, explain the execution of the compareTo() function, so that the students understand the difference between equals() and compareTo() functions.
- Explain the importance of ASCII values of the letters (both uppercase and lowercase) to evaluate the output value.
- Clarify to the students how relational operators works in output statement and results as true or false.
- Practically show to the students how to extract the number of characters in substring function.
- Interpret clearly the purpose and the working of the second argument in substring() function.
- Explain to the students that '?' represents if and: represents else.
- Make the students practice conversion from if- else to ternary and ternary to if- else in the class.
- Explain all the variations of for loop.
- Teach loops with different initial and final values and their output after execution.

## MARKING SCHEME

### Question 3

(a)	$Math.sqrt(3 * x + x * x)/(a + b)$ OR <code>Math.sqrt(3*x+Math.pow(x,2))/(a+b);</code>
(b)	33
(c)	i. -5.0 ii. 12.0

(d)	(i) Constructors have the same name as that of the class they belong to. (ii) Constructors are executed when an object is created.
(e)	Incredible  world
(f)	switch(var) { case 1: System.out.println("good"); break; case 2: System.out.println("better"); break; case 3: System.out.println("best"); break; default: System.out.println("invalid"); }
(g)	(i) ACHIAVAMANT (ii) -18
(h)	(i) false (ii) JAI
(i)	discount = bill>1000? bill*10.0/100: bill *5.0/100;
(j)	Output: 20                      Loop is not executed even once

## SECTION B (60 Marks)

*Attempt any four questions from this Section*

### Question 4

Design a class **RailwayTicket** with following description:

[15]

- Instance variables/data members :
- String name : To store the name of the customer
- String coach : To store the type of coach customer wants to travel
- long mobno : To store customer's mobile number

int amt : To store basic amount of ticket

int totalamt : To store the amount to be paid after updating the original amount

Member methods :

void accept () – To take input for name, coach, mobile number and amount.

void update () – To update the amount as per the coach selected

(extra amount to be added in the amount as follows)

Type of Coaches	Amount
First_AC	700
Second_AC	500
Third_AC	250
sleeper	None

void display () – To display all details of a customer such as name, coach, total amount and mobile number.

Write a main method to create an object of the class and call the above member methods.

## Comments of Examiners

Numerous types of errors were committed in this question by many candidates, namely:

- Used a different class name other than the one given in the question.
- Variable declaration was done at many places i.e. immediately after class and also inside accept()
- Function names and variable names were different from the ones asked in the question.
- The 'totalamt' which was to be calculated was accepted from the user.
- Update() function.
- Syntax error while writing input statement e.g. nextInt() - which letter to be written in capital.
- While comparing types of coaches, words were not written within double quotes.
- Calculation of extra amount.
- Syntax errors in the object creation and function call statement.

## Suggestions for teachers

- Train students to write programs involving multiple functions.
- Explain the difference between global and local variables and how to use them in different functions in a program.
- How to create an object and invoke the function using the object needs to be explained thoroughly and practised properly.

## MARKING SCHEME

### Question 4

```

import java.util.*;
class RailwayTicket
{
    String name, coach;
    long mobno;
    int amt ,totalamt;
    void accept()
    {
        Scanner sc = new Scanner(System.in);
        System.out.println("enter name , coach and mobile no and amount");
        name=sc.next();
        coach=sc.next();
        mobno=sc.nextLong();
        amt=sc.nextInt(); }
    void update()
    {
        if(coach. equalsIgnoreCase ("First_AC"))
            totalamt=amt+700;
        else if(coach. equalsIgnoreCase ("Second_AC"))

            totalamt=amt+500;
        else if(coach. equalsIgnoreCase ("Third_AC"))
            totalamt=amt+250;
        else
            totalamt=amt;
    }
    void display()          {
        System.out.println("name is :"+name);
        System.out.println("mobile no is :"+mobno);
        System.out.println("coach is :"+ coach);
        System.out.println("total amount is :"+totalamt); }
    public static void main (String arg[])
    {
        RailwayTicket ob =new RailwayTicket();
        ob.accept();
        ob.update();
        ob.display(); }}
    
```

STEPS
class name
Variable declaration
void accept ()
Four inputs
void update()
Four conditions and calculations
void display ()
Output (totalamt compulsory)
Object creation <i>with one function call</i>
Variable description/Mnemonic codes

## Question 5

[15]

Write a program to input a number and check and print whether it is a **Pronic** number or not. (Pronic number is the number which is the product of two consecutive integers)

Examples:  $12 = 3 \times 4$

$20 = 4 \times 5$

$42 = 6 \times 7$

### Comments of Examiners

Many candidates answered this question correctly. Some common errors observed were:

- Output in most cases was written inside the loop and the break and flag statements were not written.
- “Not a pronic number” was not printed,
- In the *if* condition, instead of  $(i*((i+1))==num)$ ,  $(i*i+1==num)$  was written.
- In the *if* statement, instead of the relational operator ( $==$ ), assignment operator ( $=$ ) was used.
- Mentioning that *else* statement inside the loop results in printing both the statements “Pronic Number” and also “Not a pronic number”.

### Suggestions for teachers

- Train students about the use of *for* loop.
- Give sufficient practice on problems to print an appropriate output message.
- Explain Operator precedence and importance of use of parenthesis in proper place.
- The effect of not putting parenthesis should be shown on the computer.
- Teach students the use of *break* statement to come out of the loop.
- Advise the students to read the questions carefully.

## MARKING SCHEME

### Question 5

```
import java.util.*;
class pronic {
    public static void main(String arg[])
    {
        Scanner sc = new Scanner(System.in);
        System.out.println("enter the number");
        int num=sc.nextInt();
        int flag=0;
        for(int i=0;i<num;i++)
        {
            if( i*(i+1)==num)
            {
                flag=1;
                break;
            }
        }
        if(flag==1)
            System.out.println("Pronic number");
        else
            System.out.println("Not a pronic number");
    }
}
```

#### Alternate method

```
class Q5
{
    public void pronic (int n)
    {
        int s=(int)Math.sqrt(n);
        if(s*(s+1)==n)
            System.out.println(n +" is a pronic number");
        else
            System.out.println(n+" is not a pronic number ");
    }
}
Variable description
```

STEPS
Input
flag initialisation
for loop
if condition
flag=1
Jump statement break
if else and displaying message
Variable description/Mnemonic codes

## Question 6

[15]

Write a program in Java to accept a string in lower case and change the first letter of every word to upper case. Display the new string.

Sample input: we are in cyber world

Sample output: We Are In Cyber World

### Comments of Examiners

Majority of the candidates were unable to answer this question correctly. Some of the commonly made mistakes were:

- Not taking care of the index of charAt() or substring() function.
- Incorrect initial and final values of the for loop-forr(int i=1; i<=l;i++).
- Accessing first character after space was done by writing (ch+1).
- Converting the first character after space was done as:  
ch.toUpperCase() or (ch+1).toUpperCase()
- Displaying output without space between the words.
- Getting an incorrect output as only the first character was converted and displayed.

### Suggestions for teachers

- Illustrate to the students that in the String, since position starts with 0, so condition should be either  $i < s.length()$  or  $i \leq s.length() - 1$ .
- Explain practically to the students how to access character after space.
- Give adequate number of exercises on String and Character functions to avoid confusion with the students when they use it in the program.

## MARKING SCHEME

### Question 6

```
import java.util.*;
class ques6 {
public static void main(String args[]) {
Scanner obj= new Scanner(System.in);
String s="";
System.out.println(" enter a string");
String sen=obj.nextLine();
sen=' ' +sen;
int l= sen.length();
for(int x= 0; x<l; x++) {
char ch= sen.charAt(x);
if (ch==' ') {
```

```

int i= sen.charAt(x+1);
i=i-32;
s= s+ '+' + (char)i;
x++; }
else s=s+ch;}
System.out.print(s);
}}

```

**Alternate Method**

// To change the first letter of each word to uppercase

```

import java.util.*;
class Q6_Title_Case
{
    public static void main(String args[])
    {
        Scanner sc= new Scanner(System.in);
        String sent,x="";
        char k , p ;
        int len, i;
        System.out.print("Enter a sentence");
        sent= sc.nextLine();
        sent= sent.toLowerCase();
        sent= " " + sent;
        len = sent.length();
        for(i=0;i<len-1;i++)
        {
            k=sent.charAt(i);
            p=sent.charAt(i+1);
            if(k==' ' && p!=' ')
                p=Character.toUpperCase(p);
            x=x+p;
        }
        System.out.println("The new string is "+ x);
    }
}

```

**Alternate Method**

```

class change
{
    void change( String s)
    {
        int l=s.length(),p,i;
        String str="";
        s=s.toLowerCase();
        s=s+" ";
        for(i=0;i<l;i++)
        {

```

STEPS
Input
Declaration of variable
Finding length
Logic
Output
Variable description/Mnemonic codes

```
p=s.indexOf(" ",i);
str=s.substring(i,p);
System.out.print(Character.toUpperCase(str.charAt(0))+str.substring(1)+" ");
i=p;
}}}
```

[15]

## Question 7

Design a class to overload a function volume() as follows:

- (i) double volume (double R) – with radius (R) as an argument, returns the volume of sphere using the formula.

$$V = \frac{4}{3} \times \frac{22}{7} \times R^3$$

- (ii) double volume (double H, double R) – with height(H) and radius(R) as the arguments, returns the volume of a cylinder using the formula.

$$V = \frac{22}{7} \times R^2 \times H$$

- (iii) double volume (double L, double B, double H) – with length(L), breadth(B) and Height(H) as the arguments, returns the volume of a cuboid using the formula.

$$V = L \times B \times H$$

## Comments of Examiners

Most candidates answered correctly. However, a few general mistakes were observed in all the three functions:

- Declaration of the variable V, was missing.
- Void was used in the prototype, but return statement was used in the body of the function.
- Double was used in the prototype, but System.out.println() was used in the body of the function and 'return' statement was missing.
- Void with System.out.println() was used. Though it shows no error, the combination is not as per the requirement in the question.
- In the keyword return R was capital.
- In the formula, instead of  $R * R * R$ ,  $R^3$  was written.
- Function name was not as per the question.
- Function overloading concept was not followed.
- Three different functions with different names were used.

## Suggestions for teachers

- Explain the topic on Functions with the various ways in which the functions can be used, step-by-step - without passing arguments; with arguments; without return data type and return statement; using return data type and return statement.
- Illustrate basic syntax of function in detail such as when 'void' return type is mentioned, the function has to have output statement in the function and when, instead of void, any other data type is written then return jump statement should be there in function.
- Teach the concept of rules of function overloading.
- Lay stress on writing keyword return, in small case.

Instruct students to place proper operators while writing Java expression in the program

## MARKING SCHEME

### Question 7

```
import java .io.*;
public class mensuration
{
    double volume(double R)
    {
        double V= 4.0/3* 22.0/7*Math.pow(R,3);
        return(V);
    }
    double volume( double H,double R)
    {
        double V= 22.0/7 * Math.pow(R,2) *H;
        return(V);
    }
    double volume( double L, double B , double H)
    {
        double V = L*B*H;
        return( V);
    }
}
```

STEPS
3 Functions with correct argument
Formulas in all the 3 functions
return statements in all the 3 functions
Variable description/Mnemonic codes

### Question 8

[15]

Write a menu driven program to display the pattern as per user's choice.

**Pattern 1**

ABCDE

ABCD

ABC

AB

A

**Pattern 2**

B

LL

UUU

EEEE

For an incorrect option, an appropriate error message should be displayed.

## Comments of Examiners

Majority of the candidates attempted well barring a few exceptions who showed following lapses:

- Menu was not displayed.
- Input for choice was not taken.
- Pattern printing was done using multiple print statements.
- Nested loop logic was not used,
- To start with new line, `println()` was missing,.
- ASCII value was used to print ABCDE but was not converted properly. (Loop value started with 65 and was checked for `<= 69`, but conversion to character was incorrect).
- Two separate classes were written instead of one.
- In the statements, `switch` and `default`. first character `s` and `d` were in capital letters.

### Suggestions for teachers

- Give sufficient practice for writing menu – driven programs.
- programs on pattern printing and series calculation which require nested looping.
- Teach problems involving conversion of character to its ASCII value and vice-versa.
- Give adequate practice of nested for loop, show the working and effect on the output when condition is changed. i.e. instead of `<` when `>` sign is used.
- Allow students to do lot of programs with `String` and nested for and show output on the computer when `println()` is missing.
- Provide adequate practice to the students on Syntax of `switch` case.
- Lay stress on keywords, which have to be written in small case.

## MARKING SCHEME

### Question 8

```
import java .util.*;
class series
{
    public static void main(String arg[])
    {
        Scanner sc = new Scanner(System.in);
        System.out.println("1.Series1 2.series2");
        System.out.println("enter the choice");
        int choice=sc.nextInt();
        switch(choice)
        {
            case 1:String s="ABCDE";
                for(int i=s.length();i>0;i--)
                {
```

#### STEPS

Displaying menu, input choice, switch
Initialising/ input string in case 1
for loop with length of string
Display and substring function
Initialising string in case 2/Input
Outer for loop
Inner for loop
Extraction , and display
<code>println()</code>
Default
Variable description/Mnemonic codes

```

        System.out.println(s.substring(0,i));
    }
    break;
case 2 :String s1="BLUE";
    for(int i=0;i<s1.length();i++)
    {
        for(int j=0;j<=i;j++)
        { System.out.print(s1.charAt(i)); }
        System.out.println();
    }
    break;
default :System.out.println("invalid choice");
} }
}

```

## Question 9

[15]

Write a program to accept name and total marks of **N** number of students in two single subscript array **name[ ]** and **totalmarks[ ]**.

Calculate and print:

- (i) The average of the total marks obtained by **N** number of students.  
[average = (sum of total marks of all the students)/**N**]
- (ii) Deviation of each student's total marks with the average.  
[deviation = total marks of a student – average]

## Comments of Examiners

Most of the candidates wrote the program correctly. However, a few could not attempt it correctly due to following miscues:

- Value of **N** (size of the array) was taken after arrays, name and total marks were created
- While accepting the values into the array, the index number was missing. Instead of name[i], candidates wrote name[]= sc.next().
- All operations i.e. input, calculating an average and deviation was done in one Loop.
- The average of the total marks obtained by **N** number of students was calculated inside for loop.
- Deviation of each student's total marks with average marks was not Displayed, rather it was printed outside for loop.

## Suggestions for teachers

- Allow plentiful practice on Array creation with different data types.
- Give sufficient practice of using array concept, accepting the values into the array, and printing the values from the array.
- Clarify the concept of array index.
- Teach using arrays, not only searching and sorting techniques but other general programs like, billing for many customers, pay slips for many employees etc.

## MARKING SCHEME

### Question 9

```
import java .io.*;
import java.util.*;
class ques9
{
public static void main(String aa[])
{
Scanner obj = new Scanner(System.in);
int n;
System.out.println("enter number of students");
n= obj.nextInt();
String name[] = new String[n];
double marks[] = new double[n];
double ave, total=0.0;
for(int x=0; x<n; x++)
{
System.out.println(" enter name and totalmarks");
name[x]= obj.nextLine();
marks[x]= obj.nextDouble();
total=total+ marks[x];

}
ave= total/n;
System.out.println("Name " + " " + " Deviation");
for(int x= 0; x<n;x++)
System.out.println( (marks[x]-ave));
}
}
```

STEPS
Accepting n, total=0.0
Declaring two array
Loop
Accepting value in array
total=total + marks ave = total/n
Output headings, loop for output
Output with calculation deviation
Variable description/Mnemonic codes

## GENERAL COMMENTS

### Topics found difficult/ confusing by candidates

- Library functions - its appropriate use.
- Arrays used with commercial / calculation-based programs.
- Menu driven program with pattern printing.
- Identifying the return data type of the library functions-print( ) and println( )
- Confusion in the use of print() and println() functions.
- Conversion of if-else –if to switch-case
- Output questions.
- CompareTo() function and its working.
- Syntax of switch-case and array declaration and initialization.
- Pattern printing using nested loop.
- String and Character functions.
- Return a value from a user-defined function.
- Using the key terms in definitions while writing the differences.
- Output of Math.floor() of negative values.
- Post increment and shorthand assignment operator.
- How to convert first character of a word to uppercase after space.

### Suggestions for candidates

- Revise and integrate the concepts studied in Class IX with the Class X syllabus.
- Avoid selective study.
- Give equal importance to all the topics mentioned in the syllabus.
- Practise each topic/sub –topic with as many examples as possible.
- Being an application-oriented subject, apply what is taught in the program and explain its outcome.
- Give sufficient practice to output-based questions.
- Understand all key terms and definitions comprehensively and learn.
- Learn correct use of all statements to eliminate syntax errors.
- Practise programs on various types of loops, their working, conversion from one loop to another and working of nested loop.
- Use proper variable names and ensure that every program has a variable description table.
- Avoid writing abbreviations like SOP, SOPLN, PSVM.
- Understand the logic of a program instead of memorising it.
- Explain programs using Mnemonic variables and comments.
- After writing the program, dry run it with different inputs.
- Write variable description /Mnemonic codes for every program.
- Practise previous years' ICSE question papers,
- Read the questions carefully and write the answers according to their requirements.
- Utilize the reading time to clearly understand the nature of the question.