ICSE 2025 EXAMINATION

Sample Question Paper – 3

Computer Applications

Time: 2 Hours **Max. Marks: 100**

General Instructions:

1. Answers to this Paper must be written on the paper provided separately.
2. You will not be allowed to write during the first 15 minutes.
3. This time is to be spent in reading the question paper.
4. The time given at the head of this Paper is the time allowed for writing the answers.
5. This Paper is divided into two Sections.
6. Attempt all questions from Section A and any four questions from Section B.
7. The intended marks for questions or parts of questions are given in brackets [ ].

SECTION A

Attempt all questions from this part.

QUESTION 1.
Choose the correct answer and write the correct option.

(Do not copy the question, write the correct answers only.)

(i) What will this code produce as output ?

```
int a[] = {2, 4, 6, 8, 10};
a[0] = 23;
a[3] = a[1];
int c = a[0] + a[1];
System.out.println("Sum = " + c);
```

(a) 26 (b) 23
(c) 25 (d) 27

Answer: (d) 27

(ii) The process of binding the data and method together as one unit is called as

(a) Dynamic binding (b) Inheritance
(c) Polymorphism (d) Encapsulation

Answer: (b) Inheritance

(iii) A collection of classes is known as a ____.

(a) Package (b) Function
(c) Method (d) Wrapper class

Answer: (a) Package

(iv) What is the return type of the following function?
is Letter Or Digit (char)

(a) String (b) Char
(c) Integer (d) Boolean

Answer: (d) Boolean

(v) What will the following code output?

```
String str = "Computer Applications" + 1 + 0;
System.out.println("Understanding " + str);
```

(a) Understanding Computer Applications 10
(b) Understanding Computer Application 20
(c) Understanding Computer Application 10
(d) Error

Answer: (a) Understanding Computer Applications 10

(vi) What is the output of the following code.

```
String s1 = "welcome";
String s2 = "island";
System.out.println(s1.substring(0).concat(s2.substring(2)));
System.out.println(s2.toUppercase());
```

(a) Welcome-land ISLAND (b) Welcome Land Island
(c) WELCOME LAND (d) Welcome land ISLAND Island

Answer: (a) Welcome-land ISLAND

(vii) This term refers to the name assigned to a package, class, interface, method, or variable.

(a) Keyword (b) Token
(c) Literal (d) Identifier

Answer: (d) Identifier

(viii) These constructors require parameters to be provided when creating objects.

(a) Copy
(b) Default
(c) Parameterized
(d) None of the above

Answer: (c) Parameterized

(ix) int code[ ] = [25, 37, 38, 42];
The given statement

(a) assigns 37 to code [1]
(b) assigns 25 to code [1]
(c) assigns 38 to code [3]
(d) assigns 42 to code [0]

Answer: (a) assigns 37 to code [1]

(x) The automatic conversion of a primitive data type into an object of its corresponding wrapper class is called:
(a) auto-boxing
(b) explicit conversion
(c) shifting
(d) None of the above

Answer: (a) auto-boxing

(xi) The parseInt ( ) function is a member of
(a) integer wrapper class
(b) character wrapper class
(c) boolean wrapper class
(d) None of these

Answer: (a) integer wrapper class

(xii) ____members are accessible inside their own class, classes within the package and subclasses.
(a) Private
(b) Protected
(c) Public
(d) None of these

Answer: (b) Protected

(xiii) Iteration is also known as
(a) looping
(b) repetition
(c) Both (a) and (b)
(d) None of these

Answer: (c) Both (a) and (b)

(xiv) ____are reference types, which hold the reference id of a memory location.
(a) Composite types
(b) Primitive types
(c) Attribute types
(d) None of the above

Answer: (a) Composite types

(xv) Classes that inherit from Runtime-exception are referred to as:

(a) Checked exceptions (b) Unchecked exceptions
(c) Impure exceptions (d) Static exceptions

Answer: (b) Unchecked exceptions

(xvi) Which of the following are invoked directly when an object is created?

(a) Methods (b) Arrays
(c) Constructors (d) Strings

Answer: (c) Constructors

(xvii) Assertion (A) : Array is a data type which can store multiple homogenous variables.
Reason (R) : Elements of array are stored in an indexed manner starting with index 0.
(a) Both Assertion (A) and Reason (R) are true and Reason (R) is a correct explanation of Assertion (A).
(b) Both Assertion (A) and Reason (R) are true and Reason (R) is not a correct explanation of Assertion (A).
(c) Assertion (A) is true and Reason (R) is false.
(d) Assertion (A) is false and Reason (R) is true.

Answer: (b) Both Assertion (A) and Reason (R) are true and Reason (R) is not a correct explanation of Assertion (A).

(xviii)Read the following text and choose the correct answer.
Abstraction is a key concept in Object-Oriented Programming (OP) that focuses on representing essential features while omitting unnecessary background details or explanations..

What is an abstraction?
(a) Abstraction is more about 'What' a class can do.
(b) Abstraction is more about 'How' to achieve that functionality.
(c) It binds data and methods in a single unit.
(d) It implements using private access modifier.

Answer: (a) Abstraction is more about 'What' a class can do.

(xix) Assertion (A) : The factory of object means a factory that produces the objects.
Reason (R) : Class is known as object factory because single class generates a lot of objects.
(a) Both Assertion (A) and Reason (R) are true and Reason (R) is a correct

explanation of Assertion (A).
(b) Both Assertion (A) and Reason (R) are true and Reason (R) is not a correct explanation of Assertion (A).
(c) Assertion (A) is true and Reason (R) is false.
(d) Assertion (A) is false and Reason (R) is true.

Answer: (a) Both Assertion (A) and Reason (R) are true and Reason (R) is a correct explanation of Assertion (A).

(xx) Which of the following statement is incorrect?
(a) Array can be initialized when they are declared.
(b) Array can be initialized using 'comma' separated expressions surrounded by curly braces.
(c) It is necessary to use 'new' operator to initialize an array.
(d) None of the above

Answer: (a) Array can be initialized when they are declared.

QUESTION 2.

(i) If int a[ ] = {7, 3, 4, 8, 9, 2}; what are the values of x and y?

(a) x=a[1]*a[0]+a[3]
(b) y=a. length

Answer:

```
(a) x = a[1]*a[0]+a[3]
    a[1] = 3, a[0] = 7, and a[3] = 8
      = 3*7+8
      = 21+8
    x = 29
(b) y = a.length
The length of array a is the total
number of elements in it, which is 6.
So, y = 6.
```

(ii) Determine the values of n and m after executing the following code:

```
int m;
int n;
m=5;
n=(5*++m)%3;
System.out.println("n="+n+"m="+m);
```

Answer:
n=0
m=6

(iii) Identify the keyword that:

(a) Indicates an error has occurred during an input/output operation.

(b) Differentiates between an instance variable and a class variable.

Answer:
(a) Throws
(b) Static

(iv) What will be the outcome of the following statements?

```
int a=3;
System.out.println(" "+(1+a));
System.out.println(" "+1+a);
```

Answer:
Output
4
13

(v) Name the primitive data type in Java that is a 64 bits integer and is used when you need a range of values under than those provided by int.

Answer: long

(vi) Give the prototype of a function search, which receives a sentence sentc and a word wrd and returns 1 or 0.

Answer: int search (string sentc, String wrd)

(vii) Determine the output of the following code.

```
String str = "I Love My Family";
System.out.println(Integer.toStr
                ing(str.length()));
System.out.println(str.substring
                        (12));
```

Answer:
Output
16
Family

(viii) Write a Java statement for the following mathematical expression :

$$V = \frac{1}{3}\pi r^2 h$$

Answer: double V = (1.0/3) * Math.PI*r*r * h

(ix) What will be the output of the following expressions?

(a) Math.pow(3.4, 2)+2*Math. sqrt(64)
(b) Math.ceil(3.4)+2*Math. floor(3.4)+2

Answer:
(a) 27.56
(b) 12.0

(x) How many times will the loop in the following code execute?

```
int x=2, y=50;
do
{
    ++x;
  y— =x++;
}while (x<=10);
return y;
```

Answer: The loop will be executed 5 times.

SECTION B

Attempt any four questions from this section.

QUESTION 3.
Write a program to enter a number and check whether the number is Neon or not. A number is said to be Neon if sum of the digits of the square of the number is equal to the number itself.

e.g.Input : 9

Output : 9*9 = 81, 8+1 = 9
9 is Neon number.

Answer:

```java
import java.util.Scanner;

public class NeonNumber {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        System.out.print("Enter a number: ");
        int number = scanner.nextInt();

        int square = number * number;
        int sumOfDigits = 0;

        while (square > 0) {
            sumOfDigits += square % 10;
            square /= 10;
        }

        if (sumOfDigits == number) {
            System.out.println(number + " is a Neon number.");
        } else {
            System.out.println(number + " is not a Neon number.");
        }
    }
}
```

QUESTION 4.

Write a program to print following patterns.

(i)
```
1 2 3 4 5
1 2 3 4
1 2 3
1 2
1
```

(ii)
```
A
B A
C B A
D C B A
E D C B A
```

Answer:

```java
public class KboatPattern
{
    public static void main(String args[]) {
        System.out.println("Pattern 1: ");
        for (int i = 5; i >= 1; i--) {
            for (int j = 1; j <= i; j++) {
                System.out.print(j + " ");
            }
```

```
            System.out.println();
        }
        System.out.println();
        System.out.println("Pattern 2: ");

        char ch = 'A';
        for (int i = 0; i < 5; i++) {
            for (int j = i; j >= 0; j--) {
                System.out.print((char)(65 + j) + " ");
            }
            System.out.println();
        }
    }
}
```

QUESTION 5.

Create a class The-string that takes a string input of up to 100 characters and counts the number of vowels, consonants, digits, and spaces in the provided sentence using if-else statements in Java.

Answer:

```
import java.util.Scanner;

public class The_string {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.println("Enter a string (up to 100 characters):");
        String input = scanner.nextLine();
        scanner.close();

        if (input.length() > 100) {
            System.out.println("Input exceeds 100 characters. Please try
again.");
            return;
        }

        int vowels = 0, consonants = 0, digits = 0, spaces = 0;
        input = input.toLowerCase();

        for (int i = 0; i < input.length(); i++) {
            char ch = input.charAt(i);
```

```
            if (ch >= 'a' && ch <= 'z') {
                if (ch == 'a' || ch == 'e' || ch == 'i' || ch == 'o' ||
ch == 'u') {
                    vowels++;
                } else {
                    consonants++;
                }
            } else if (ch >= '0' && ch <= '9') {
                digits++;
            } else if (ch == ' ') {
                spaces++;
            }
        }

        System.out.println("Vowels: " + vowels);
        System.out.println("Consonants: " + consonants);
        System.out.println("Digits: " + digits);
        System.out.println("Spaces: " + spaces);
    }
}
```

QUESTION 6.

Define a class StringMinMax to identify the smallest and the largest word present in the string.

e.g. Input: Hello this is wow world

Output:Smallest word: is Largest word: Hello

Answer:

```
public class StringMinMax {
    public static void main(String[] args) {
        String input = "Hello this is wow world";
        findMinMaxWords(input);
    }
```

```
    public static void findMinMaxWords(String str) {
        String[] words = str.split(" ");
        String smallest = words[0];
        String largest = words[0];

        for (int i = 1; i < words.length; i++) {
            if (words[i].length() < smallest.length()) {
                smallest = words[i];
            }
            if (words[i].length() > largest.length()) {
                largest = words[i];
            }
        }

        System.out.println("Smallest word: " + smallest);
        System.out.println("Largest word: " + largest);
    }
}
```

QUESTION 7.

Define a class to accept values in integer array of size 10. Find sum of one digit number and sum of two digit numbers entered. Display them separately.

Example : Input

a [ ] ={2, 12, 4, 9, 18, 25, 3, 32, 20, 1}

Output Sum of one digit numbers

$$2 + 4 + 9 + 3 + 1 = 19$$

Sum of two digit numbers

$$12 + 18 + 25 + 32 + 20 = 107$$

Answer:

```java
import java.util.Scanner;

public class NumberSumCalculator {
    public static void main(String[] args) {
        int[] numbers = new int[10];
        Scanner scanner = new Scanner(System.in);

        // Input values
        System.out.println("Enter 10 integer values:");
        for (int i = 0; i < 10; i++) {
            numbers[i] = scanner.nextInt();
        }

        // Calculate sums
        int sumOneDigit = 0;
        int sumTwoDigit = 0;

        for (int num : numbers) {
            if (num >= 0 && num < 10) {
                sumOneDigit += num;
            } else if (num >= 10 && num < 100) {
                sumTwoDigit += num;
            }
        }
        // Display results
        System.out.println("Sum of one digit numbers: " + sumOneDigit);
        System.out.println("Sum of two digit numbers: " + sumTwoDigit);

        scanner.close();
    }
}
```

QUESTION 8.

Write a program to input integer elements into an array of size 20 and perform the following operations :

(i) Display largest number from the array.

(ii) Display smallest number from the array.

(iii) Display sum of all the elements of the array.

Answer:

```java
import java.util.Scanner;

public class ArrayOperations {
    public static void main(String[] args) {
        int[] arr = new int[20];
        Scanner scanner = new Scanner(System.in);

        // Input array elements
        System.out.println("Enter 20 integer elements:");
        for (int i = 0; i < 20; i++) {
            arr[i] = scanner.nextInt();
        }

        // Find largest and smallest numbers
        int largest = arr[0];
        int smallest = arr[0];
        int sum = arr[0];

        for (int i = 1; i < arr.length; i++) {
            if (arr[i] > largest) {
                largest = arr[i];
            }
            if (arr[i] < smallest) {
                smallest = arr[i];
            }
            sum += arr[i];
        }

        // Display results
        System.out.println("Largest number: " + largest);
        System.out.println("Smallest number: " + smallest);
        System.out.println("Sum of all elements: " + sum);

        scanner.close();
    }
}
```