

# Outline of Test-A (Chapters 1-3)

---

- **Group-A (10%)**

Examine the following statements and put a check mark  in the square of the correct ones. Note that if you are not sure about a statement then leave its square blank because you lose points for checking an incorrect statement. Example:

- The `print` method in `type.lang.IO` is overloaded
- `class` is a reserved word in Java
- In Java, the acronym *VM* stands for virtual memory.

- **Group-B (20%)**

For each question, write in the box the output of the shown fragment. If you believe the fragment has errors, identify the error; specify its type (syntax or runtime); and write a brief yet complete explanation. You can assume all needed packages have been imported. Example:

```
int a = 30;
int b = 28;
a = a + 5 / 2;
b = b % 2 - b % 3;
IO.println(a);
IO.println(b);
```

- **Group-C (30%)**

Given the API of a class, answer questions about it and write an app that uses it.

- **Group-D (40%)**

Write an app that satisfies a given specification.

**Note:**

*You are assumed to have memorized the names of the primitive types, the arithmetic and relational operators, the assignment algorithm, and the main features of the API of the classes: `type.lang.IO`, `type.lang.SE`, `java.lang.Math`. Nevertheless, the following sheet will be provided as part of the test:*

# Data Sheet for Test-A

---

## **Primitive Types:**

byte [-128,+127], short, char, int, long, float, double, boolean

## **Arithmetic Operators:**

+ - \* / % ++ --

## **Relational Operators:**

< <= > >= == !=

## **The SE class**

```
public static void require(boolean condition, String msg)
```

*Do nothing if the condition is true. Otherwise, terminate the app and print the msg.*

## **The IO class:**

*For unformatted output, use one of the following two methods passing the value to be printed:*

```
public static void print(anything)
public static void println(anything)
```

*For formatted output, use one of the following two methods passing the value to be printed and the desired format descriptor:*

```
public static void print(anything, String format)
public static void println(anything, String format)
```

*For input, use the static method: readX(), where X is Byte, Short, Char, Int, Long, Float, Double, Boolean, or Line.*

## **The Math class:**

```
public static double abs(double x)
public static double pow(double x, double y)
public static double rint(double x)
public static double floor(double x)
public static double ceil(double x)
```