1. Basics

1.1 Variables

- Non-decimal number systems. Transfer of a number from a non-Decimal number system to the Decimal and vice versa.
- Declaration and initialization of variables.
- Primitive types: 4 integer types, 2 floating point types, boolean, char.
- Integer types. Size of integer types, understanding the bounds of integer types. Binary representation. Arithmetical operations, binary operations over them. Operations over variables from the different integer types.
- Assignment operator, assignment operator with the other operations. Increment/decrement, prefix and postfix.
- Floating point types. Arithmetical operations. Arithmetical operations over different numeric types including integer types.
- Character type.
- Boolean type. Logical operations, difference between “&” and “&&” or ”|” and “||”. Operations mapping numeric types (including char) to boolean, like “==”, “>” “<”.

1.2 Conditional statements

- If operator, syntax. If, else if, else.
- Switch operator, syntax. Work principals. Case with Break and without. Equality of Cases.

1.3 Loops

- While loop and do-while loop. Difference between them.
- For loop, syntax. Transform For loop into While loop.
- Break and Continue operators. Break and Continue with a Label.

1.4 Arrays

- One-dimensional arrays. Declaration, initialization. Arrays functionality. Max index.
- N-dimensional arrays. Declaration, initialization. Max dimensionality.

1.5 Functions (aka static methods)

- Functions, return statement.

2. Java Core

2.1 OOP

- Class understanding, objects. Object fields, object methods. Static modifier, Fields and Methods of a Class (not depending on an object).
- Reference types. Addressing to the fields/methods of an object/class.
- Inheritance mechanism in java. Types of inheritance. Interfaces, Abstract Classes, difference. Amount of inherited Classes and Interfaces.
- Access modifiers: public, private, protected, package-private (no modifier). Encapsulation understanding.
- Polymorphism understanding. Two types of polymorphism in Java. Method overriding and overloading.
- Object class as the superclass of all classes in Java. Understanding the method toString.
- Generics.
- Enums, functions in enums, constructors in enums, inheritance with enums.

2.2 Patterns

- Comparator, Comparable.
- Iterator, Iterable, syntax “for each”.

2.3 Data structures and basic algorithms

- Complexity, big O notation.
- Data structure “List”. Implementation via array, “ArrayList” class in Java Collections API. Complexity of all common methods. Implementation via references, “LinkedList” class in Java Collections API.
- Data structures “Queue” and “Stack”. Implementation via array with moving pointer to the first element. Java classes: “LinkedList” and “ArrayDeque”.
- Data structure “Map”. Understanding Object.equals and Object.hashCode methods. The rules about their relation in case of overriding the standard ones. Implementations via Hash Table and Binary Tree. Java classes: “HashMap”, “TreeMap”. Complexity of commonly used operations.
- Data structure “Set”. Implementations inherited from the “Map”. Complexity of commonly used operations. Uniq functionality over sets (intersection, union and subtraction). Java classes: “HashSet”, “TreeSet”.
- Understanding Binary search. Java implementation.
- Understanding recursion.

2.4 Necessary Libraries

- Class “String”. Commonly used methods.
- Work with concatenation of strings. StringBuilder, StringBuffer, the key difference between them. Commonly used methods.
- Classes-wrappers over the primitives. What for, commonly used methods (like Integer.parseInt, Integer.valueOf etc.). Transformation of a string to a number and vice versa.
- Class “Arrays”, as a utility for working on arrays.
- Class “Collections”, as a utility for working on collections.
- Basic working with files. “FileInputStream”, “FileOutputStream”.
- “PrintWriter”, “PrintStream” for working with strings. The key difference.
- “BufferedReader”

2.5 Multithreading

- Understanding a thread, class “Thread” in java, “Runnable” interface. Starting a new thread.
- Thread interruption mechanism, InterruptionException, Thread.sleep method.
- Daemon threads, methods “Thread.join”, Thread.sleep.
- Definition of a deadlock.
- Definitions of “Mutex” and “Monitor”. Which objects better to use as a mutex. Synchronization mechanism in Java. Methods “Object.wait”, “Object.notify”, “Object.notifyAll”.
- Understanding concurrency collections. Blocking queues. Alternative methods working with synchronization via Concurrency API in Java.
- Atomic variables in Java.
- Executors. Running a bunch of tasks via an executor. Running a repeated task via executor.

2.6 New features in Java 1.8.

- Java 8 DateTime API. LocalDate, LocalDateTime, ZonedDateTime and operations over them. “Period”, “ChronoUnit”.
- Lambdas, method references.
- Preducates, Functions, Operators, Suppliers and Consumers.
- Stream API. Ways to create a stream (from an array, collection etc.)
- Methods of streams: Intermediate operations and Terminal operations. Transforming Int, Long or Double streams to general streams and vice versa.
- Collectors, included in Java collectors (groupingBy etc.).