Question 1:

An algorithm is a step-by-step procedure or set of rules used to solve a specific problem or perform a particular task. It is a well-defined sequence of instructions that can be executed by a computer or followed by a human to solve a problem. Algorithms provide a systematic approach to problem-solving, allowing for efficient and reliable solutions.

Question 2:

Pseudocode is a high-level, informal representation of an algorithm. It is a way to express the logic of an algorithm using a combination of natural language and programming language-like constructs. Pseudocode is not tied to any specific programming language syntax, which makes it more accessible and easier to understand for both developers and non-developers.

Pseudocode is useful in the development of algorithms because it allows programmers to plan and design the solution to a problem before implementing it in a specific programming language. It serves as a blueprint for the algorithm, helping to clarify the overall structure and flow of the solution. Pseudocode can be used to communicate and collaborate with others, aiding in the understanding and review of the algorithm's logic before writing the actual code.

Question 3:

In Java, a block statement, also known as a block of code, is a group of zero or more statements enclosed in curly braces ({ }). It is used to create a compound statement, where multiple statements are treated as a single unit. Block statements are used to define the scope of variables, control the flow of execution, and organize code in a structured manner.

Block statements in Java are commonly used in control structures such as if-else statements, loops (for, while, do-while), and method bodies. They allow for the execution of multiple statements within a specific scope, ensuring that the statements inside the block are treated as a cohesive unit.

Question 4:

The main difference between a while loop and a do..while loop lies in when the loop condition is checked.

In a while loop, the condition is evaluated before the loop body is executed. If the condition is false initially, the loop body is skipped entirely, and the program moves on to the next statement after the loop.

In a do..while loop, the loop body is executed at least once, regardless of the condition. After executing the loop body, the condition is evaluated. If the condition is true, the loop body is executed again, and the process continues. If the condition is false, the loop is terminated, and the program moves on.

In summary, the key difference is that a while loop checks the condition before executing the loop body, while a do. While loop checks the condition after executing the loop body.

Question 5:

To "prime a loop" means to initialize the loop control variable before entering the loop. The loop control variable is typically used to keep track of the loop's progress and determine when the loop should terminate.

Priming a loop involves assigning an initial value to the loop control variable before the loop starts. By doing this, you ensure that the loop has a valid starting point and that the loop condition can be properly evaluated. Without priming the loop, the loop control variable might have an undefined value, leading to unpredictable behaviour or potential errors.

Priming a loop is important for loops that rely on the loop control variable's initial value or depend on the loop control variable to track iterations accurately.

Question 6:

An animation refers to a dynamic sequence of visual images or objects that creates an illusion of motion or change over time. It involves displaying a series of frames in rapid succession to create the perception of movement. Animations can be used for various purposes, including entertainment, user interfaces, educational content, and simulations.

In computer graphics, animations are typically created using a technique called frame-based animation. Each frame represents a specific moment in time, and by displaying frames

Question 7:

for (int i = 3; i <= 36; i += 3) {

System.out.print(i + " ");

}

Question 8:

public static void main(String[] args) {

System.out.print("Enter an integer: ");

int number = TextIO.getInt();

if (number % 2 == 0) {

System.out.println(number + " is even.");

} else {

System.out.println(number + " is odd.");

}

}

Question 9:

Random random = new Random();

int num1 = random.nextInt(10) + 1;

int num2 = random.nextInt(10) + 1;

while (num2 == num1) {

num2 = random.nextInt(10) + 1;

}

System.out.println("Random integers: " + num1 + " " + num2);

Question 10:

String s1 = "123";

String s2 = "456";

try {

int sum = Integer.parseInt(s1) + Integer.parseInt(s2);

System.out.println("Sum: " + sum);

} catch (NumberFormatException e) {

System.out.println("Error: Invalid input format.");

}

Question 11:

Output is:

2

4

8

16

32

Question 12:

Output is:

0

0

0

0

0

Question 13:

Output is:

R

M

N

Question 14:

int[] numbers = {10, 42, 15, 42, 8, 42, 25, 30};

int count = 0;

for (int i = 0; i < numbers.length; i++) {

if (numbers[i] == 42) {

count++;

}

}

System.out.println("The number 42 occurs " + count + " times in the array.");

Question 15:

double[] raceTimes = {9.5, 7.2, 8.9, 6.4, 9.1};

double minTime = raceTimes[0];

double maxTime = raceTimes[0];

for (int i = 1; i < raceTimes.length; i++) {

if (raceTimes[i] < minTime) {

minTime = raceTimes[i];

}

if (raceTimes[i] > maxTime) {

maxTime = raceTimes[i];

}

}

double range = maxTime - minTime;

System.out.println("The range of raceTimes is: " + range);