Infinite Loops

In this lesson, an explanation of how infinite loops might emerge in a loop is provided.

We'll cover the following

Emergence of infinite loops

One common programming mistake is to create an **infinite** loop. An **infinite** loop refers to a loop, which under certain valid (or at least plausible) input, will never **exit**.

> **Note:** Beginning programmers should be careful to examine all the *possible* inputs into a *loop* to ensure that for each such set of inputs, there is an **exit** condition that will eventually be reached.

Compilers, *debuggers*, and other *programming tools* can only help the programmer so far in detecting **infinite** loops.

> **Note:** In the fully general case, it is not possible to automatically detect an infinite loop. This is known as the **halting** problem.

While the halting problem is not solvable in the fully general case, it is possible to determine whether a loop will **halt** for some *specific* cases.

Uses of infinite loops

The conditions when infinite loops can be extremely useful are the following:

- When continuous input is required to the program or system, and it has to

respond accordingly without stopping.

- When we do not know the exit condition, some calculations need to be performed inside the loop, which will decide when to exit or terminate the loop.

Example of infinite loop

Below is an example of an *infinite* loop.

> **Note:** You will get an **error** when you try to run the code below because an infinite number of print statements cannot be shown here.

```
class HelloWorld {
    public static void main(String args[]) {
        boolean a = true;
        while (a) { // the while condition will always be met as will always return true
            System.out.println("Infinite loop");
        }
    }
}
```

In a normal case, the code above will print **"Infinite loop"** without stopping because the condition statement in the `while` loop will always evaluate to **true**. There is no point at which it'll return **false** hence we'll get stuck in an *infinite* loop, and the code will execute forever.

> **Note:** The generation of infinite loops is not constrained to only `while` loops. It can very well occur in other looping structures too.

Next up, some challenges are provided to test the understanding of loops.