Concurrency

Franck van Breugel

March 6, 2019

1 Predict the final value

1. One thread executes

   ```
   v = 1;
   v = v + 1;
   ```

   and another thread executes

   ```
   v = 0;
   ```

   What is the final value of v?

2. One thread executes

   ```
   v = v + 1;
   ```

   and another thread executes

   ```
   v = v + 1;
   ```

   If the initial value of v is 0, then what is the final value of v?

3. One thread executes

   ```
   v = 0;
   ```

   and another thread executes

   ```
   v = Long.MAX_VALUE;
   ```

   How many different final values can v have?

2 Printers

1. Develop a Java class called **`Printer`** that is a **`Thread`** and prints its name 1000 times.

```
public class Printer extends Thread {









}
```

2. Develop an app that creates two **`Printer`**s with names 1 and 2 and run them concurrently.

```
public class TwoPrinters {
  public static void main(String[] args) {









  }
}
```

3. Develop a Java class called **`Printer`** that implements **`Runnable`** and prints the thread's name 1000 times

```
public class Printer implements Runnable {
```

```
}
```

4. Develop an app that creates two **`Printer`**s (developed in 3.) with names 1 and 2 and run them concurrently.

```
public class TwoPrinters {
  public static void main(String[] args) {









  }
}
```

3 Incrementers

1. Develop a Java class called **`Incrementer`** that is a **`Thread`** and increments a shared static attribute named **`value`**, which is initialized to 0.

```
public class Incrementer extends Thread {









}
```

2. Develop an app that creates two **`Incrementer`**s and run them concurrently. Assert that the final value of **`value`** is two.

```
public class TwoIncrementers {
  public static void main(String[] args) {









  }
}
```

4 How many executions?

1. One thread prints 1 one. Another thread prints 1 two. How many different executions are there?

2. One thread prints 2 ones. Another thread prints 2 twos. How many different executions are there?

3. One thread prints 3 ones. Another thread prints 3 twos. How many different executions are there?

4. One thread prints 1000 ones. Another thread prints 1000 twos. How many different executions are there?

5. One thread executes n instructions. Another thread executes n instructions. How many different executions are there?

6. There are k threads. Each thread executes n instructions. How many different executions are there?