

Q10. What will the following code's output be in sequence and explain why?

```
function printNumber(num) {
```

```
    console.log(num);
```

```
}
```

```
console.log(1);
```

```
setTimeout(printNumber, 0, 2);
```

```
setTimeout(printNumber, 100, 3);
```

```
console.log(4);
```

A4: Same explanation as that of question 2 and question 8 of the test

Here, log(1) and log(4) will simply be pushed into the execution call stack, they will be executed seamlessly, that is why they are seen first. Logs for 2 and 3 will first be pushed into the call stack as an execution context by the javascript engine, the event loop will then push them in the call back queue after timer gets expired. Once the execution call stack is empty, logs for 2 and 3 will be pushed back into the call stack for execution.

Valid output:

1

4

2

3