

COMP1531

✓ Correctness - Code Coverage

Lecture 7.1

Author(s): Hayden Smith



[\(Download as PDF\)](#)

In This Lecture

- **Why?** 🤔
 - We need a mechanism to tell how much our tests actually test
- **What?** 📄
 - Coverage
 - Code Coverage
 - Coverage in jest



Coverage

A measure of how thorough our tests are.



Coverage

- **Test Coverage:** a measure of how much of the feature set is covered with tests. This is often left to human judgement
- **Code coverage:** a measure of how much code is executed during testing. This can be computed and quantified



Coverage

- Measure code coverage as a percentage of statements (lines) executed
- Can give us a good indication how much of our code is executed by the tests and most importantly highlight what has not been executed.

👁️ Let's Check Some Code Coverage!

We've got two pieces of sample code. Let's write tests with jest and the `--coverage` flag (in `package.json`) and use coverage to help us write better tests!



Example: Leap Years

```
1 export function isLeapYear(year: number) {
2   if (year % 4 !== 0) {
3     return false;
4   } else if (year % 100 !== 0) {
5     return true;
6   } else if (year % 400 !== 0) {
7     return false;
8   } else {
9     return true;
10  }
11 }
```

7.1_is_leap_year.ts



Example: Leap Years

```
1 // Given a number of days from
2 // January 1st 1970, return the year.
3
4 import { isLeapYear } from './7.1_is_leap_year';
5
6 function dayToyear(days: number) {
7   let year = 1970;
8
9   while (days > 365) {
10    if (isLeapYear(year)) {
11      if (days > 366) {
12        days -= 366;
13        year += 1;
14      }
15    } else {
16      days -= 365;
17      year += 1;
18    }
19  }
20
21  return year;
22 }
```



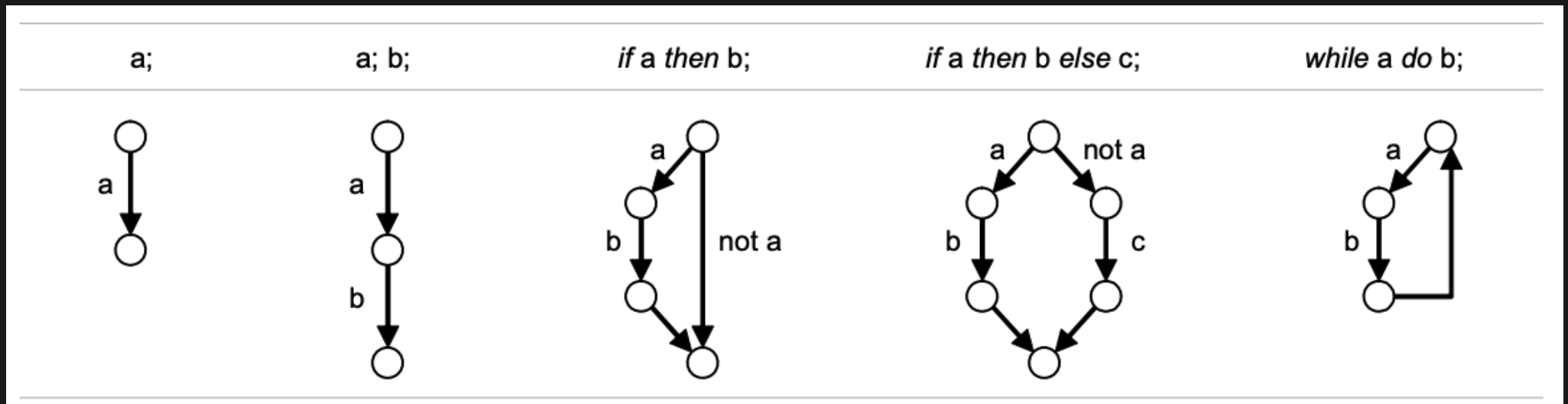

Coverage Command Summary

- Run `jest --coverage` to run tests with coverage on.
- Recommended to put this in a `package.json` script.
- You can access an HTML summary in `coverage/lcov-report/index.html`.



Branch Coverage Checking

- For lines that can potentially jump to more than one other line (e.g. if statements), check how many of the possible branches were taken during execution
- Done automatically with jest.
- Sometimes referred to as edge coverage.



Summary

- Code coverage is useful.
- It's more important to look at what's not covered than the coverage percentage.
- Branch coverage is a more accurate measurement so you should use it instead of statement coverage.
- Like all measurements, it's important to understand what meaning to attach to it.

Feedback



Or go to the [form here](#).

