Software Testing, Quality Assurance and Maintenance Winter 2017

Lecture 24 — March 6, 2017

Patrick Lam *version 0*

This material will come up later. I'm just putting it here right now, but I don't actually expect it to show up in Lecture 24. I want to talk more about engineering test suites first.

Using Linters

We will also talk about linters in this lecture, based on Jamie Wong's blog post `jamie-wong.com/2015/02/02/linters-as-invariants/`.

First there was C. In statically-typed languages, like C,

```
#include <stdio.h>

int main() {
  printf("%d\n", num);
  return 0;
}
```

the compiler saves you from yourself. The guaranteed invariant:

"if code compiles, all symbols resolve."

Less-nice languages. OK, so you try to run that in JavaScript and it crashes right away. Invariant?

"if code runs, all symbols resolve?"

But what about this:

```
function main(x) {
  if (x) {
    console.log("Yay");
  } else {
    console.log(num);
  }
}

main(true);
```

Nope! The above invariant doesn't work.

OK, what about this invariant:

> “if code runs without crashing, all symbols referenced in the code path executed resolve?”

Nope!

```
function main() {
  try {
    console.log(num);
  } catch (err) {
    console.log("nothing to see here");
  }
}

main();
```

So, when you’re working in JavaScript and maintaining old code, you always have to deduce:

- is this variable defined?
- is this variable always defined?
- do I need to load a script to define that variable?

We have computers. They’re powerful. Why is this the developer’s problem?!

```
1 function main(x) {
2   if (x) {
3     console.log("Yay");
4   } else {
5     console.log(num);
6   }
7 }
8
9 main(true);
```

Now:

```
$ nodejs /usr/local/lib/node_modules/jshint/bin/jshint --config jshintrc foo.js
foo.js: line 5, col 17, ’num’ is not defined.

1 error
```

Invariant:

> “If code passes JSHint, all top-level symbols resolve.”

Strengthening the Invariant. Can we do better? How about adding a pre-commit hook?

“If code is checked-in and commit hook ran,
all top-level symbols resolve.”

Of course, sometimes the commit hook didn’t run. Better yet:

- Block deploys on test failures.

Better invariant.

“If code is deployed,
all top-level symbols resolve.”

Even better yet. It is hard to tell whether code is deployed or not. Use git feature branches, merge when deployed.

“If code is in master,
all top-level symbols resolve.”