Cyclomatic Complexity
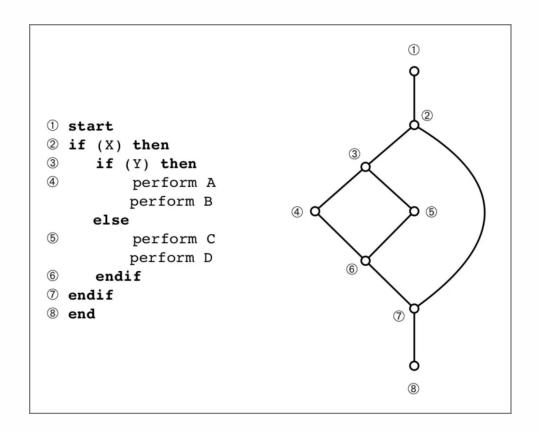
Yegor Bugayenko

Lecture #2 out of 24
80 minutes

1. Some programmers mistakenly feel proud of higher complexity of their code.

"Cyclomatic Complexity (CC) is a count of the number of decisions in the source code. The higher the count, the more complex the code. The formula is simple: $C = E - N + 2$."

— Thomas J. McCabe, Sr, 1976

What is the complexity of this program?

I found this picture here.

2. In his presentation "*Software Quality Metrics to Identify Risk*", Tom McCabe introduces the following categorisation to interpret cyclomatic complexity: 1–10 little risk, 11–20 moderate risk, 21–50 high risk, 50+ very high risk.

3. Graylin JAY et al., *Cyclomatic Complexity and Lines of Code: Empirical Evidence of a Stable Linear Relationship*, Journal of Software Engineering & Applications, Vol. 2, 2009, pp. 137–143

4. What is a complexity of a class? How about a module?

5. <u>Feature creep</u> is one of the most common sources of cost and schedule overruns; it can even kill products and projects — Wikipedia.

6. Tom McCabe suggested to prohibit functions where CC is larger than ten. Modern static analyzers may help you control this automatically.

Read this:

The Better Architect You Are, The Simpler Your Diagrams (2015)

Are You a Hacker or a Designer? (2014)