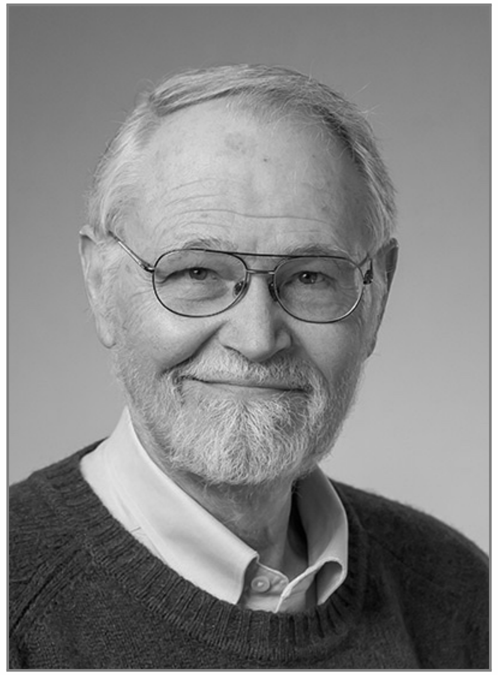
Code Style

Yegor Bugayenko

Lecture #22 out of 24
80 minutes



Brian Kernighan

> “The harder it is for people to grasp the intent of any given section, the longer it will be before the program becomes operational. Trying to outsmart a compiler defeats much of the purpose of using one. Write clearly — don’t sacrifice clarity for ‘efficiency.”’

— Brian W. Kernighan and Phillip James Plauger. *The Elements of Programming Style.* McGraw-Hill, Inc., 1974

Which One Is Better?

```
int f(int n)
{
  if (n == 1 || a < 2)
    return 1;
  int a = f (n-1);
  int b = f (n-2);
  return a + b;
  }
```

```
int fibonacci(int n) {
  if (n <= 2) {
    return 1;
  }
  return fibonacci(n - 1)
    + fibonacci(n - 2);
}
```

My Favorite Style Checkers

- Qulice for Java (Checkstyle + PMD)
- ESLint for JavaScript
- Clang-Tidy for C++
- Pylint for Python
- Rubocop for Ruby
- PHP_CodeSniffer for PHP
- rustfmt for Rust

References

Brian W. Kernighan and Phillip James Plauger. *The Elements of Programming Style*. McGraw-Hill, Inc., 1974.