“Simplicity is the prerequisite for reliability”

- Edsger W. Dijkstra

Edsger W. Dijkstra and Structured Programming

A mini-lecture series

CSE498 Collaborative Design (W) - Secure and Efficient C++ Software Development

03/19/2025

Kira Chan

https://cse.msu.edu/~chanken1/

Edsger Dijkstra

EWD1036-0

On the cruelty of really teaching computing science

The second part of this talk pursues some of the scientific and educational consequences of the assumption that computers represent a radical novelty. In order to give this assumption clear contents, we have to be much more precise as to what we mean in this context by the adjective "radical". We shall do so in the first part of this talk, in which we shall furthermore supply evidence in support of our assumption.

The usual way in which we plan today for tomorrow is in yesterday's vocabulary. We do so, be-

Edsger Dijkstra

- Dijkstra’s shortest path algorithm
- Semaphore
- Reverse Polish Notation to reduce computer memory access when evaluating expressions
- Banker’s algorithm
- System self-stabilization
- Separations of concerns
- …

Biggest Contribution: Structured Programming

Spaghetti code

TI-84

- Graphing calculator
- Supports programming using a BASIC-like language
- Sequential programming instructions
- Rather than functions, you would jump to a different segment of code, using:
  - Goto statements
  - Labels

Quadratic equation program

- Input -> A, B, C
- Discriminant_sqrt = 0
- Goto **dis_label**
- Label **combine_label**
- X_pos = (-b + discriminant) / 2a
- X_neg = (-b – discriminant) / 2a
- Print(Solutions: X_pos, X_neg)

- Label **dis_label:**
- Discriminant_sqrt = sqrt(b^2 – 4ac)
- Goto **combine_label**

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Scalability

- Now, imagine hundred thousands of lines in the same file, with many goto, labels, and jump commands.

Spaghetti code

Issues

A Case Study of Toyota Unintended Acceleration and Software Safety

Prof. Phil Koopman

Why does it matter?

- If all languages are turing complete, then why does it matter?

Structured Programming

- Promotes the use of structured control flows
  - As opposed to unstructured jumps to various sections of the program using Goto statements
- Elimination of global variables
- Loops and functions
- Hierarchical decomposing of the program

Structured vs Unstructured Programming

- Unstructured programming can also achieve the same result as structured programming
- But are much more prone to errors, development issues, lack of clarity, etc.

- It also does not mean you cannot write (basically) unstructured code in a structured programming language

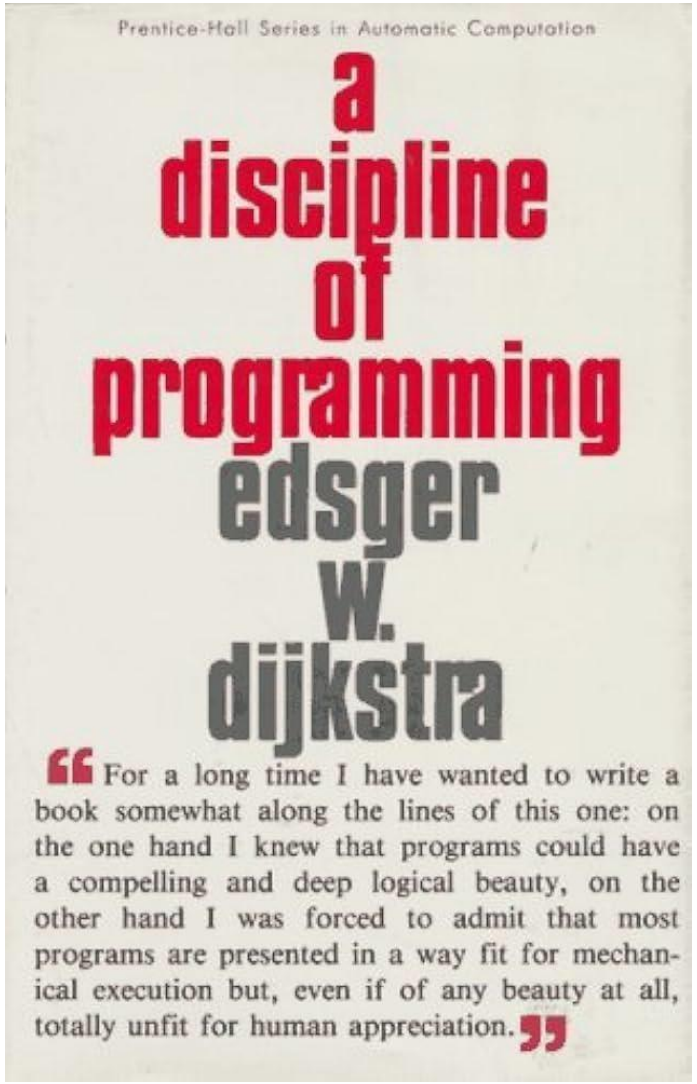
Examples

- Procedural Programming
  - E.g., functional programming
- Object-oriented Programming
- Model-based Programming
  - E.g., databases or query languages

Advantages

- Improved readability of code, and it forces the developer and reader to follow the logical structure of the code
- More efficient
- Easier to understand and modify
- Reusability

A discipline of programming



Person of the day
Edsger Dijkstra

- Famous for Dijkstra shortest path
- Structured Programming
  - No goto statements
  - Modern programming practices
- Semaphores
- Never used a computer (What??)
- Favourite composer is Mozart
- “Simplicity is a great virtue but it requires hard work to achieve it and education to appreciate it. And to make matters worse: complexity sells better.”