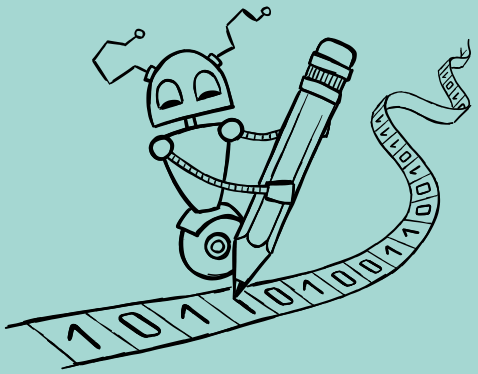

2^5 Things I Learned in Computer Science

University of Basel, Department of Mathematics and Computer Science

2^5 Things I Learned in Computer Science

Department of Mathematics and Computer Science

University of Basel

Author’s Note

Studying Computer Science is like taking a deep dive into an endless ocean of ideas, concepts, and tools. It is easy to get lost in this wide landscape, even more so if it is not one’s main field of expertise.

This is why Computer Science students of the University of Basel set out to identify the essential concepts and ideas they discovered during their studies. The intention was to go beyond the scope of traditional textbooks and also include the pecularities that are not formally part of typical Computer Science curricula. Finally, the goal was to present the fruit of this work to people outside of our field. The project was organized in the form of a Master’s level seminar, which had its first iteration during the spring semester 2019.

The result is **2^5 Things I Learned in Computer Science**, a collection of short illustrated articles. It is available in the form of this booklet, and online on the companion website https://tilics.dmi.unibas.ch

We hope that you enjoy these info bytes,
and that you pass them on to your friends!

Seminar participants (fall semester 2024)

Fabricio Arend Torres
Nils Bühlmann
Rebecca Dold
Simon Dold
Omnia Kahla
Patrick Kahr
Marcel Lüthi
Sebastian Philipps
Linard Schwendener
Tim Steindl
Christian Tschudin
Marco Vogt
Moira Zuber

List of Topics

0 Hello World
1 Handshake
2 Automation
3 Binary system
4 Run time complexity
5 Finite state machine
6 Turing machine
7 Polygon mesh
8 XOR
9 Monte Carlo method
10 Turing test
11 ASCII
12 System vs service uptime
13 Overflow
14 Camel case
15 Zero vs Null
16 Programming bugs
17 Patching
18 Octal system
19 $25_{dec} = 31_{oct}$
20 Version control systems
21 Interface in software engineering
22 Deadlock
23 Numeric precision
24 Mutex
25 Quine
26 Greedy algorithms
27 Speculative execution
28 Reverse engineering
29 Brute force attack
30 Year 2038 problem
31 Forking open source projects