
Preface

To measure is to know.

Lord Kelvin

The number of programmable devices in the world is currently in the order of billions and grows at an immense pace.¹ They operate on various hardware platforms, different operating systems, and feature an untold number of software modules written in hundreds of programming languages.

This ubiquity and versatility of programmable devices creates growing demand for software that needs to be commissioned, designed, written, tested, integrated, deployed, and maintained. No wonder software engineering nowadays belongs to the most demanded professional skills, also in the domains historically not associated with computer science (CS), like biology, medicine, or psychology. CS is an integral part of curriculum not only in universities, but also in high and elementary schools.

Despite the manyfold increase of the number of CS graduates in recent decades, the growing supply of CS professionals still does not meet the demand, even though software development is much more efficient nowadays and routinely supported by tools that the IT specialists in the past could have only dreamed of. Integrated development environments, debuggers, profilers, testing frameworks, and other computer-aided software engineering tools are indispensable in today's software engineer's toolbox. Yet, despite all these aids, software development is still a challenging and resource-demanding process, in part because contemporary software artifacts can be orders of magnitude more complex than they used to be in the past.

¹ There were 1.2 billion computers worldwide in 2011 and 6.7 billion mobile cellular subscriptions in 2013 (wolframalpha.com), and the number of PCs in use is likely to pass the 2 billion mark in 2015 (worldometers.info).

Behavioral Program Synthesis with Genetic
Programming

Krawiec, K.

2016, XXI, 172 p. 25 illus., 15 illus. in color., Hardcover

ISBN: 978-3-319-27563-5