1. Record Nr.   UNINA9910296448403321

   Autore   Barr Adam

   Titolo   The problem with software : why smart engineers write bad code / / by Adam Barr

   Pubbl/distr/stampa   Cambridge : , : The MIT Press, , 2018

   ISBN   0-262-34820-9

   Descrizione fisica   1 PDF (320 pages)

   Collana   The MIT Press

   Disciplina   005.3

   Soggetti   Computer software - Development
                                 Computer programmers
                                 Anecdotes.

   Lingua di pubblicazione   Inglese

   Formato   Materiale a stampa

   Livello bibliografico   Monografia

   Nota di bibliografia   Includes bibliographical references and index.

   Sommario/riassunto   An industry insider explains why there is so much bad software - and why academia doesn't teach programmers what industry wants them to know. Why is software so prone to bugs? So vulnerable to viruses? Why are software products so often delayed, or even canceled? Is software development really hard, or are software developers just not that good at it? In The Problem with Software , Adam Barr examines the proliferation of bad software, explains what causes it, and offers some suggestions on how to improve the situation. For one thing, Barr points out, academia doesn't teach programmers what they actually need to know to do their jobs: how to work in a team to create code that works reliably and can be maintained by somebody other than the original authors. As the size and complexity of commercial software have grown, the gap between academic computer science and industry has widened. It's an open secret that there is little engineering in software engineering, which continues to rely not on codified scientific knowledge but on intuition and experience. Barr, who worked as a programmer for more than twenty years, describes how the industry has evolved, from the era of mainframes and Fortran to today's embrace of the cloud. He explains bugs and why software has so many of them, and why today's interconnected computers offer fertile ground

for viruses and worms. The difference between good and bad software can be a single line of code, and Barr includes code to illustrate the consequences of seemingly inconsequential choices by programmers. Looking to the future, Barr writes that the best prospect for improving software engineering is the move to the cloud. When software is a service and not a product, companies will have more incentive to make it good rather than "good enough to ship."