

| | |
|-------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1. Record Nr. | UNINA9910869166003321 |
| Autore | Dooley John F. |
| Titolo | Software Development, Design, and Coding : With Patterns, Debugging, Unit Testing, and Refactoring // by John F. Dooley, Vera A. Kazakova |
| Pubbl/distr/stampa | Berkeley, CA : , : Apress : , : Imprint : Apress, , 2024 |
| ISBN | 979-88-6880-285-0 |
| Edizione | [3rd ed. 2024.] |
| Descrizione fisica | 1 online resource (521 pages) |
| Disciplina | 001.642 |
| Soggetti | Software engineering Java (Computer program language) Compilers (Computer programs) Software Engineering Java Compilers and Interpreters |
| Lingua di pubblicazione | Inglese |
| Formato | Materiale a stampa |
| Livello bibliografico | Monografia |
| Nota di bibliografia | Includes bibliographical references and index. |
| Nota di contenuto | Chapter 1: Introduction to Software Development -- PART ONE: MODELS AND TEAM PRACTICES -- Chapter 2: Software Process Models -- Chapter 3: Project Management Essentials -- Chapter 4: Ethics and Professional Practice -- Chapter 5: Intellectual Property, Obligations, and Ownership -- Chapter 6: Requirements -- PART TWO: DESIGN PRACTICES -- Chapter 7: Software Architecture -- Chapter 8: Design Principles -- Chapter 9: Structured Design -- Chapter 10: Object-Oriented Overview -- Chapter 11: Object-Oriented Analysis and Design -- Chapter 12: Object-Oriented Design Principles -- Chapter 13: Design Patterns -- Chapter 14: Parallel Programming -- Chapter 15: Parallel; Design Patterns -- PART THREE: CODING PRACTICES -- Chapter 16: Code Construction -- Chapter 17: Debugging -- Chapter 18: Unit Testing -- Chapter 19: Code Reviews and Inspections -- Chapter 20: Wrapping It All Up. |
| Sommario/riassunto | Learn the principles of good software design and then turn those principles into great code. This book introduces you to software engineering, from the application of engineering principles to the development of software. You'll see how to run a software development |

project, examine the different phases of a project, and learn how to design and implement programs that solve specific problems. This book is also about code construction — how to write great programs and make them work. Whether you're new to programming or have written hundreds of applications, in this book you'll re-examine what you already do, and you'll investigate ways to improve. Using examples in the Java and C programming languages, you'll look deeply into coding standards, debugging, unit testing, modularity, and other characteristics of effective programs. This new third edition incorporates new content, new figures, clarifying revisions, and content reorganization across all chapters. The Software Development Approaches chapter has been updated to highlight the differences between lean and agile general approaches, their various specific implementations, and how they can be effectively combined in software development practices. The Project Management Essentials chapter has been expanded to incorporate "SoftAware Development": a new paradigm which centers individuals, interpersonal relationships, and workplace culture as the heart of healthy and sustainable joint creation of software. Finally, a brand new chapter on intellectual property discusses copyright, patents, and how joint work affects code ownership rights. You Will Learn Modern agile methodologies How to work on and with development teams How to leverage the capabilities of modern computer systems with parallel programming How to work with design patterns to exploit application development best practices How to use modern tools for development, collaboration, and source code controls.

| | |
|-------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 2. Record Nr. | UNINA9910971119503321 |
| Autore | Morris Peter |
| Titolo | Introduction to Game Theory // by Peter Morris |
| Pubbl/distr/stampa | New York, NY : , : Springer New York : , : Imprint : Springer, , 1994 |
| ISBN | 1-4612-4316-5 |
| Edizione | [1st ed. 1994.] |
| Descrizione fisica | 1 online resource (XXVI, 252 p.) |
| Collana | Universitext, , 2191-6675 |
| Disciplina | 519.3 |
| Soggetti | Discrete mathematics Discrete Mathematics |
| Lingua di pubblicazione | Inglese |
| Formato | Materiale a stampa |
| Livello bibliografico | Monografia |
| Note generali | Bibliographic Level Mode of Issuance: Monograph |
| Nota di bibliografia | Includes bibliographical references and index. |
| Nota di contenuto | 1. Games in Extensive Form -- 1.1. Trees -- 1.2. Game Trees -- 1.3. Choice Functions and Strategies -- 1.4. Games with Chance Moves -- 1.5. Equilibrium N-tuples of Strategies -- 1.6. Normal Forms -- 2. Two-Person Zero-Sum Games -- 2.1. Saddle Points -- 2.2. Mixed Strategies -- 2.3. Small Games -- 2.4. Symmetric Games -- 3. Linear Programming -- 3.1. Primal and Dual Problems -- 3.2. Basic Forms and Pivots -- 3.3. The Simplex Algorithm -- 3.4. Avoiding Cycles and Achieving Feasibility -- 3.5. Duality -- 4. Solving Matrix Games -- 4.1. The Minimax Theorem -- 4.2. Some Examples -- 5. Non-Zero-Sum Games -- 5.1. Noncooperative Games -- 5.2. Solution Concepts for Noncooperative Games -- 5.3. Cooperative Games -- 6. N-Person Cooperative Games -- 6.1. Coalitions -- 6.2. Imputations -- 6.3. Strategic Equivalence -- 6.4. Two Solution Concepts -- 7. Game-Playing Programs -- 7.1. Three Algorithms -- 7.2. Evaluation Functions -- Appendix. Solutions. |
| Sommario/riassunto | The mathematical theory of games has as its purpose the analysis of a wide range of competitive situations. These include most of the recreations which people usually call "games" such as chess, poker, bridge, backgammon, baseball, and so forth, but also contests between companies, military forces, and nations. For the purposes of developing the theory, all these competitive situations are called games. The analysis of games has two goals. First, there is the descriptive goal of understanding why the parties ("players") in |

competitive situations behave as they do. The second is the more practical goal of being able to advise the players of the game as to the best way to play. The first goal is especially relevant when the game is on a large scale, has many players, and has complicated rules. The economy and international politics are good examples. In the ideal, the pursuit of the second goal would allow us to describe to each player a strategy which guarantees that he or she does as well as possible. As we shall see, this goal is too ambitious. In many games, the phrase "as well as possible" is hard to define. In other games, it can be defined and there is a clear-cut "solution" (that is, best way of playing).

| | |
|-------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 3. Record Nr. | UNINA9910983493403321 |
| Autore | Verma Ashok Kumar |
| Titolo | Process Design for Chemical and Environmental Engineering / / by Ashok Kumar Verma |
| Pubbl/distr/stampa | Cham : , : Springer Nature Switzerland : , : Imprint : Springer, , 2025 |
| ISBN | 9783031648618 3031648617 |
| Edizione | [1st ed. 2025.] |
| Descrizione fisica | 1 online resource (606 pages) |
| Disciplina | 660 |
| Soggetti | Chemical engineering Chemical processes Production engineering Chemical Process Engineering Process Chemistry Thermal Process Engineering |
| Lingua di pubblicazione | Inglese |
| Formato | Materiale a stampa |
| Livello bibliografico | Monografia |
| Nota di contenuto | Introduction to Process Design -- Process Design Fundamentals -- Design of Heat Exchangers -- Mass Transfer Equipment -- Solid-Liquid and Liquid-Liquid Extraction -- Humidification, Dehumidification and Drying -- Reactor Design. |
| Sommario/riassunto | This book discusses the design methodology for chemical process |

equipment carrying out heat and mass transfer operations and various types of reactors. Process design is an important step before achieving a mechanical design of chemical process equipment. It requires comprehensive knowledge of thermodynamics, fluid flow, heat, and mass transfer operations, and chemical reaction engineering, which is covered by the various chapters in this book. It covers process design of (1) heat exchangers, condensers, and reboilers; (2) packed and stage columns for distillation and gas absorption in chapter; (3) liquid–liquid extractor and solid–liquid leaching systems; (4) cooling towers; and (5) four different types of catalytic reactors, packed bed, fluidized bed, slurry bubble column, and mechanically agitated slurry reactor. The book emphasizes using correlations and equations in place of design data available in graphical or tabular forms to make it suitable for solving problems using spreadsheets and other software. It includes new correlations if not available in the literature and references to data available on web resources. The book covers all major topics for the course Chemical Process Engineering for undergraduate students and is also helpful in carrying out process design calculations for undergraduate design projects.
