Prolog Programming For Artificial Intelligence 3rd Edition

Prolog ProgrammingProlog Programming for Artificial IntelligenceProlog Programming for Artificial IntelligenceKnowledge RepresentationTechniques of Prolog Programming with Implementation of Logical Negation and Quantified GoalsArtificial Intelligence Through PrologAdventure in PrologProlog and Expert SystemsThe Craft of PrologProlog Programming for StudentsArtificial Intelligence Techniques in PrologAgent-Oriented ProgrammingArtificial Intelligence Techniques in PrologThe Implementation of PrologLanguage Processing with Perl and PrologAn Introduction to Programming in PrologArtificial Intelligence in Chemical EngineeringLogic Programming, Knowledge Representation, and Nonmonotonic ReasoningA Guide to Artificial Intelligence with Visual PrologSimply LogicalBeginning Artificial Intelligence with the Raspberry PiLogic, Programming and PrologTurbo Prolog Owner's HandbookIntroduction to Logic ProgrammingProlog by ExampleProgramming in PrologClause and EffectAI Algorithms, Data Structures, and Idioms in Prolog, Lisp, and JavaTurbo PrologLogic ProgrammingCatalogue of Artificial Intelligence ToolsLogic Programming with PrologProlog Programming for Artificial IntelligenceThinking As ComputationThe Art of PrologArtificial Intelligence and the Design of Expert SystemsThe Question of Artificial IntelligenceParadigms of Artificial Intelligence ProgrammingLogical and Relational LearningProlog Programming in Depth

Prolog Programming

Provides a thorough discussion of AI's theoretical foundations and advanced applications, including expert system design and knowledge-based programming. It is a wealth of advanced AI topics and applications that should appeal to a broad audience.

Prolog Programming for Artificial Intelligence

Representation and reasoning; Logic programs; Programming style; Data structures; Program verification; Formal program synthesis; Implementation; Broader contributions to computing.

Prolog Programming for Artificial Intelligence

Gain a gentle introduction to the world of Artificial Intelligence (AI) using the Raspberry Pi as the computing platform. Most of the major AI topics will be explored, including expert systems, machine learning both shallow and deep, fuzzy logic control, and more! AI in action will be demonstrated using the Python language on the Raspberry Pi. The Prolog language will also be introduced and used to demonstrate fundamental AI concepts. In addition, the Wolfram language will be used as part of the deep machine learning demonstrations. A series of projects will walk you through how to implement AI concepts with the Raspberry Pi. Minimal expense is needed for the projects as only a few sensors and actuators will be required. Beginners and hobbyists can jump right in to creating AI projects with the Raspberry PI using this book. What You'll Learn What AI is and—as importantly—what it is not Inference and expert systems Machine learning both shallow and deep Fuzzy logic and how to apply to an actual control system When AI might be appropriate to include in a system Constraints and limitations of the Raspberry Pi AI implementation Who This Book Is For Hobbyists, makers, engineers involved in designing autonomous systems and wanting to gain an education in fundamental AI concepts, and non-technical readers who want to understand what AI is and how it might affect their lives.

Knowledge Representation

Approaches the subject by applying the format used in successful language courses. Offers a comprehensive exhibition of Prolog programming techniques in four stages--declarative, procedural, advanced and meta-programming. Presents simple and efficient implementation of logical negation and quantified goals which are necessary in expert systems. The dynamics of these new features are shown in the construction of a multilingual expert system shell that supports negative and quantified queries as well as subtypes. The easy-to-follow tutorial style and numerous fully-solved exercises facilitate understanding. Comes with 3.5 inch disk containing all programs in the book.

Techniques of Prolog Programming with Implementation of Logical Negation and Quantified Goals

Paradigms of AI Programming is the first text to teach advanced Common Lisp techniques in the context of building major AI systems. By reconstructing authentic, complex AI programs using state-of-the-art Common Lisp, the book teaches students and professionals how to build and debug robust practical programs, while demonstrating superior programming style and important AI concepts. The author strongly emphasizes the practical performance issues involved in writing real working programs of significant size. Chapters on troubleshooting and efficiency are included, along with a discussion of the fundamentals of object-oriented programming and a description of the main CLOS functions. This volume is an excellent text for a course on AI programming, a useful supplement for general AI courses and an indispensable reference for the professional programmer.

Artificial Intelligence Through Prolog

Artificial Intelligence Techniques in Prolog introduces the reader to the use of well-established algorithmic techniques in the field of artificial intelligence (AI), with Prolog as the implementation language. The techniques considered cover general areas such as search, rule-based systems, and truth maintenance, as well as constraint satisfaction and uncertainty management. Specific application domains such as temporal reasoning, machine learning, and natural language are also discussed. Comprised of 10 chapters, this book begins with an overview of Prolog, paying particular attention to Prolog terms and rules (and Prolog facts as special cases); unification; the and-or computation tree induced by a Prolog program and a query; the depth-first, left-to-right traversal of that tree by the standard Prolog interpreter; and built-in predicates such as unification and equality. Subsequent chapters deal with search and representation of graphs in Prolog; backward-chaining methods; truth maintenance systems; and constraint satisfaction. Reasoning with uncertainty, planning and temporal reasoning, and machine learning are also tackled. The book concludes with an assessment of natural language processing and some of the linguistic notions that are easily encoded in Prolog. This monograph will be of interest to both students and practitioners in the fields of AI and computer science.

Adventure in Prolog

Artificial intelligence (AI) is the part of computer science concerned with designing intelligent computer systems (systems that exhibit characteristics we associate with intelligence in human behavior). This book is the first published textbook of AI in chemical engineering, and provides broad and in-depth coverage of AI programming, AI principles, expert systems, and neural networks in chemical engineering. This book introduces the computational means and methodologies that are used to enable computers to perform intelligent engineering tasks. A key goal is to move beyond the principles of AI into its applications in chemical engineering. After reading this book, a chemical engineer will have a firm grounding in AI, know what chemical engineering applications of AI exist today, and understand the current challenges facing AI in engineering. Allows the reader to learn AI quickly using inexpensive personal computers Contains a large number of illustrative examples, simple exercises, and complex practice problems and solutions Includes a computer diskette for an illustrated case study Demonstrates an expert system for separation synthesis (EXSEP) Presents a detailed review of published literature on expert systems and neural networks in chemical engineering

Prolog and Expert Systems

This second edition contains revised chapters taking into account recent research advances. More advanced exercises have been included, and "Part II The Prolog Language" has been modified to be compatible with the new Prolog standard. This is a graduate level text that can be used for self-study.

The Craft of Prolog

This Festschrift volume, published in honor of Michael Gelfond on the occasion of his 65th birthday, contains a collection of papers written by his closest friends and colleagues. Several of these papers were presented during the Symposium on Constructive Mathematics in Computer Science, held in Lexington, KY, USA on October 25-26, 2010. The 27 scientific papers included in the book focus on answer set programming. The papers are organized in sections named “Foundations: ASP and Theories of LP, KR, and NMR”, “ASP and Dynamic Domains”, and “ASP – Applications and Tools”.

Prolog Programming for Students

The fourth edition of this best-selling guide to Prolog and Artificial Intelligence has been updated to include key developments in the field while retaining its lucid approach to these topics. New and extended topics include Constraint Logic Programming, abductive reasoning and partial order planning. Divided into two parts, the first part of the book introduces the programming language Prolog, while the second part teaches Artificial Intelligence using Prolog as a tool for the implementation of AI techniques. This textbook is meant to teach Prolog as a practical programming tool and so it concentrates on the art of using the basic mechanisms of Prolog to solve interesting problems. The fourth edition has been fully revised and extended to provide an even greater range of applications, making it a self-contained guide to Prolog, AI or AI Programming for students and professional programmers.

Artificial Intelligence Techniques in Prolog

Inludes glossary, index, Borland Software

Agent-Oriented Programming

Originally published in 1987 when Artificial Intelligence (AI) was one of the most hotly debated subjects of the moment; there was widespread feeling that it was a field whose ‘time had come’, that intelligent machines lay ‘just around the corner’. Moreover, with the onset of the revolution in information technology and the proclamation from all corners that we were moving into an ‘information society’, developments in AI and advanced computing were seen in many countries as having both strategic and economic importance. Yet, aside from the glare of publicity that tends to surround new scientific ideas or technologies, it must be remembered that AI was a relative newcomer among the sciences; that it had often been the subject of bitter controversy; and that though it had been promising to create intelligent machines for some 40 years prior to publication, many believe that it had actually displayed very little substantive progress. With this background in mind, the aim of this collection of essays was to take a novel look at AI. Rather than following the path of old well-trodden arguments about definitions of intelligence or the status of computer chess programs, the objective was to bring new perspectives to the subject in order to present it in a different light. Indeed, instead of simply adding to the endless wrangling ‘for’ and ‘against’ AI, the source of such divisions is made a topic for analysis in its own right. Drawing on ideas from the philosophy and sociology of scientific knowledge, this collection therefore broke new ground. Moreover, although a great deal had been written about the social and cultural impact of AI, little had been said of the culture of AI scientists themselves – including their discourse and style of thought, as well as the choices, judgements, negotiations and competitive struggles for resources that had shaped the genesis and development of the paradigmatic structure of their discipline at the time. Yet, sociologists of science have demonstrated that the analysis of factors such as these is a necessary part of understanding the development of scientific knowledge. Hence, it was hoped that this collection would help to redress the imbalance and provide a broader and more interesting picture of AI.

Artificial Intelligence Techniques in Prolog

The purpose of this catalogue is to promote interaction between members of the AI' community. It will do this by announcing the existence of Ai techniques and portable software. and acting as a pOinter into the literature. Thus the AI community wili have access to a common. extensional definition of the field. which will: promote a common terminology. discourage the reinvention of wheels. and act as a clearing house for ideas and software. The cataiogue is a reference work providing a quick guide to the AI tools available for different jobs. It is not intended to be a textbook like the Artificial Intelligence Handbook. It. intentionally. only provides a brief description of each tool. with no extended discussion of the historical origin of the tool or how it has been used in particular AI programs, The focus is on techniques abstracted from their historical origins. The original version of the catalogue. was hastily built in 1983 as part of the UK SERC-Dol. IKBS. Architecture Study [lKBS Architecture Study 831. it has now been adopted by the SERC Specially Promoted Programme in IKBS and is kept as an on line document undergoing constant revision and refinement and published as a paperback by Springer Verlag.

The Implementation of Prolog

A book that furnishes no quotations is, me judice, no book – it is a plaything. TL Peacock: Crochet Castle The paradigm presented in this book is proposed as an agent programming language. The book charts the evolution of the language from Prolog to intelligent agents. To a large extent, intelligent agents rose to prominence in the mid-1990s because of the World Wide Web and an ill-structured network of multimedia information. Age- oriented programming was a natural progression from object-oriented programming which C++ and more recently Java popularized. Another strand of influence came from a revival of interest in robotics [Brooks, 1991a; 1991b]. The quintessence of an agent is an intelligent, willing slave. Speculation in the area of artificial slaves is far more ancient than twentieth century science fiction. One documented example is found in Aristotle’s Politics written in the fourth century BC. Aristotle classifies the slave as “an animate article of property”. He suggests that slaves or subordinates might not be necessary if “each instrument could do its own work at command or by anticipation like the statues of Daedalus and the tripods of Hephaestus”. Reference to the legendary robots devised by these mythological technocrats, the former an artificer who made wings for Icarus and the latter a blacksmith god, testify that the concept of robot, if not the name, was ancient even in Aristotle’s time.

Language Processing with Perl and Prolog

This book is an introduction to Prolog (£rQgramming in ~ic). It presents the basic foundations of Prolog and basic and fundamental programming methods. This book is written for programmers familiar with other programming languages, as well as for novices in computer science, willing to have an original introduction to programming. The approach adopted in this book is thus based on methodological elements together with some pragmatic aspects. The book is composed of two parts. In the fIrst part the major aspects of programming in Prolog are presented step by step. Each new aspect is illustrated by short examples and exercises. The second part is composed of more developed examples, which are often games, that illustrate major aspects of artifIcial intelligence. More advanced books are given in the bibliography and will allow the reader to deepen his or her know ledge of Prolog. Prolog was first designed in France at OJ.A., Marseille, with a specific syntax. We have adopted here a more common notation, defIned at Edinburgh, which tends to be an implicit norm. At the end of each chapter of the fIrst part, there are exercises that the reader is invited to do and to test on his or her machine. Complete answers are given in Appendix A, at the end of the book.

An Introduction to Programming in Prolog

This new edition covers the latest advances in logic programming research. Its clear and authoritative treatment of the theory is combined with an emphasis on practical programming issues. This is a broad and highly readable introduction to the subject for advanced students and programmers.

Artificial Intelligence in Chemical Engineering

This book is for people who have done some programming, either in Prolog or in a language other than Prolog, and who can find their way around a reference manual. The emphasis of this book is on a simplified and disciplined methodology for discerning the mathematical structures related to a problem, and then turning these structures into Prolog programs. This book is therefore not concerned about the particular features of the language nor about Prolog programming skills or techniques in general. A relatively pure subset of Prolog is used, which includes the 'cut', but no input/output, no assert/retract, no syntactic extensions such as if then-else and grammar rules, and hardly any built-in predicates apart from arithmetic operations. I trust that practitioners of Prolog program ming who have a particular interest in the finer details of syntactic style and language features will understand my purposes in not discussing these matters. The presentation, which I believe is novel for a Prolog programming text, is in terms of an outline of basic concepts interleaved with worksheets. The idea is that worksheets are rather like musical exercises. Carefully graduated in scope, each worksheet introduces only a limited number of new ideas, and gives some guidance for practising them. The principles introduced in the worksheets are then applied to extended examples in the form of case studies.

Logic Programming, Knowledge Representation, and Nonmonotonic Reasoning

Students explore the idea that thinking is a form of computation by learning to write simple computer programs for tasks that require thought. This book guides students through an exploration of the idea that thinking might be understood as a form of computation. Students make the connection between thinking and computing by learning to write computer programs for a variety of tasks that require thought, including solving puzzles, understanding natural language, recognizing objects in visual scenes, planning courses of action, and playing strategic games. The material is presented with minimal technicalities and is accessible to undergraduate students with no specialized knowledge or technical background beyond high school mathematics. Students use Prolog (without having to learn algorithms: "Prolog without tears "), learning to express what they need as a Prolog program and letting Prolog search for answers. After an introduction to the basic concepts, Thinking as Computation offers three chapters on Prolog, covering back-chaining, programs and queries, and how to write the sorts of Prolog programs used in the book. The book follows this with case studies of tasks that appear to require thought, then looks beyond Prolog to consider learning, explaining, and propositional reasoning. Most of the chapters conclude with short bibliographic notes and exercises. The book is based on a popular course at the University of Toronto and can be used in a variety of classroom contexts, by students ranging from first-year liberal arts undergraduates to more technically advanced computer science students.

A Guide to Artificial Intelligence with Visual Prolog

The emphasis in The Craft of Prolog is on using Prolog effectively. It presents a loose collection of topics that build on and elaborate concepts learned in a first course.

Simply Logical

This bestselling guide to Prolog has been fully revised and extended to provide an even greater range of applications, enhancing its value as a stand-alone guide to Prolog, Artificial Intelligence or AI programming for professional programmers and students alike.

Beginning Artificial Intelligence with the Raspberry Pi

Logic Programming was effectively defined as a discipline in the early seventies. It is only during the early to mid eighties that books, conferences and journals devoted entirely to Logic Programming began to appear. Consequently, much of the work done during this first crucial decade in Marseilles, Edinburgh, London, Budapest and Stockholm (to name a few) is often overlooked or difficult to trace. There are now two main regular conferences on Logic Programming, and at least five journals: The Journal of Logic Programming, New Generation Computing, Automated Reasoning, The Journal of SJmbolic Computation, and Future Generation Computer Systems. Logic Programming, however, has its roots in Automated Theorem Proving and via the expanding area of expert systems, strongly influences researchers in such varied fields as Civil Engineering, Chemistry, Law, etc. Consequently, many papers related to Logic Programming appear in a wide variety of journals and proceedings of conferences in other disciplines. This is particularly true of Computer Science where a revolution is taking place in hardware design, programming languages, and more recently databases. One cannot overestimate the importance of such a bibliography.

Logic, Programming and Prolog

The book uses Edinburgh syntax.

Turbo Prolog Owner's Handbook

Written for those who wish to learn Prolog as a powerful software development tool, but do not necessarily have any background in logic or AI. Includes a full glossary of the technical terms and self-assessment exercises.

Introduction to Logic Programming

This first textbook on multi-relational data mining and inductive logic programming provides a complete overview of the field. It is self-contained and easily accessible for graduate students and practitioners of data mining and machine learning.

Prolog by Example

This book is intended as an introduction to the Prolog language, with an emphasis on one of the most recent versions - Turbo-Prolog.

Programming in Prolog

Clause and Effect

AI Algorithms, Data Structures, and Idioms in Prolog, Lisp, and Java

Turbo Prolog

This text covers natural language processing in Prolog and presumes knowledge of Prolog, but not of linguistics. It includes simple but practical database query systems; covers syntax, formal semantics, and morphology; emphasizes working computer programs that implement subsystems of a natural language processor; features programs that are clearly designed and compatible with any Edinburgh-compatible prolog implementation (Quintas, ESL, Arity, ALS etc.); and contains nearly 100 hands-on Prolog programming exercises and problem sets.

Logic Programming

The computer programming language Prolog is quickly gaining popularity throughout the world. Since Its beginnings around 1970. Prolog has been chosen by many programmers for applications of symbolic computation. including: D relational databases D mathematical logic D abstract problem solving D understanding natural language D architectural design D symbolic equation solving D biochemical structure analysis D many areas of artificial Intelligence Until now. there has been no textbook with the aim of teaching Prolog as a practical programming language. It Is perhaps a tribute to Prolog that so many people have been motivated to learn It by referring to the necessarily concise reference manuals. a few published papers. and by the orally transmitted 'folklore' of the modern computing community. However. as Prolog is beginning to be Introduced to large numbers of undergraduate and postgraduate students. many of our colleagues have expressed a great need for a tutorial guide to learning Prolog. We hope this little book will go some way towards meeting this need. Many newcomers to Prolog find that the task of writing a Prolog program Is not like specifying an algorithm in the same way as In a conventional programming language. Instead. the Prolog programmer asks more what formal relationships and objects occur In his problem.

Catalogue of Artificial Intelligence Tools

Prolog has a declarative style. A predicate definition includes both the input and output parameters, and it allows a programmer to define a desired result without being concerned about the detailed instructions of how it is to be computed. Such a declarative language offers a solution to the software crisis, because it is shorter and more concise, more powerful and understandable than present-day languages. Logic highlights novel aspects of programming, namely using the same program to compute a relation and its inverse, and supporting deductive retrieval of informa tion. This is a book about using Prolog. Its real point is the examples introduced from Chapter 3 onwards, and so a Prolog programmer does not need to read Chapters 1 and 2, which are oriented more to teachers and to students, respec tively. The book is recommended for introductory and advanced university courses, where students may need to remember the basics about logic program ming and Prolog, before starting doing. Chapters 1 and 2 were also kept for the sake of unity of the whole material. In Chapter 1 a teaching strategy is explained based on the key concepts of Pro log which are novel aspects of programming. Prolog is enhanced as a computer programming language used for solving problems that involve objects and the relationships between objects. This chapter provides a pedagogical tour of pre scriptions for the organization of Prolog programs, by pointing out the main draw backs novices may encounter.

Logic Programming with Prolog

The areas of natural language processing and computational linguistics have continued to grow in recent years, driven by the demand to automatically process text and spoken data. With the processing power and techniques now available, research is scaling up from lab prototypes to real-world, proven applications. This book teaches the principles of natural language processing, first covering practical linguistics issues such as encoding and annotation schemes, defining words, tokens and parts of speech and morphology, as well as key concepts in machine learning, such as entropy, regression and classification, which are used throughout the book. It then details the language-processing functions involved, including part-of-speech tagging using rules and stochastic techniques, using Prolog to write phase-structure grammars, syntactic formalisms and parsing techniques, semantics, predicate logic and lexical semantics and analysis of discourse and applications in dialogue systems. A key feature of the book is the author's hands-on approach throughout, with sample code in Prolog and Perl, extensive exercises, and a detailed introduction to Prolog. The reader is supported with a companion website that contains teaching slides, programs and additional material. The second edition is a complete revision of the techniques exposed in the book to reflect advances in the field the author redesigned or updated all the chapters, added two new ones and considerably expanded the sections on machine-learning techniques.

Prolog Programming for Artificial Intelligence

This book covers all that is needed by students on a one-year introductory Prolog course at first or second year degree level. It introduces Prolog to students as simply and painlessly as possible. Where Artificial Intelligence (AI) topics are introduced, they are easier ones and are treated simply. This book is Prolog for Students, with examples from AI, not a book on AI using Prolog. The text assumes access to a suitable, good, Prolog interpreter, such as LPA Prolog. It also assumes that students with an aptitude for research will follow it up with more advanced study, perhaps a third or fourth year option, and further reading suggestions are included. The book is organised with the basics of the subject introduced first, and covered gradually, so they can be fully understood before moving on to harder topics. The topics that students find more difficult, such as recursion and lists, are not covered until about half way through the book. There are many in-text questions, student self-testing questions and programming practice exercises throughout the book. If used to accompany a taught course, the material of one chapter can be covered in each week. This book covers all that is needed by students on a one-year introductory Prolog course at first or second year degree level. It introduces Prolog to students as simply and painlessly as possible. Where Artificial Intelligence (AI) topics are introduced, they are easier ones and are treated simply. This book is Prolog for Students, with examples from AI, not a book on AI using Prolog. The text assumes access to a suitable, good, Prolog interpreter, such as LPA Prolog. It also assumes that students with an aptitude for research will follow it up with more advanced study, perhaps a third or fourth year option, and further reading suggestions are included. The book is organised with the basics of the subject introduced first, and covered gradually, so they can be fully understood before moving on to harder topics. The topics that students find more difficult, such as recursion and lists, are not covered until about half way through the book. There are many in-text questions, student self-testing questions and programming practice exercises throughout the book. If used to accompany a taught course, the material of one chapter can be covered in each week.

Thinking As Computation

A semantically well-defined programming language widely used in artificial intelligence, Prolog has greatly influenced other programming languages since its introduction in the late 1970s. A user may find Prolog deceptively easy, however, and there are a number of different implementations. In this book Patrice Boizumault draws from his extensive experience in Prolog implementation to describe for students of all levels the concepts, difficulties, and design limits of a Prolog system. Boizumault introduces the specific problems posed by the implementation of Prolog, studies and compares different solutions--notably those of the schools of Marseilles and Edinburgh--and concludes with three examples of implementation. Major points of interest include identifying the important differences in implementing unification and resolution; presenting three features of Prolog II--infinite trees, dif, and freeze--that introduce constraints; thoroughly describing Warren's Abstract Machine (WAM); and detailing a Lisp imple-mentation of Prolog. Originally published in 1993. The Princeton Legacy Library uses the latest print-on-demand technology to again make available previously out-of-print books from the distinguished backlist of Princeton University Press. These editions preserve the original texts of these important books while presenting them in durable paperback and hardcover editions. The goal of the Princeton Legacy Library is to vastly increase access to the rich scholarly heritage found in the thousands of books published by Princeton University Press since its founding in 1905.

The Art of Prolog

Although many texts exist offering an introduction to artificial intelligence (AI), this book is unique in that it places an emphasis on knowledge representation (KR) concepts. It includes small-scale implementations in PROLOG to illustrate the major KR paradigms and their developments.****back cover copy:**Knowledge representation is at the heart of the artificial intelligence enterprise: anyone writing a program which seeks to work by encoding and manipulating knowledge needs to pay attention to the scheme whereby he will represent the knowledge, and to be aware of the consequences of the choices made.****The book's distinctive approach introduces the topic of AI through a study of knowledge representation issues. It assumes a basic knowledge of computing and a familiarity with the principles of elementary formal logic would be advantageous.****Knowledge Representation: An Approach to Artificial Intelligence develops from an introductory consideration of AI, knowledge representation and logic, through search technique to the three central knowledge paradigms: production rules, structured objects, and predicate calculus. The final section of the book illustrates the application of these knowledge representation paradigms through the Prolog Programming language and with an examination of diverse expert systems applications. The book concludes with a look at some advanced issues in knowledge representation.****This text provides an introduction to AI through a study of knowledge representation and each chapter contains exercises for students. Experienced computer scientists and students alike, seeking an introduction to AI and knowledge representations will find this an invaluable text.

Artificial Intelligence and the Design of Expert Systems

An introduction to Prolog programming for artificial intelligence covering both basic and advanced AI material. A unique advantage to this work is the combination of AI, Prolog and Logic. Each technique is accompanied by a program implementing it. Seeks to simplify the basic concepts of logic programming. Contains exercises and authentic examples to help facilitate the understanding of difficult concepts.

The Question of Artificial Intelligence

Not long ago" Dennis Merritt wrote one of the best books that I know of about implementing expert systems in Prolog, and I was very glad he published it in our series. The only problem is there are still some unfortunate people around who do not know Prolog and are not sufficiently prepared either to read Merritt's book, or to use this extremely productive language, be it for knowledge-based work or even for everyday programming. Possibly this last statement may surprise you if you were under the impression that Prolog was an "artificial intelligence language" with very limited application potential. Please believe this editor's statement that quite the opposite is true: for at least four years, I have been using Prolog for every programming task in which I am given the option of choosing the language. Therefore, I 'am indeed happy that Dennis Merritt has written another good book on my language of choice, and that it meets the high standard he set with his prior book, Building Expert Systems in Prolog. All that remains for me to do is to wish you success and enjoyment when taking off on your Adventure in Prolog.

Paradigms of Artificial Intelligence Programming

Artificial Intelligence Techniques in Prolog introduces the reader to the use of well-established algorithmic techniques in the field of artificial intelligence (AI), with Prolog as the implementation language. The techniques considered cover general areas such as search, rule-based systems, and truth maintenance, as well as constraint satisfaction and uncertainty management. Specific application domains such as temporal reasoning, machine learning, and natural language are also discussed. Comprised of 10 chapters, this book begins with an overview of Prolog, paying particular attention to Prolog terms and rules (and Prolog facts as special cases); unification; the and-or computation tree induced by a Prolog program and a query; the depth-first, left-to-right traversal of that tree by the standard Prolog interpreter; and built-in predicates such as unification and equality. Subsequent chapters deal with search and representation of graphs in Prolog; backward-chaining methods; truth maintenance systems; and constraint satisfaction. Reasoning with uncertainty, planning and temporal reasoning, and machine learning are also tackled. The book concludes with an assessment of natural language processing and some of the linguistic notions that are easily encoded in Prolog. This monograph will be of interest to both students and practitioners in the fields of AI and computer science.

Logical and Relational Learning

Designed for undergraduate courses on Expert Systems, PROLOG or introductory Artificial Intelligence, this informally-styled text assumes no background in PROLOG or Logic Programming, but combines an introduction to PROLOG with a mastery of its application to expert systems programming.

Prolog Programming in Depth

Get started with the simplest, most powerful prolog ever: Visual Prolog If you want to explore the potential of Artificial Intelligence (AI), you need to know your way around Prolog. Prolog - which stands for "programming with logic" - is one of the most effective languages for building AI applications, thanks to its unique approach. Rather than writing a program that spells out exactly how to solve a problem, with Prolog you define a problem with logical Rules, and then set the computer loose on it. This paradigm shift from Procedural to Declarative programming makes Prolog ideal for applications involving AI, logic, language parsing, computational linguistics, and theorem-proving. Now, Visual Prolog (available as a free download) offers even more with its powerful Graphical User

Interface (GUI), built-in Predicates, and rather large provided Program Foundation Class (PFC) libraries. A Guide to Artificial Intelligence with Visual Prolog is an excellent introduction to both Prolog and Visual Prolog. Designed for newcomers to Prolog with some conventional programming background (such as BASIC, C, C++, Pascal, etc.), Randall Scott proceeds along a logical, easy-to-grasp path as he explains the beginnings of Prolog, classic algorithms to get you started, and many of the unique features of Visual Prolog. Readers will also gain key insights into application development, application design, interface construction, troubleshooting, and more. In addition, there are numerous sample examples to learn from, copious illustrations and information on helpful resources. A Guide to Artificial Intelligence with Visual Prolog is less like a traditional textbook and more like a workshop where you can learn at your own pace - so you can start harnessing the power of Visual Prolog for whatever your mind can dream up.