FRESNO STATE | Discovery. Diversity. Distinction.

Department of Computer Science

COLLEGE OF SCIENCE AND MATHEMATICS

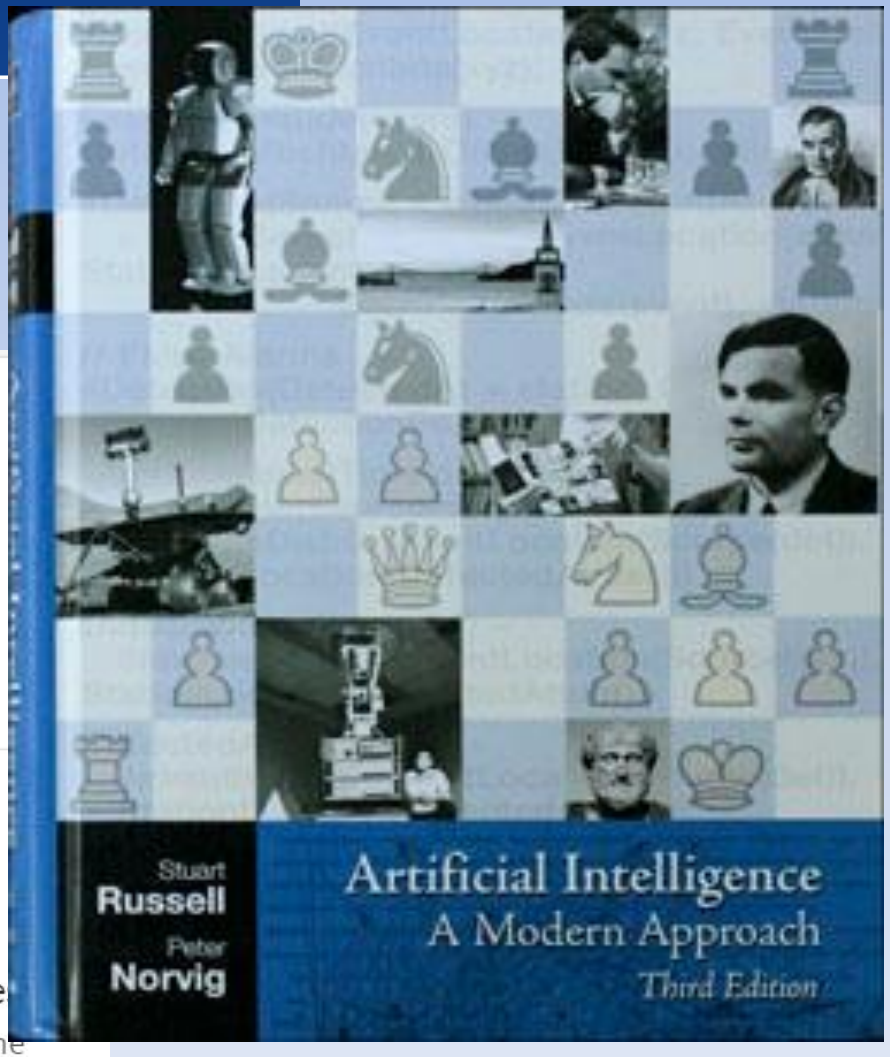
WELCOME Spring '18

```
38 int Move(string& board, int dir){
39   int row, col;
40   LocBlank(board, row, col);
41   if ((dir==1 || dir == -3) && row > 0){
42     Swap(board, row, col, row-1, col);
43   }
44   else if ((dir == 2 || dir == -4) && col < MAXDIM-1){
45     Swap(board, row, col, row, col+1);
46   }
47   else if ((dir == 3 || dir == -1) && row < MAXDIM-1){
48     Swap(board, row, col, row+1, col);
49   }
50   else if ((dir == 4 || dir == -2) && col > 0 ){
51     Swap(board, row, col, row, col-1);
52   }
53   else {
54     return -1;
55   }
56   return 0;
```

david ruby

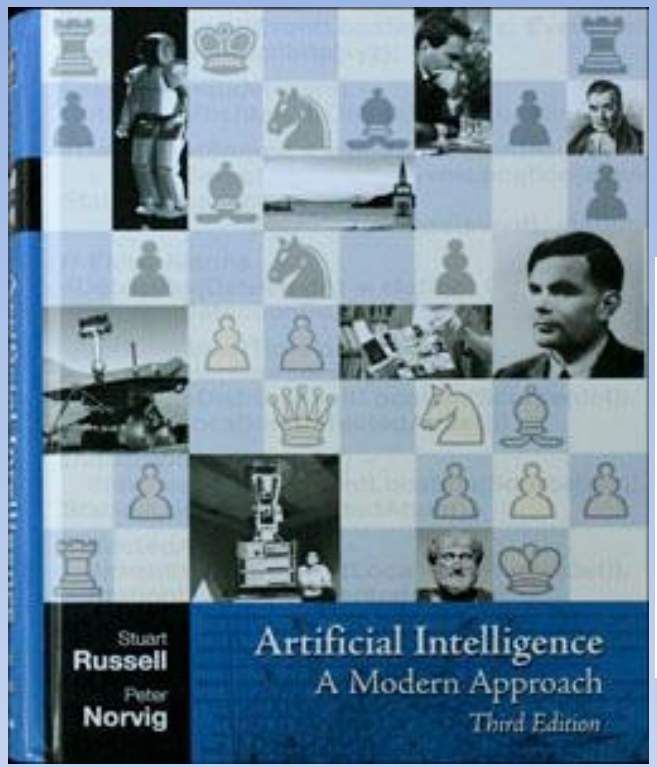
Lecturer at California State University, Fre

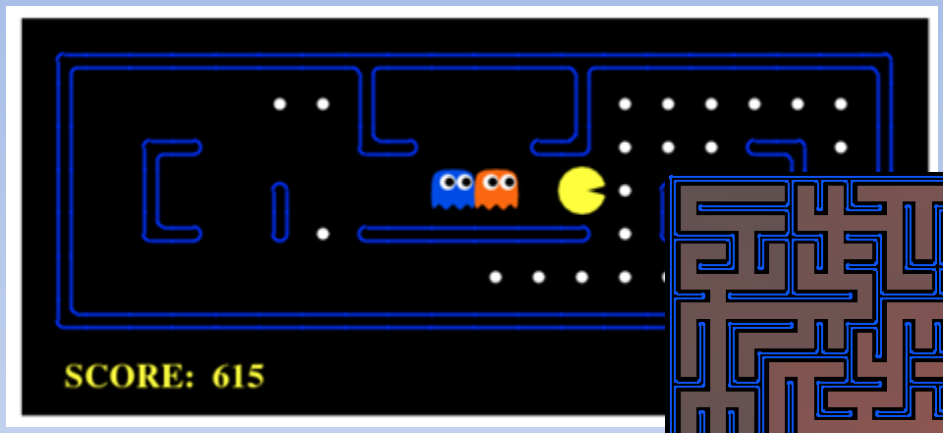
California State University, Fresno • UC Irvine

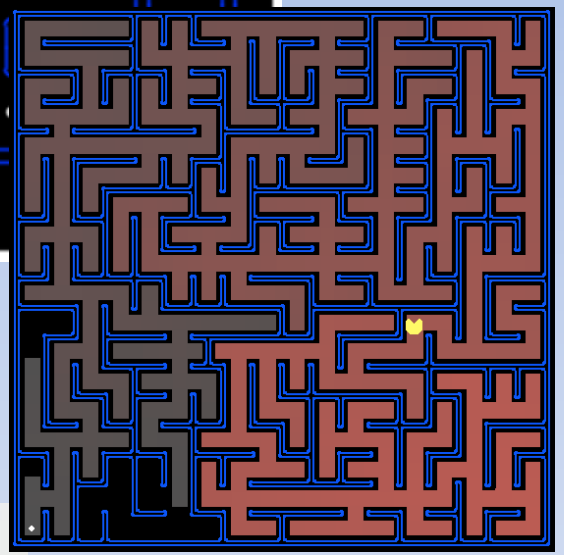


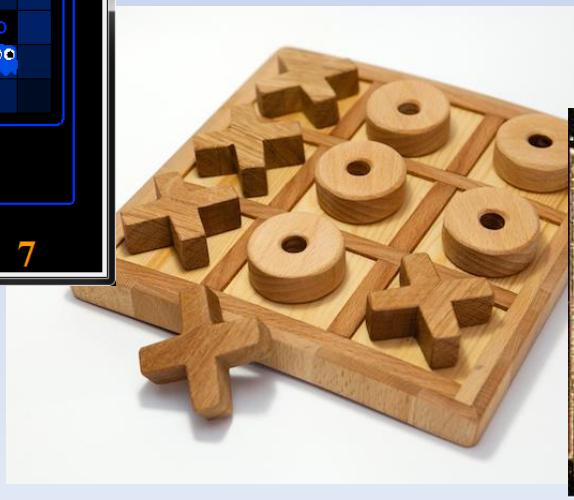
CSCI 164

AI Programming



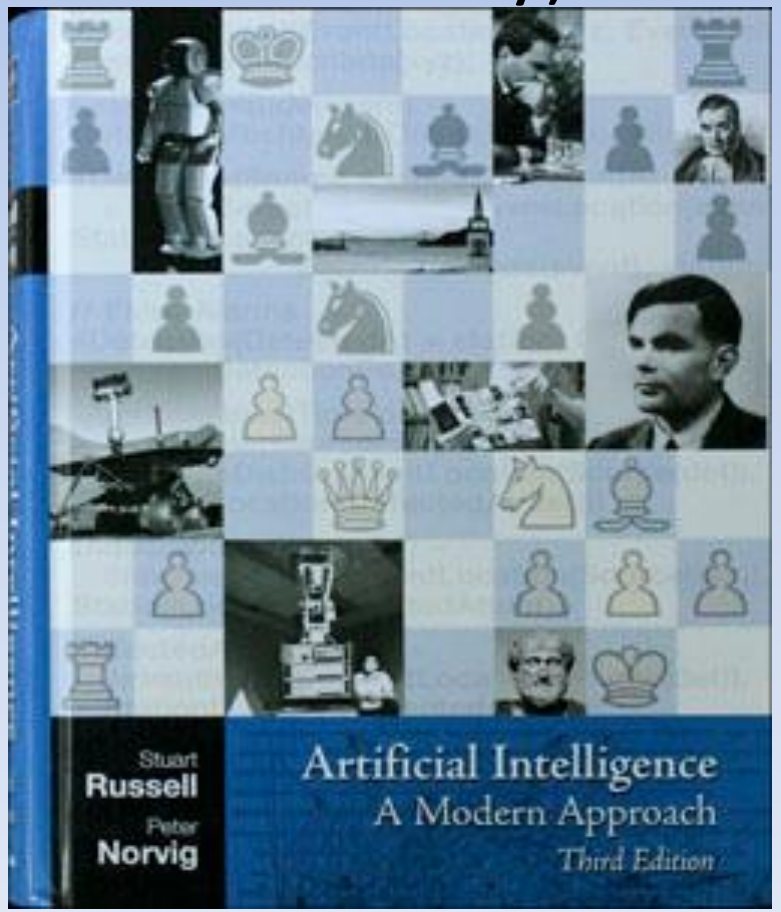






Textbook

- The 22nd most cited computer science publication on Citeseer (and 4th most cited publication of this century).



Dr. Stuart Russell

Professor of Computer Science and Michael H. Smith and Lotfi A. Zadeh Chair in Engineering,
Computer Science Division
University of California
Berkeley, CA 94720

Adjunct Professor of Neurological Surgery, University of California, San Francisco

Tel. (510) 642-4964
Fax (510) 642-5775
Email russell@cs.berkeley.edu
Home page http://www.cs.berkeley.edu/~russell

Education

B.A. (Hons.) 1st Class, Physics, Wadham College, University of Oxford, 1979--82.
Ph.D., Computer Science, Stanford University, 1982--86.

Employment history

2012--present, Professeur Invité, Université Pierre et Marie Curie, Paris
2012--present, Professeur, Fondation de l'École Normale Supérieure, Paris
2008--present, Adjunct Professor, Department of Neurological Surgery, University of California, San Francisco
2008--2010, Chair, Department of Electrical Engineering and Computer Sciences, University of California, Berkeley
2006--2010, Chair, Computer Science Division, University of California, Berkeley
1996--present, Professor, Computer Science Division, University of California, Berkeley
1991--96, Associate Professor, Computer Science Division, University of California, Berkeley
1986--91, Assistant Professor, Computer Science Division, University of California, Berkeley
1986, Summer employee, MCC, Austin, Texas, Machine learning research in the Large Scale KB Project (CYC)
1985--86, Research Assistant, Computer Science Dept., Stanford University
1983, Teaching Assistant, Computer Science Dept., Stanford University
1981, Programmer, graphics research project, IBM Los Angeles Scientific Center
1978--80 (1 year total), Programmer, IBM Systems Engineering Centre, Warwick, UK

Dr. Peter Norvig

Professional Employment (Full-Time)

2001-now	Google	Director of Research (2006-now); formerly Director of Search Quality (2002-2006) and Machine Learning (2001).
1998-2001	NASA Ames Research Center	Division Chief, Computational Sciences
1996-1998	Junglee Corp.	Chief Scientist
1994-1996	Harlequin, Inc.	Chief Designer
1991-1994	Sun Microsystems Labs	Senior Scientist
1986-1991	University of California, Berkeley	Research Faculty Member
1985-1986	University of Southern California	Assistant Professor
1978-1980	Higher Order Software, Inc.	Member of Technical Staff
1977-1977	Woods Hole Oceanographic Institute	Summer Programming Intern

I also have served as an advisory board member for various companies, including: Root-1, Fetch, CleverSet, Ask Jeeves, Thinking Software, PersonalGenie.com.

Education

1980-1985	Ph.D. Computer Science	University of California, Berkeley
1974-1978	B.S. Applied Mathematics	Brown University

Personal Information

Citizen: U.S.
Raised: RI, MA, CA.
Status: Married with 2 children.
Erdos #: 3 (Erdos to Peter Cameron to Stuart Russell to me)

Personal Computer Science Focus

CSCI 164. Artificial Intelligence Programming

Prerequisite: CSCI 117. Introduction to problem-solving methods from artificial intelligence. Production systems. Knowledge-based systems. Machine learning. Topics chosen from fuzzy logic, neural network models, genetic algorithms. Verification, validation, testing.

Units: 3

CSCI 166. Principles of Artificial Intelligence

Prerequisite: CSCI 164. Analysis of knowledge-based and neural models, including self-organization, sequential learning models, neurally inspired models of reasoning and perception. Integration of different paradigms.

Units: 3

CSCI 126. Database Systems

Prerequisites: CSCI 124. Database concepts; hierarchical and relational network models; object-oriented data models. Data normalization, data description languages, data manipulation languages, and query design.

Units: 3
Course Typically Offered: Spring

CSCI 174. Design and Analysis of Algorithms

Prerequisites: CSCI 115, CSCI 119. Models of computation and measures of complexity, algorithms for sorting and searching, set representation and manipulation, branch and bound, integer and polynomial arithmetic, pattern-matching algorithms, parsing algorithm, graph algorithm, NP-complete problems.

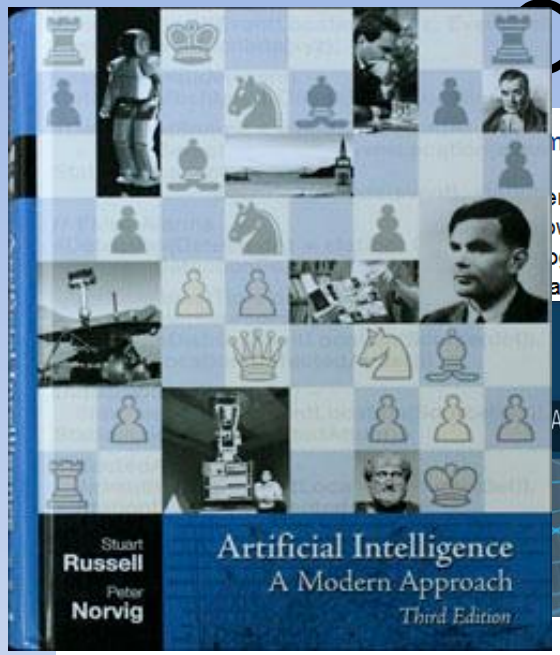
Units: 3

Personal Computer Science Focus

- Search
  - AI, Algorithms, RDMS
- Data
  - Machine Learning, Algorithms, RDMS
- Puzzles
  - AI, Algorithms

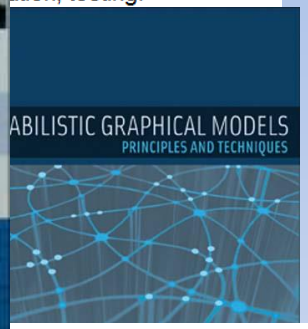
- **Books…**

Computer Science



ning

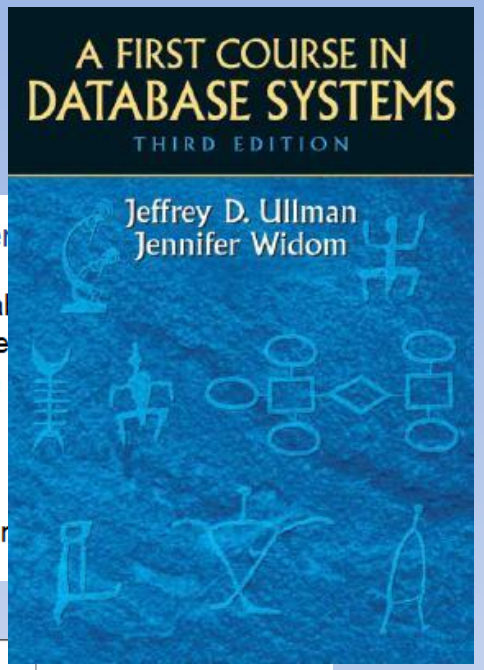
em-solving methods from
owledge-based systems.
ogic, neural network
ation, testing.



CSCI 126. Database Syste

Prerequisites: CSCI 124. Datal
relational network models; obje
normalization, data description
languages, and query design.

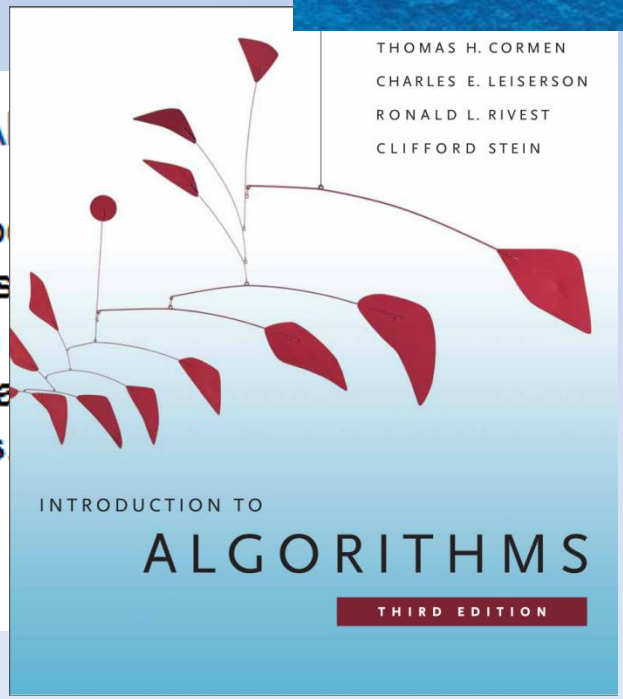
Units: 3
Course Typically Offered: Sprir



CSCI 174. Design and Analysis of A

Prerequisites: CSCI 115, CSCI 119. Mo
measures of complexity, algorithms for s
representation and manipulation, branch
polynomial arithmetic, pattern-matching a
graph algorithm, NP-complete problems

Units: 3



Grading

Period	Presentations/Examinations/Assignments	Points
Various	Presentations/In-Class	100
	Python (Assignment 1)	100
	Problem Solving w/ State-Space Search (Assignment 2)	100
	Problem Solving in Games (Assignment 3)	100
	Probabilistic Reasoning (Assignment 4)	100
	Midterm	200
	Final	300

Tests

- Midterm (20%)
- Final (30%)

Laptops

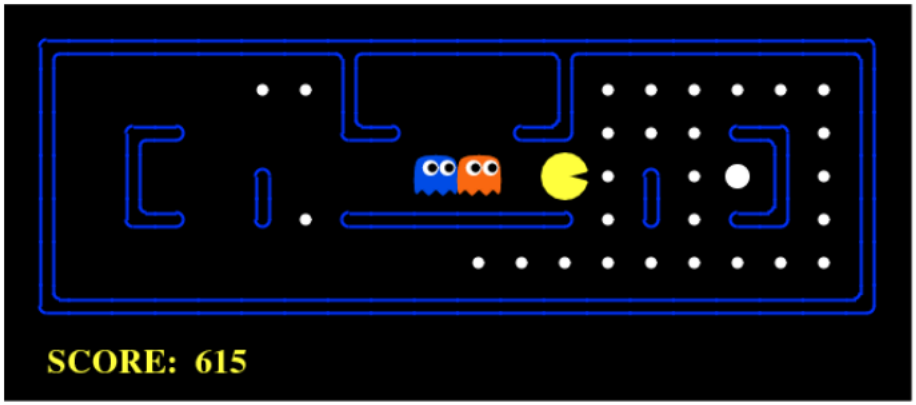
- In-Class testing using laptops.

Assignments (Programming)

- Assignment 1: Python Basics
- Assignment 2: State-Space Search
- Assignment 3: Games
- Assignment 4: Probabilistic Reasoning

Pacman Projects: Puzzles

The Pac-Man Projects



Overview

The Pac-Man projects were developed for UC Berkeley's introductory artificial intelligence course, CS 188. They apply an array of AI techniques to playing Pac-Man. However, these projects don't focus on building AI for video games. Instead, they teach foundational AI concepts, such as informed state-space search, probabilistic inference, and reinforcement learning. These concepts underly real-world application areas such as natural language processing, computer vision, and robotics.

We designed these projects with three goals in mind. The projects allow students to visualize the results of the techniques they implement. They also contain code examples and clear directions, but do not force students to wade through undue amounts of scaffolding. Finally, Pac-Man provides a challenging problem environment that demands creative solutions; real-world AI problems are challenging, and Pac-Man is too.

In our course, these projects have boosted enrollment, teaching reviews, and student

Pacman Projects: Dr. John DeNero

- John DeNero

John DeNero

Assistant Teaching Professor

Research Areas

Artificial Intelligence (AI)

Education (EDUC)

Research Centers

Berkeley Artificial Intelligence Research Lab (BAIR)

Teaching Schedule

Spring 2017

CS C8. Foundations of Data Science, MoWeFr 11:00AM - 11:59AM, Pimentel 1

CS 47A. Completion of Work in Computer Science 61A

Pacman Projects: Dr. John DeNero

- John DeNero

What and How

Research

- Publications

Teaching

- Past and Current Courses
- YouTube channel
- Composing Programs, a free online textbook for introductory computer science
- Computational and Inferential Thinking: The Foundations of Data Science, a free online textbook for introductory data science
- The Pac-Man projects for teaching artificial intelligence
- Instructor materials and info for Data 8
- Data Science Education at Berkeley

News and Awards

2017

- Appointed the inaugural Giancarlo Teaching Fellow (College of Engineering, Berkeley)
- Jim and Donna Gray Faculty Award for Undergraduate Teaching (EECS department, Berkeley)
- A UC Berkeley team was sponsored for the Alexa Prize. (Amazon)
- Microsoft Research Blog highlights the Berkeley Data Science Education Program. (Microsoft)
- What Is a Data Scientist, Anyway? an article describes the structure and goals of Berkeley's data science education program. (Wall Street Journal)
- CS 61A has record enrollment of 1760 enrolled + 150 waitlisted in Fall 2017. (The Daily Californian)

Pacman Projects
Dr. Dan Klein

Dan Klein

Professor
Computer Science Division
University of California at Berkeley

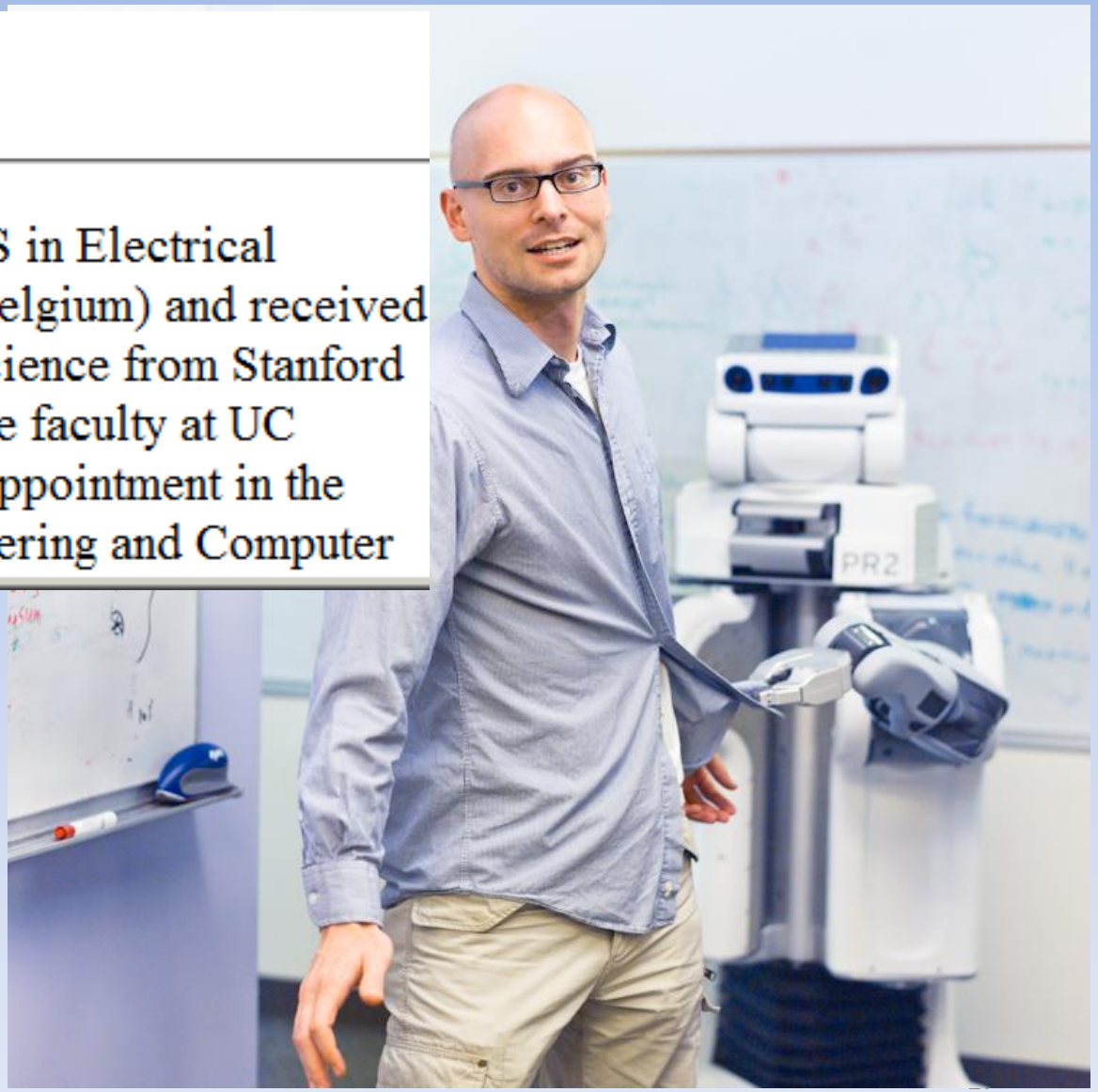
Contact Information

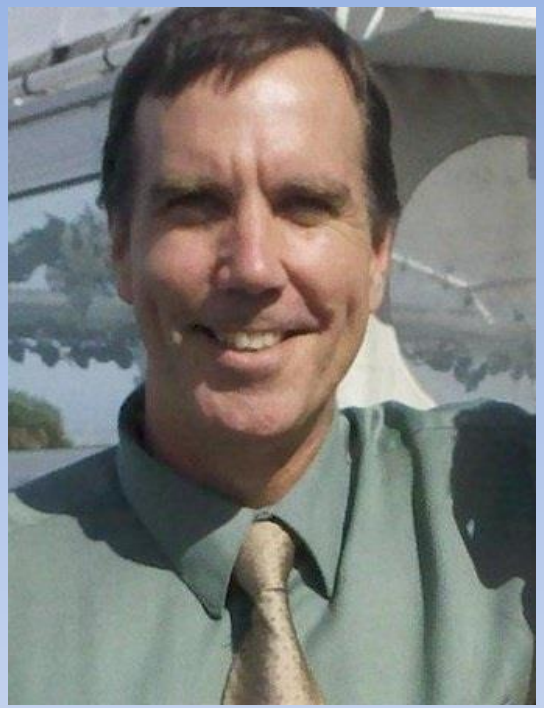
Email	klein@cs.berkeley.edu
Mail	Dan Klein, Soda Hall, Berkeley, CA
Phone	(510) 643-0805 (email works best)

Pacman Projects
Dr. Pieter Abbeel

Brief Bio

Pieter Abbeel received a BS/MS in Electrical Engineering from KU Leuven (Belgium) and received his Ph.D. degree in Computer Science from Stanford University in 2008. He joined the faculty at UC Berkeley in Fall 2008, with an appointment in the Department of Electrical Engineering and Computer





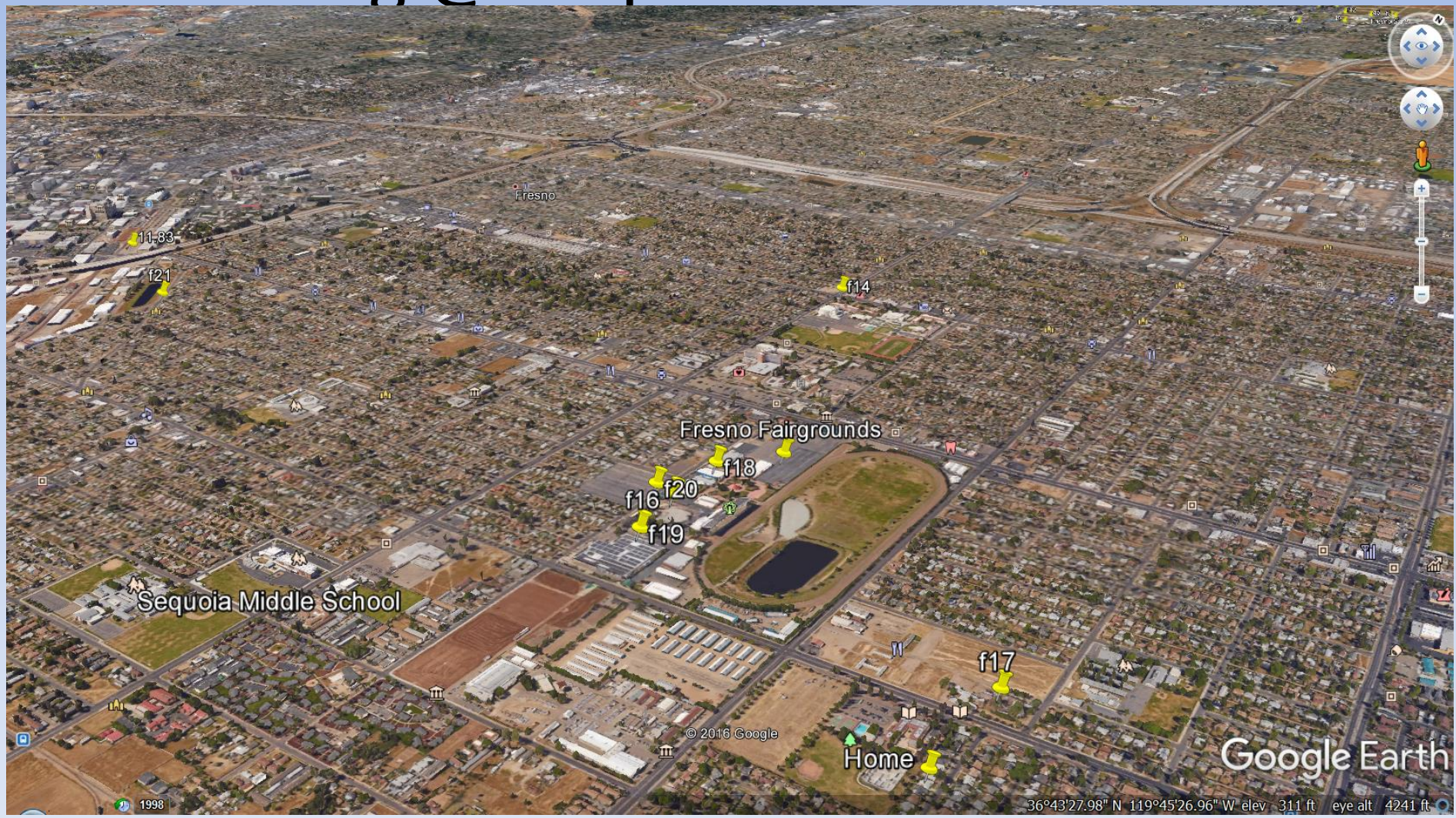
David Ruby

Class Instructor

- Office
  - Science II – 273
- Email:
  - druby@csufresno.edu

- First-Generation College Student
- How PhD?

Father Floyd Fresno Career Custodian Ending @ Sequoia Middle School




Home
Goldleaf
Fresno Fairgrounds
Fresno
North
Palmas
Tarpey
Fresno State
© 2015 Google

Compute Science Focus: Jobs/Degrees

- Students want…
  - Jobs!
  - Advanced Degrees!!

- How???

Dr. Joy Goto, Professor Biochemistry, Fresno State

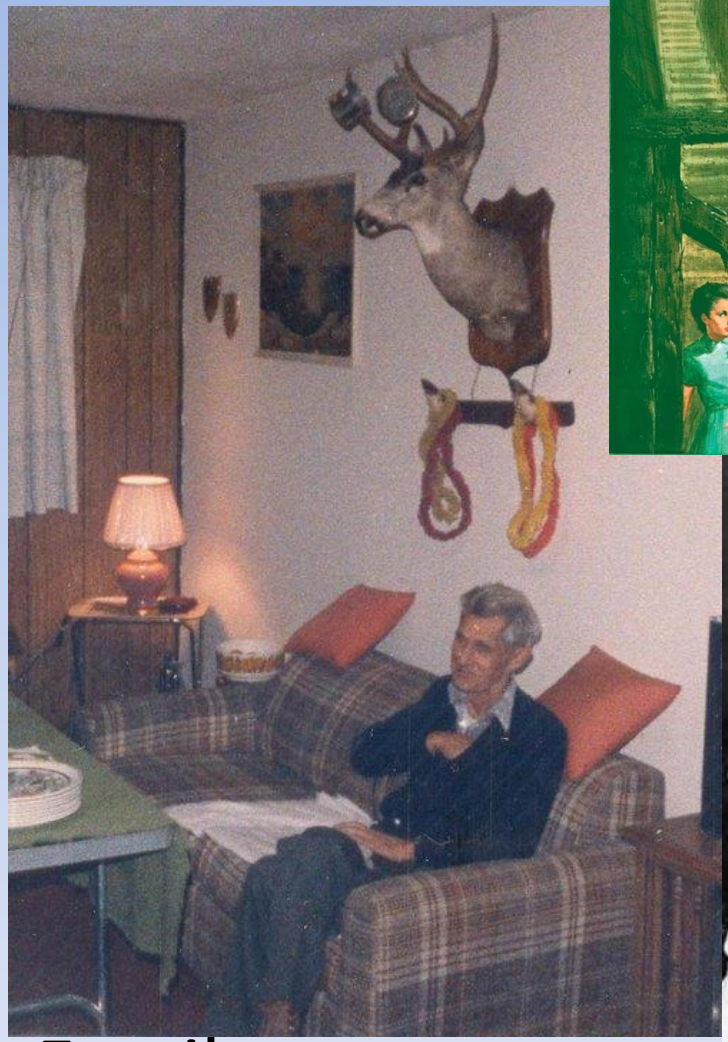
- https://youtu.be/FXUiEPrK_II

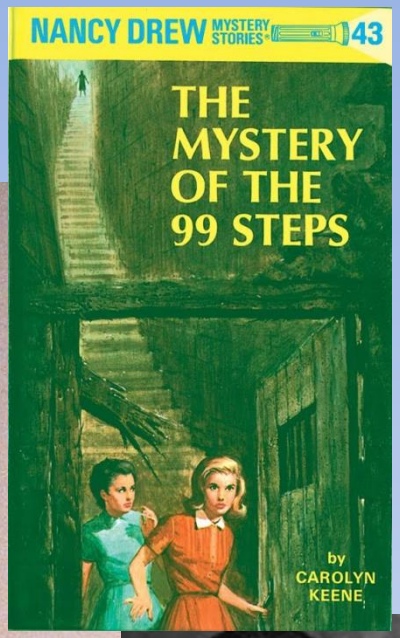
Dr. Joy Goto, Professor Biochemistry, Fresno State

- https://youtu.be/FXUiEPrK_II

- Engaging story of discovering joy in science growing up here in the central valley.

My Story



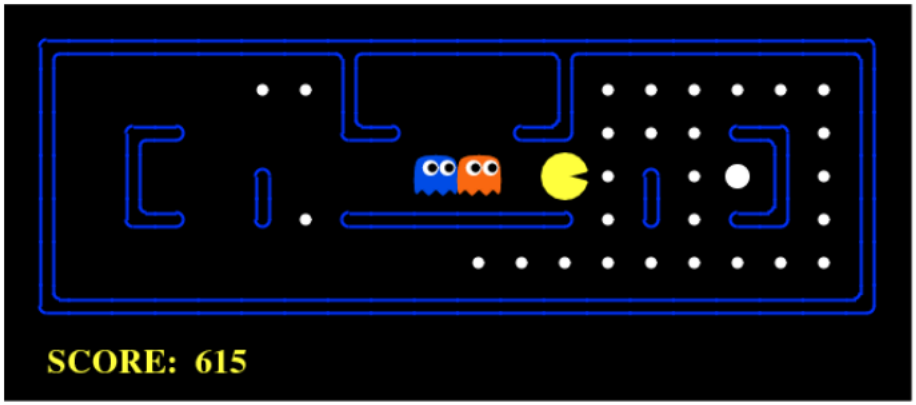


Interest In Puzzles

- Family Memories

Puzzles: Pacman Projects

The Pac-Man Projects



Overview

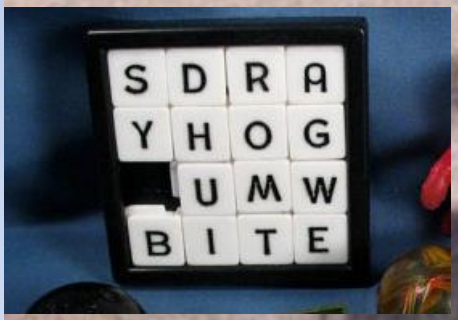
The Pac-Man projects were developed for UC Berkeley's introductory artificial intelligence course, CS 188. They apply an array of AI techniques to playing Pac-Man. However, these projects don't focus on building AI for video games. Instead, they teach foundational AI concepts, such as informed state-space search, probabilistic inference, and reinforcement learning. These concepts underly real-world application areas such as natural language processing, computer vision, and robotics.

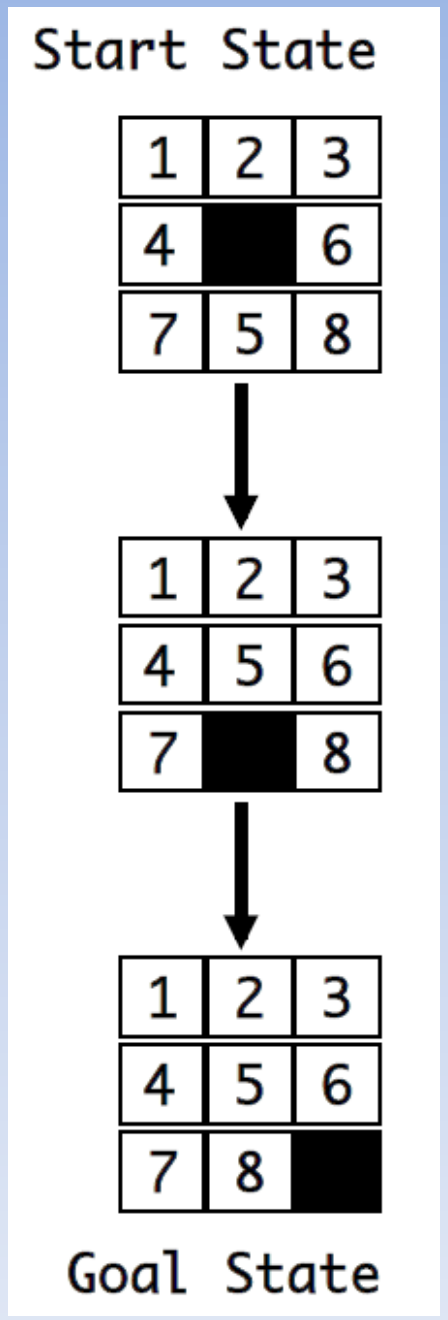
We designed these projects with three goals in mind. The projects allow students to visualize the results of the techniques they implement. They also contain code examples and clear directions, but do not force students to wade through undue amounts of scaffolding. Finally, Pac-Man provides a challenging problem environment that demands creative solutions; real-world AI problems are challenging, and Pac-Man is too.

In our course, these projects have boosted enrollment, teaching reviews, and student

Memories..
eXciting Puzzles !

- Home Hedges Maze Crawwwwwl !
- Also – First time w/ Sliding Tile Puzzle





Thesis: Tile-Sliding Puzzle

Artificial Intelligence

Branch: master ▾ ics1293 / Papers / Ruby.AAAI91-082.pdf Find file Copy path

everestso paper 017b894 4 days ago

1 contributor

1000 KB Raw History

SteppingStone: An Empirical and Analytical Evaluation*

David Ruby and **Dennis Kibler**
Department of Information & Computer Science
University of California, Irvine
Irvine, CA 92717 U.S.A.
druby@ics.uci.edu

Abstract

Decomposing a difficult problem into simpler subproblems is a classic problem solving technique. Unfortunately, the most difficult subproblems can be as difficult, if not more difficult, than the original problem. This is not an obstacle to problem solving if the difficult subproblems recur in other problems. If the difficult subproblems recur often, then its solution need only be learned once and reused. SteppingStone is a learning problem solver that decomposes a problem into simple and difficult-but-recurring subproblems. It solves the

SteppingStone operates on problems defined with a state space representation consisting of a set of goals, a set of operators, and an initial state. The goal orderer takes as input a set of goals. It orders these goals so that the constrained search method will likely solve them. It does this by ordering them so as to reduce the likelihood of subgoal interactions using a domain independent heuristic we call *openness* [Ruby and Kibler, 1989]. It produces an ordered set of subgoals as output.

The constrained search component takes as input an

My Idea…

Memories are constructed.. Not stored complete!

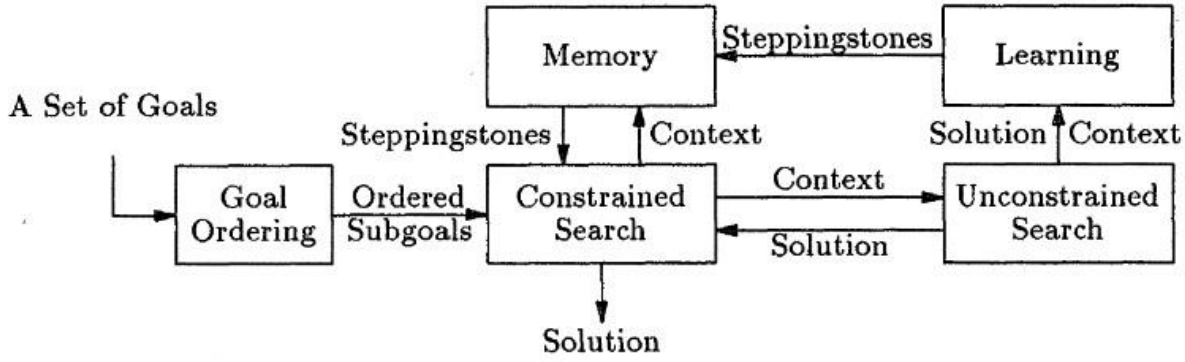


Figure 1: Overview of SteppingStone

the original impasse state.

When memory fails to return any useful stepping-stones the constrained search component calls the unconstrained search component. The unconstrained search component takes as input a context, just as the memory component did. Unconstrained search relaxes the protection on the solved subgoals in its search for a solution. If it resolves the impasse, it returns the sequence of moves found to the constrained search component. The unconstrained search component also sends its impasse solution, along with the context, to the learner.

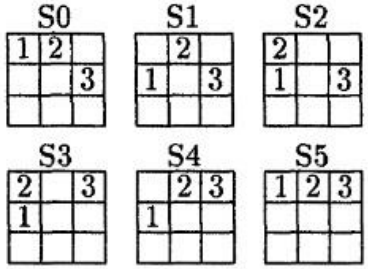
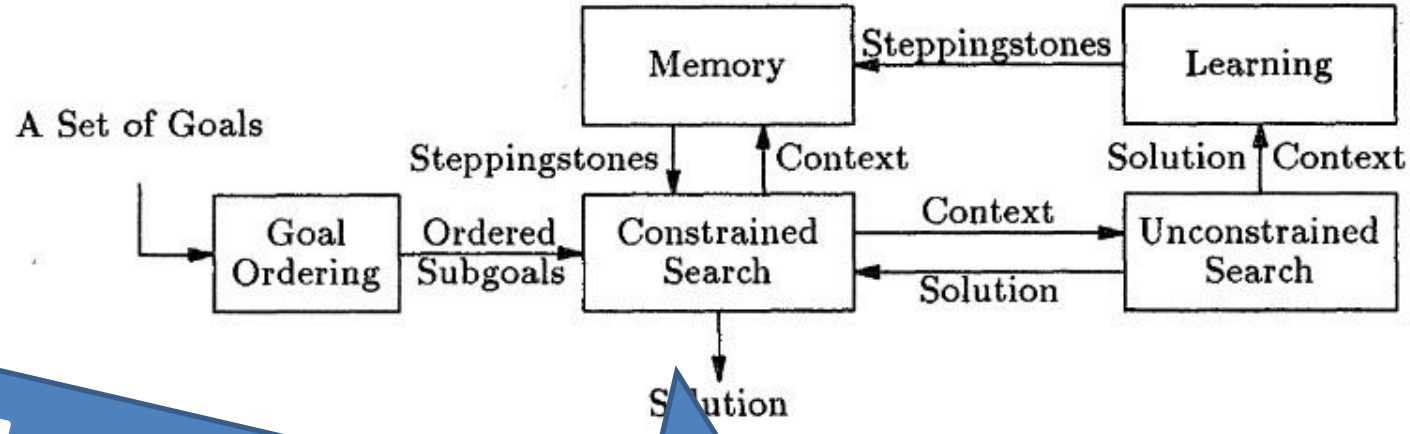


Figure 2: Steppingstones from Memory

Thesis: Tile Sliding Domain



Memories have gaps!!!

f SteppingStone

stones the constra
constrained search compo
search component takes as input a co
memory component did. Unconstrained search
the protection on the solved subgoals in its searc
a solution. If it resolves the impasse, it returns t
quence of moves found to the constrained search c po-
nent. The unconstrained search component also sends its impasse solution, along with the context, to the learner.

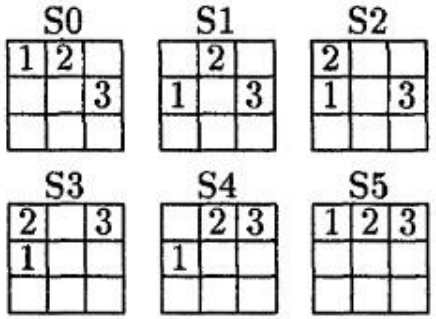
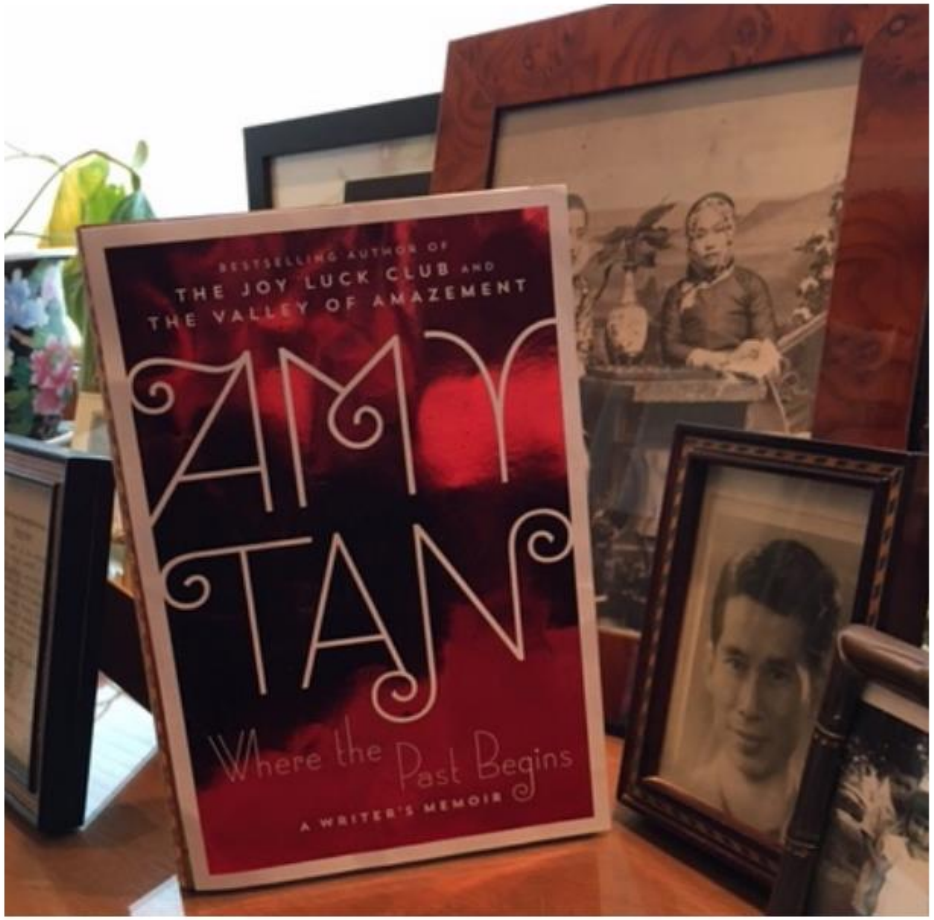
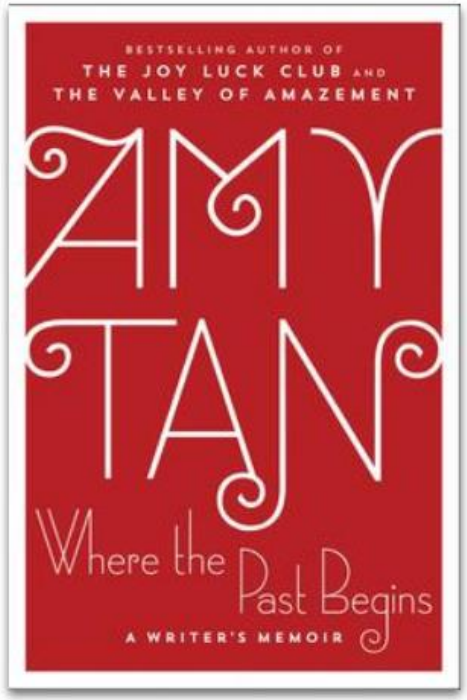


Figure 2: Steppingstones from Memory

Emotional Memory: Process of a Writer





coming
Oct 17, 2017
Pre-order

Malleable Memory (Gaps)
Learning & Memory w/ Elizabeth Loftus

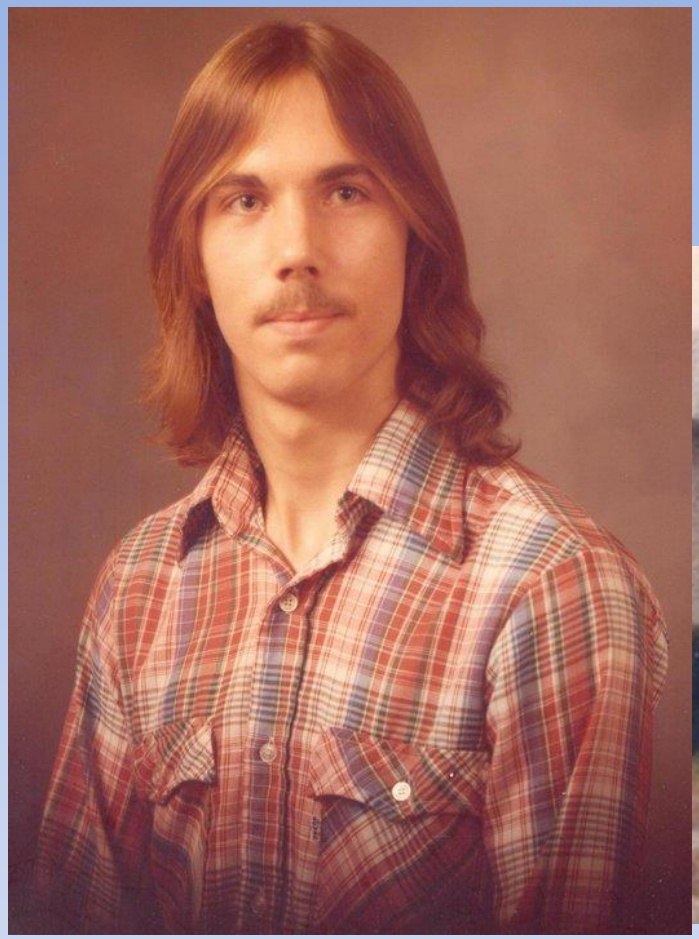
CORRUPTED MEMORY

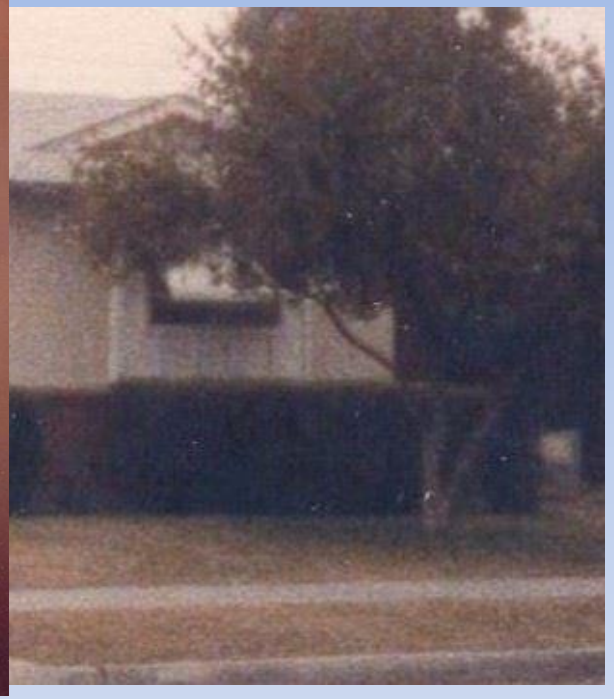
Elizabeth Loftus has spent decades exposing flaws in eyewitness testimony. Her ideas are gaining fresh traction in the US legal system.

BY MOHEB COSTANDI

Elizabeth Loftus is a cognitive psychologist at the University of California Irvine.

eXciting Mazes Memories!





- ME:
  - Do you remember the FUN maze?
- NEIGHBOR:
  - Do YOU remember this other HORRIBLE thing??
- ME:
  - Hmmm .. I guess not.
- Language influencing memory ??

Memories & Learning

JOURNAL OF VERBAL LEARNING AND VERBAL BEHAVIOR 13, 585–589 (1974)

Reconstruction of Automobile Destruction: An Example of the Interaction Between Language and Memory'

ELIZABETH F. LOFTUS AND JOHN C. PALMER

University of Washington

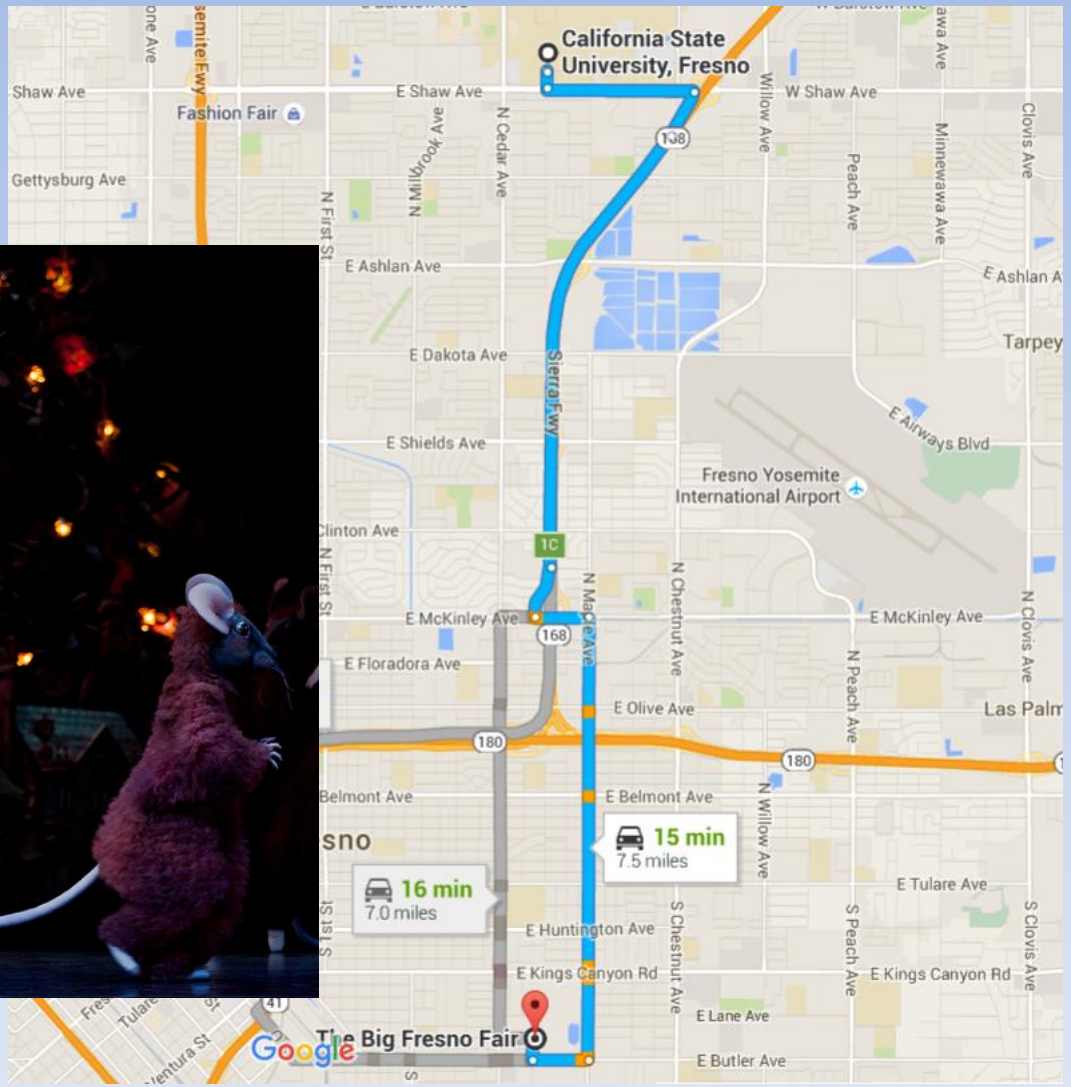
Two experiments are reported in which subjects viewed films of automobile accidents and then answered questions about events occurring in the films. The question, "About how fast were the cars going when they smashed into each other?" elicited higher estimates of speed than questions which used the verbs ***collided***, ***bumped***, ***contucted***, or ***hit*** in place of ***smashed***. On a retest one week later, those subjects who received the verb ***smashed*** were more likely to say "yes" to the question, "Did you see any broken glass?", even though broken glass was not present in the film. These results are consistent with the view that the questions asked subsequent to an event can cause a reconstruction in one's memory of that event.

Computer Science / Memories

- Puzzles
- Abstractions
- Memories

Current Interest: Abstraction

- Hello, World!



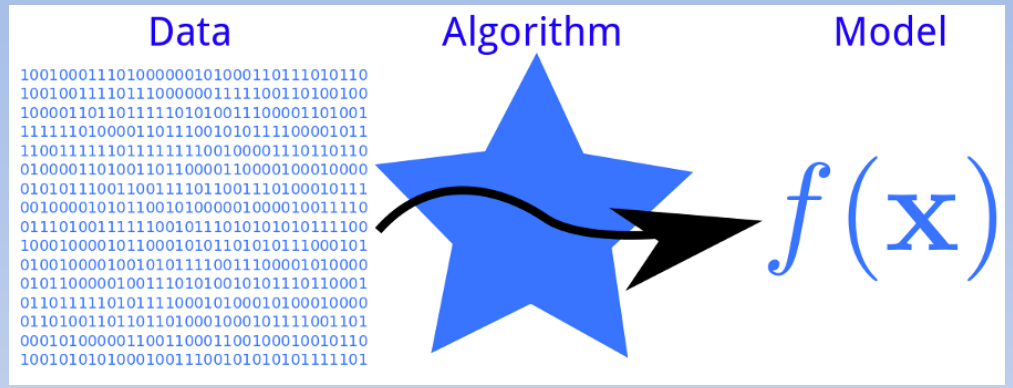
Abstraction: Computational Thinking

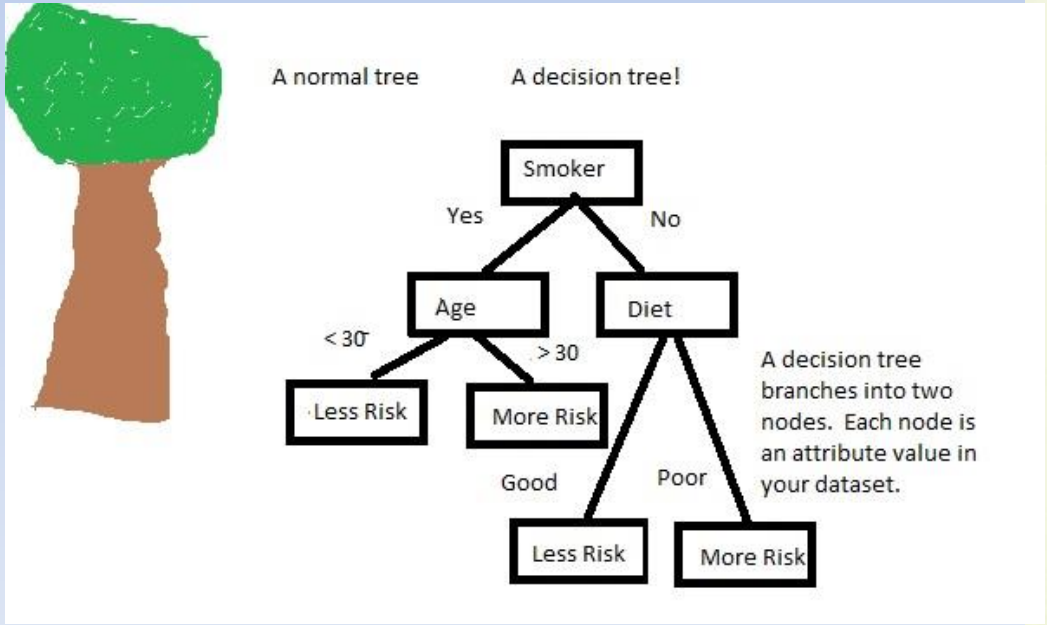
- Abstraction
- Automation
- Algorithms/Analysis

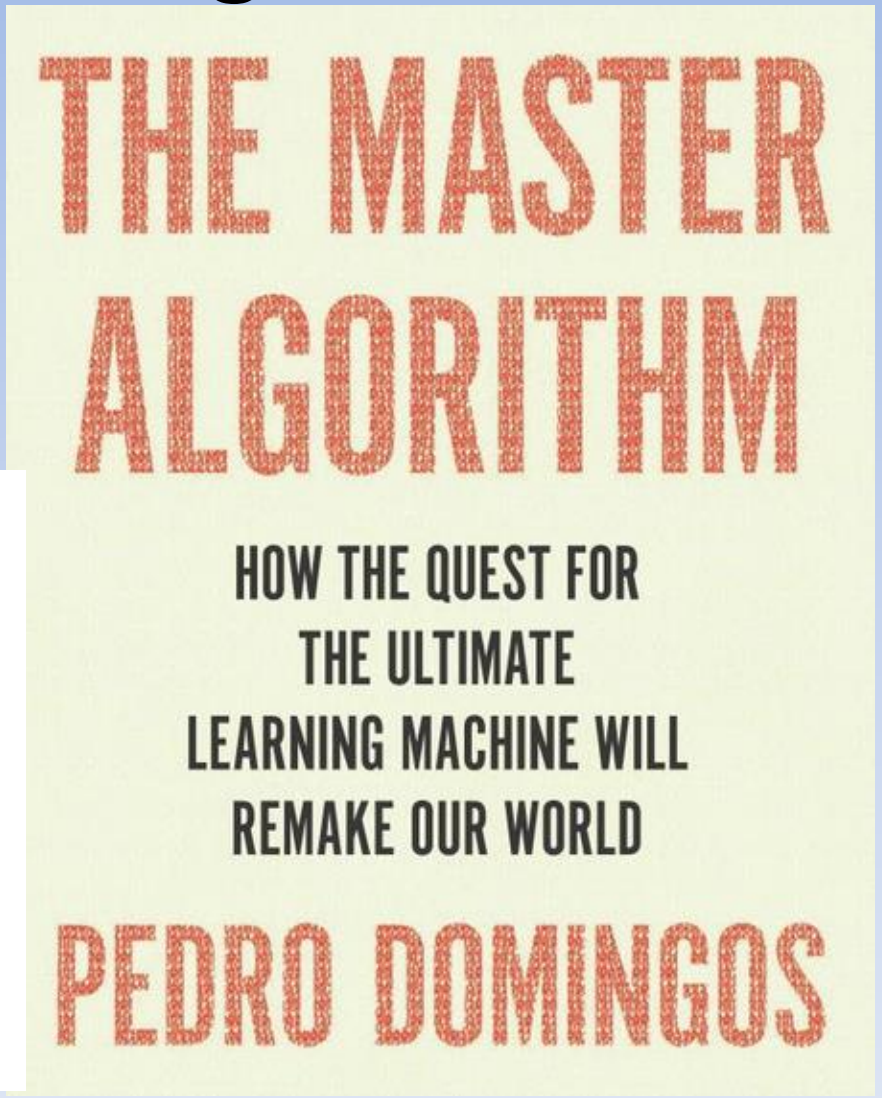
Intelligence (Problem Solving) Requires.. ..Learning from Experiences

- Intelligence requires learning from experiences.

Machine Learning - AI





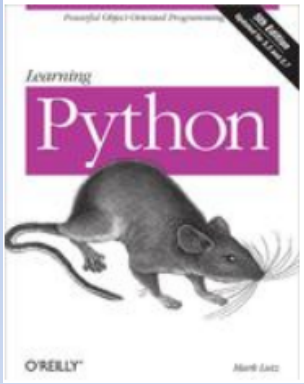


Additional Materials

Safari Books Online

Entire Site

Library Help California State University User Persona



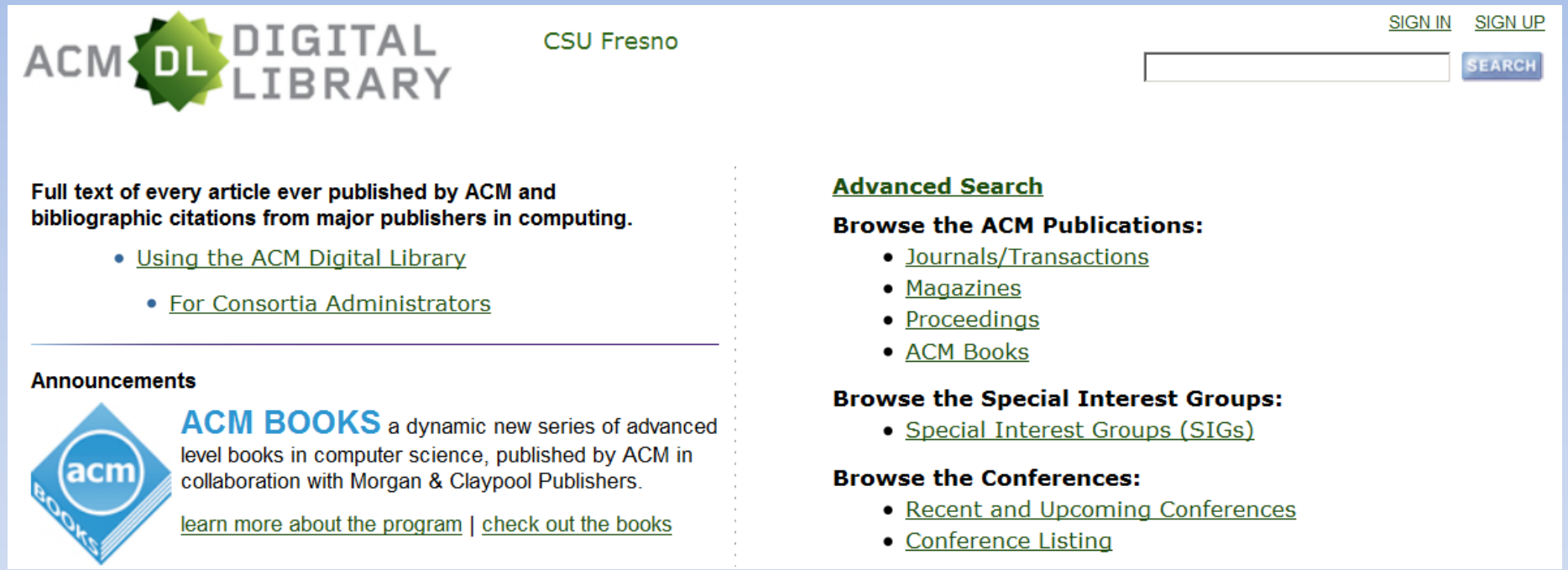
Learning Python, 5th Edition

By: Mark Lutz
Publisher: O'Reilly Media, Inc.
Pub. Date: June 26, 2013
Print ISBN-13: 978-1-4493-5573-9
Pages in Print Edition: 1600
Subscriber Rating: [14 Ratings] Subscriber Reviews

START READING

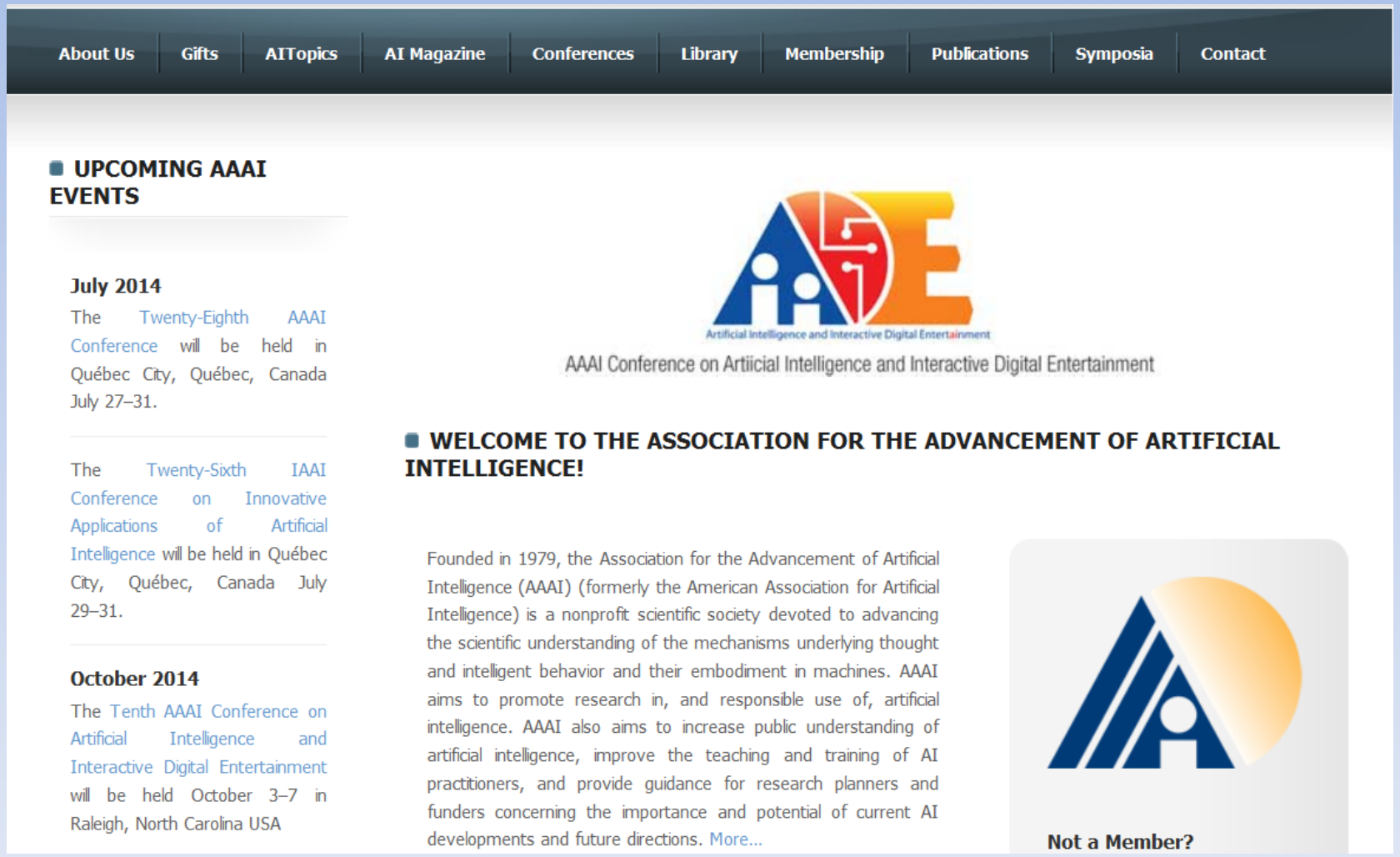
BUY FROM PUBLISI

Additional Materials

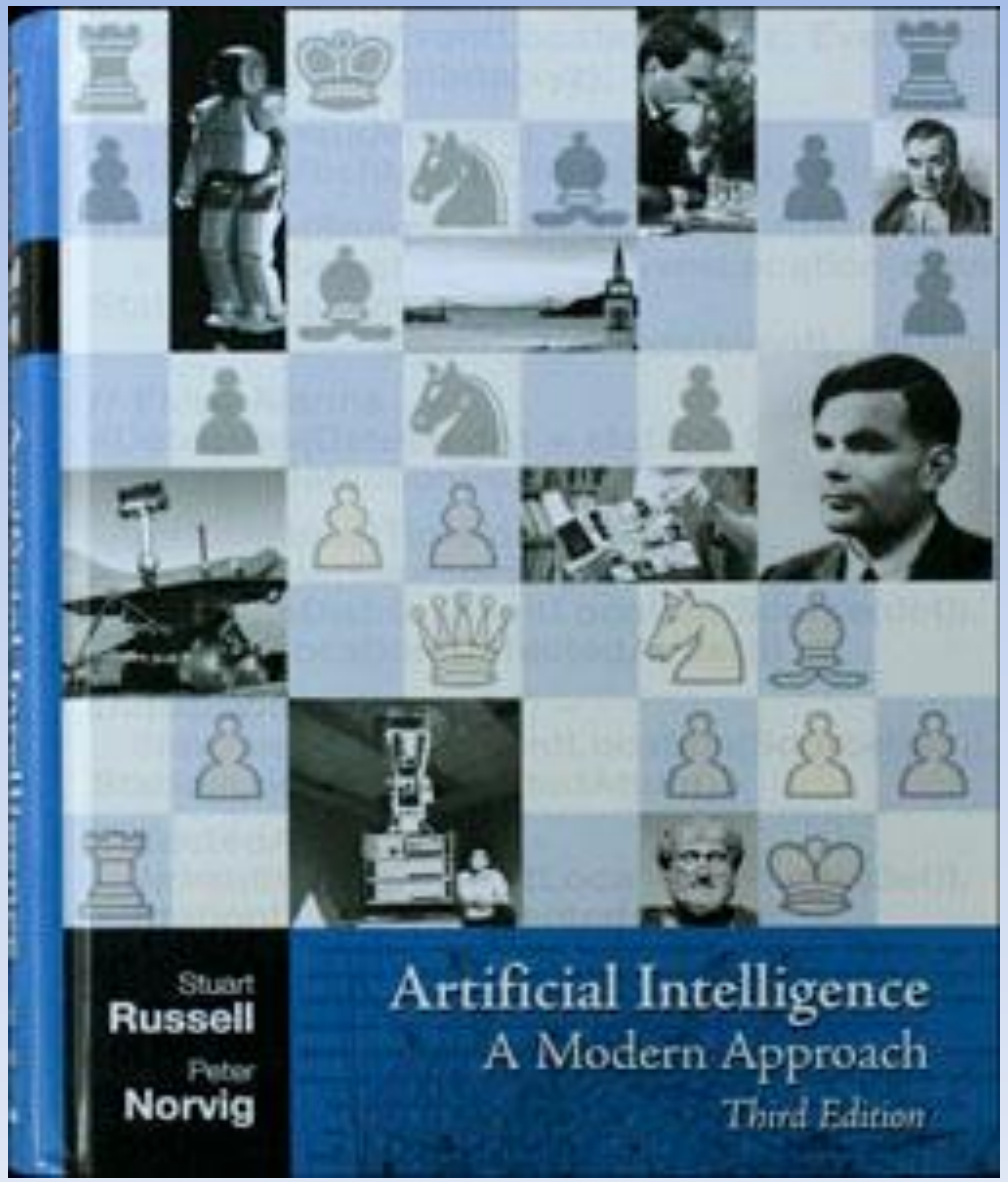


Additional Materials

AAAI.ORG
More Later



Tentative Course Schedule



- Chapters 1-7
- Chapters 13-15

Chapter 1
What is AI?

- What is Intelligence?

Chapter 1
What is AI?

- Introduce Science of Artificial Intelligence
- History of AI

Python 2.7

- Introduce Assignment 1:
- Python Introduction

Chapter 2: Intelligent Agents

- Agents operate in world
- Is Agent Intelligent?

Chapter 2: Problem Characteristics

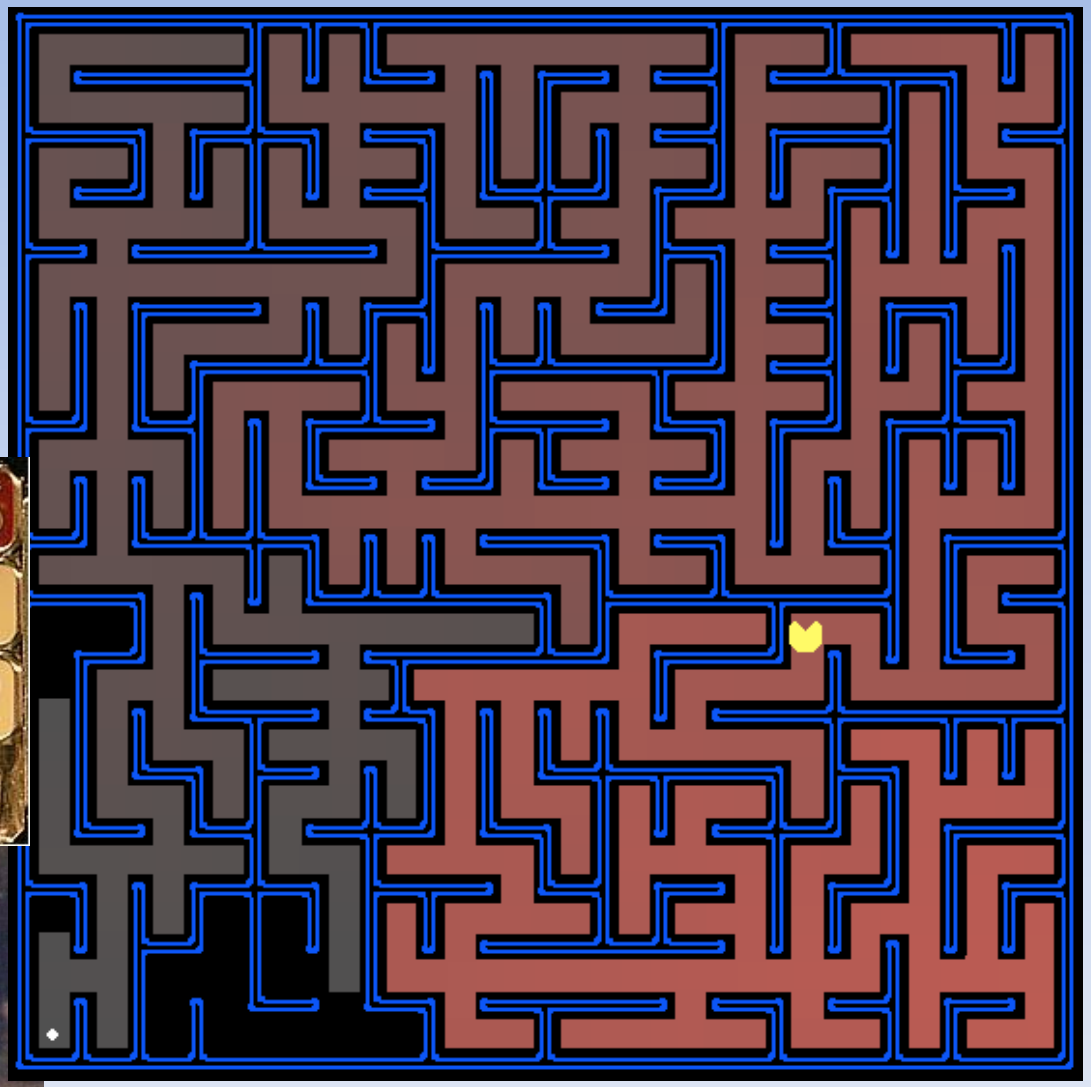
- Examine problems
  - Problems people solve
  - Problems not currently solved by computers
- Develop characteristics for describing problems.

Chapter 3:
State-Space Search

- Uniformed Search
- Heuristics
- Informed Search
  - A*

Assignment 2: The Maze

- Uniformed Search
- Heuristics
- Informed Search



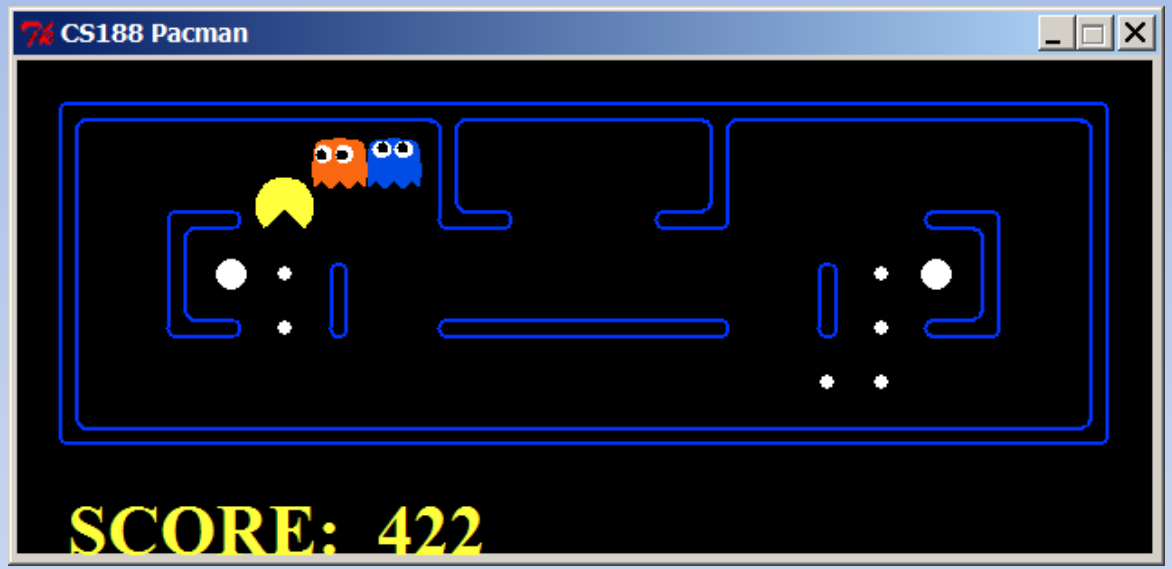
Chapter 4:
Beyond Classical Search

- Additional Search Methods
  - Beam Search
  - Genetic Algorithms
- Incomplete Knowledge
  - And/Or Trees

Chapter 5 Adversarial Search

- Multiplayer Games
- Minimax
- Alpha Beta Cutoff

Chapter 5/Assignment 3: Adversarial Search



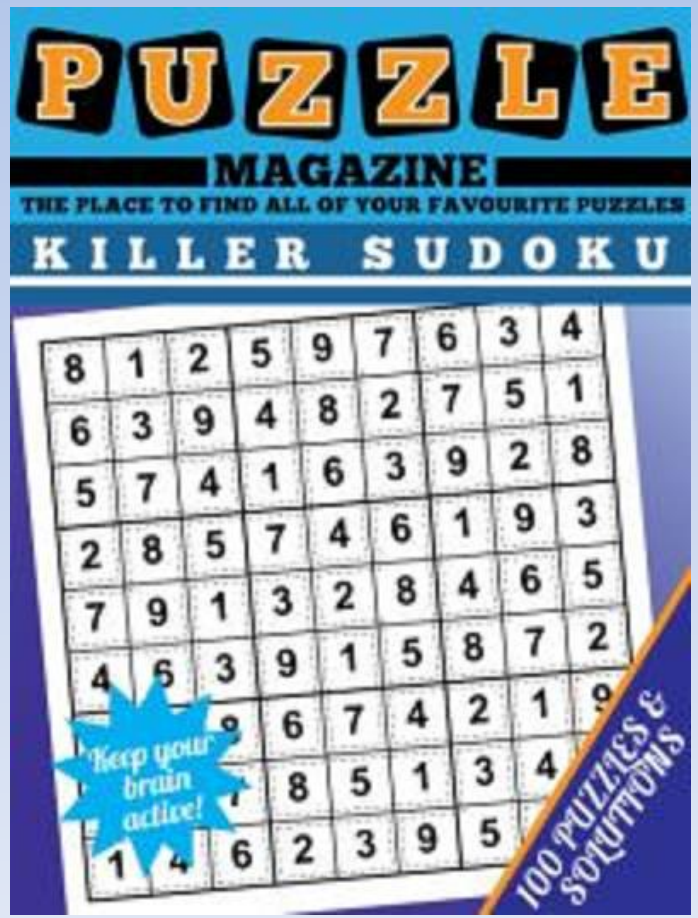
- Minimax, Alpha Beta, Expectimax

Midterm

- Chapters 1-5

Chapter 6: CSP

- Constraint Satisfaction Problems
- AC-3 Algorithm
- Cutsets
- Etc…



Chapter 7: Logical Agents

- Propositional Logic

P	Q	$\neg P$	$P \wedge Q$	$P \vee Q$	$P \Rightarrow Q$	$P \Leftrightarrow Q$
false	*false*	*true*	*false*	*false*	*true*	*true*
false	*true*	*true*	*false*	*true*	*true*	*false*
true	*false*	*false*	*false*	*true*	*false*	*false*
true	*true*	*false*	*true*	*true*	*true*	*true*

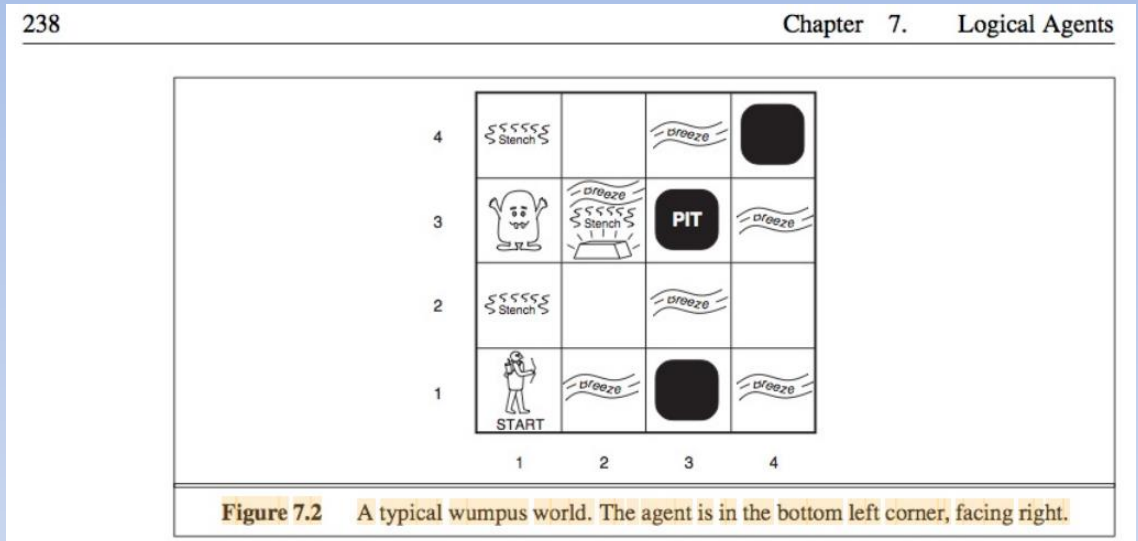
Figure 7.8 Truth tables for the five logical connectives. To use the table to compute, for
e and Q is false, first look on the left for the row
w). Then look in that row under the $P \vee Q$ column

$$
\begin{aligned}
(\alpha \wedge \beta) &\equiv (\beta \wedge \alpha) && \text{commutativity of } \wedge \\
(\alpha \vee \beta) &\equiv (\beta \vee \alpha) && \text{commutativity of } \vee \\
((\alpha \wedge \beta) \wedge \gamma) &\equiv (\alpha \wedge (\beta \wedge \gamma)) && \text{associativity of } \wedge \\
((\alpha \vee \beta) \vee \gamma) &\equiv (\alpha \vee (\beta \vee \gamma)) && \text{associativity of } \vee \\
\neg(\neg\alpha) &\equiv \alpha && \text{double-negation elimination} \\
(\alpha \Rightarrow \beta) &\equiv (\neg\beta \Rightarrow \neg\alpha) && \text{contraposition} \\
(\alpha \Rightarrow \beta) &\equiv (\neg\alpha \vee \beta) && \text{implication elimination} \\
(\alpha \Leftrightarrow \beta) &\equiv ((\alpha \Rightarrow \beta) \wedge (\beta \Rightarrow \alpha)) && \text{biconditional elimination} \\
\neg(\alpha \wedge \beta) &\equiv (\neg\alpha \vee \neg\beta) && \text{De Morgan} \\
\neg(\alpha \vee \beta) &\equiv (\neg\alpha \wedge \neg\beta) && \text{De Morgan} \\
(\alpha \wedge (\beta \vee \gamma)) &\equiv ((\alpha \wedge \beta) \vee (\alpha \wedge \gamma)) && \text{distributivity of } \wedge \text{ over } \vee \\
(\alpha \vee (\beta \wedge \gamma)) &\equiv ((\alpha \vee \beta) \wedge (\alpha \vee \gamma)) && \text{distributivity of } \vee \text{ over } \wedge
\end{aligned}
$$

Figure 7.11 Standard logical equivalences. The symbols α, β, and γ stand for arbitrary sentences of propositional logic.

Chapter 7: Logical Agents w/ Wumpus World

- Wumpus World



$P_{x,y}$ is true if there is a pit in $[x, y]$.
$W_{x,y}$ is true if there is a wumpus in $[x, y]$, dead or alive.
$B_{x,y}$ is true if the agent perceives a breeze in $[x, y]$.
$S_{x,y}$ is true if the agent perceives a stench in $[x, y]$.

PROBLEMS!!!

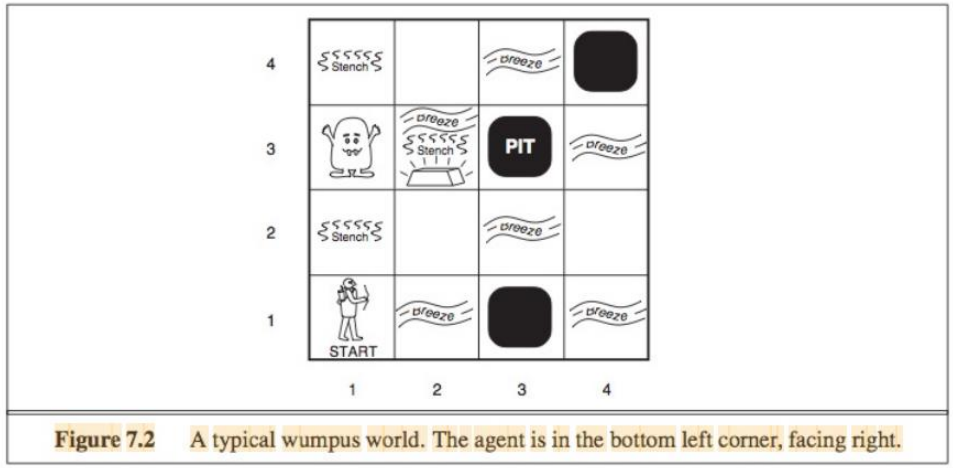
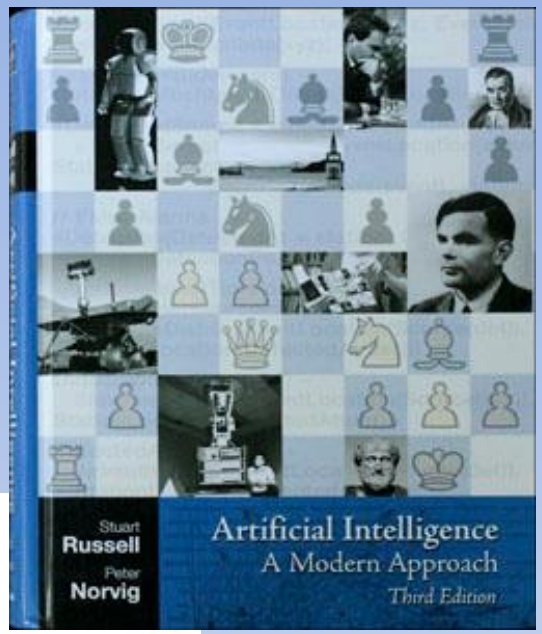


Figure 7.2 A typical wumpus world. The agent is in the bottom left corner, facing right.

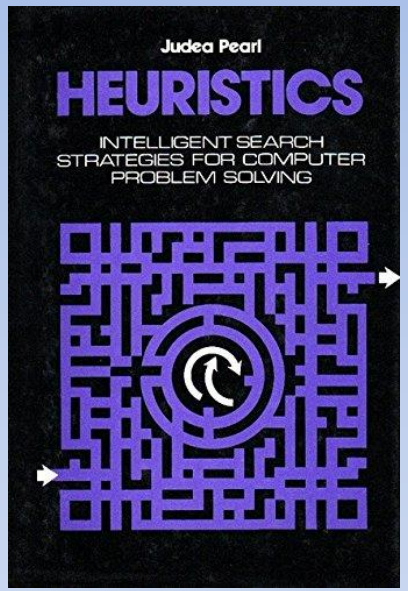
$P_{x,y}$ is true if there is a pit in $[x, y]$.
$W_{x,y}$ is true if there is a wumpus in $[x, y]$, dead or alive.
$B_{x,y}$ is true if the agent perceives a breeze in $[x, y]$.
$S_{x,y}$ is true if the agent perceives a stench in $[x, y]$.

Uncertainty

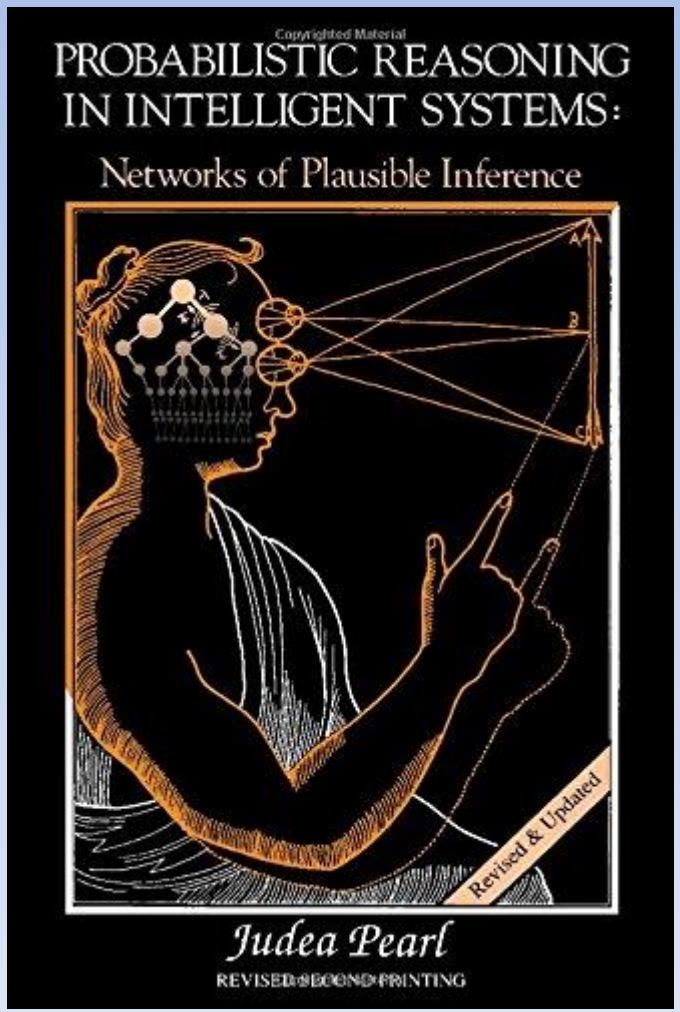
Dealing w/ Uncertainty



Judea Pearl
Heuristics -> Probabilistic Reasoning

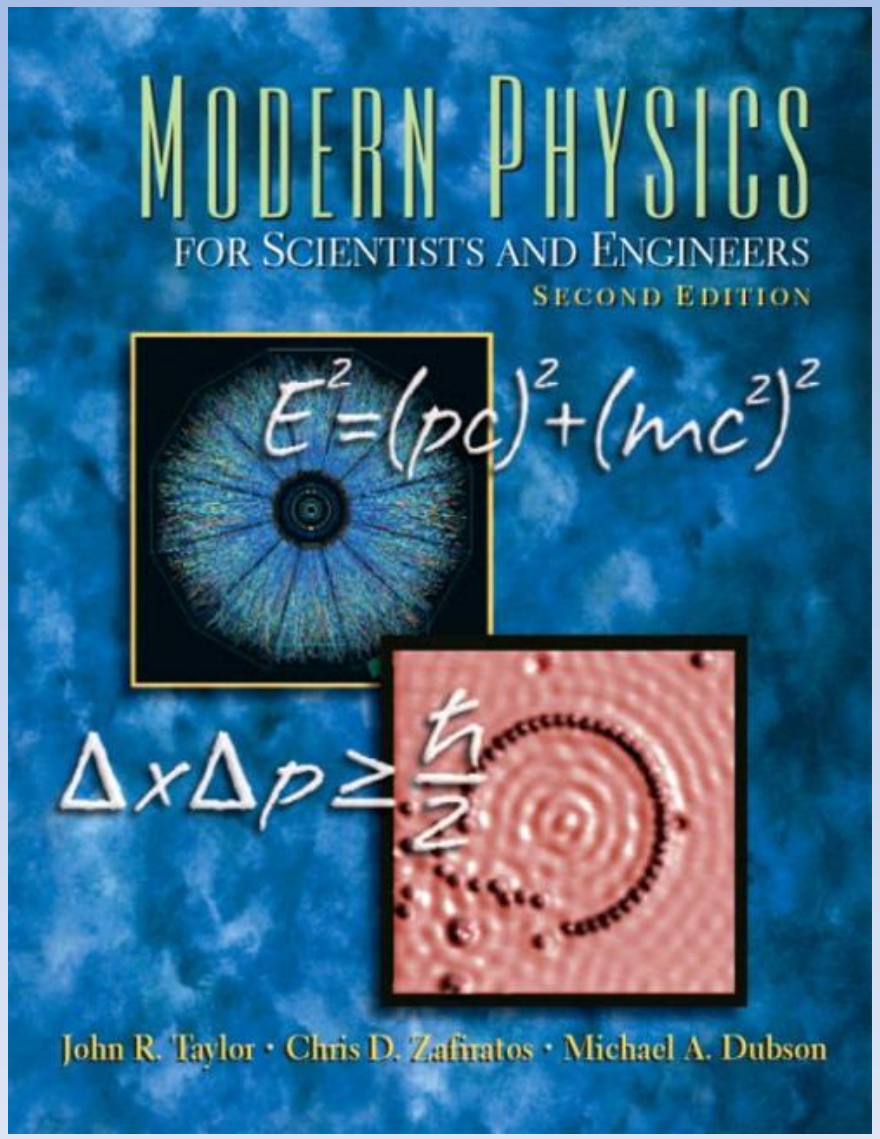


- April 1984



- September 15, 1988

Uncertainty in Science



Classical Mechanics

Probability & Physics

- Albert Einstein
- Quantum mechanics is certainly imposing. But an inner voice tells me that it is not yet the real thing. The theory says a lot, but does not really bring us any closer to the secret of the "old one." **I, at any rate, am convinced that *He* does not throw dice.**
  - Letter to Max Born (4 December 1926); *The Born-Einstein Letters* (translated by Irene Born) (Walker and Company, New York, 1971) ISBN 0-8027-0326-7.
- In a 1943 conversation with William Hermanns recorded in Hermanns' book *Einstein and the Poet*, Einstein said: **"As I have said so many times, God doesn't play dice with the world."** (p. 58).

Uncertainty in Science

Schrödinger's Equation

$$i\hbar \frac{\partial}{\partial t}\psi(\mathbf{r},t) = -\frac{\hbar^2}{2m}\nabla^2\psi(\mathbf{r},t) + V(\mathbf{r},t)\psi(\mathbf{r},t)$$

i is the imaginary number, $\sqrt{-1}$.

$\hbar$ is Planck's constant divided by 2π: 1

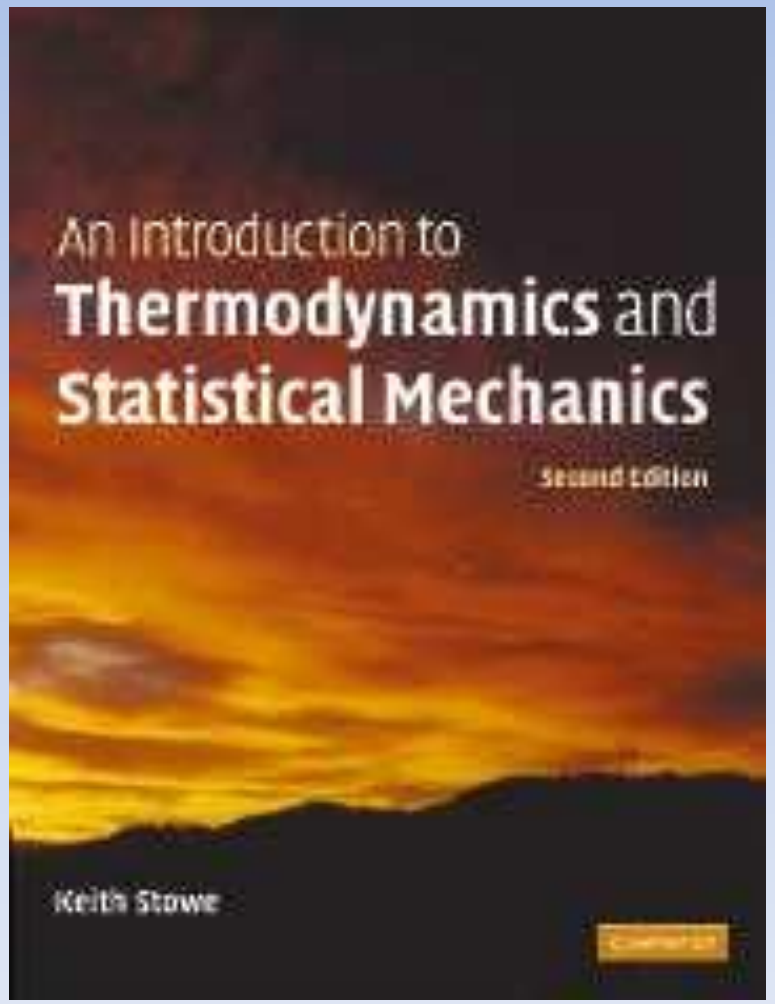
$\psi(\mathbf{r},t)$ is the wave function, defined ov

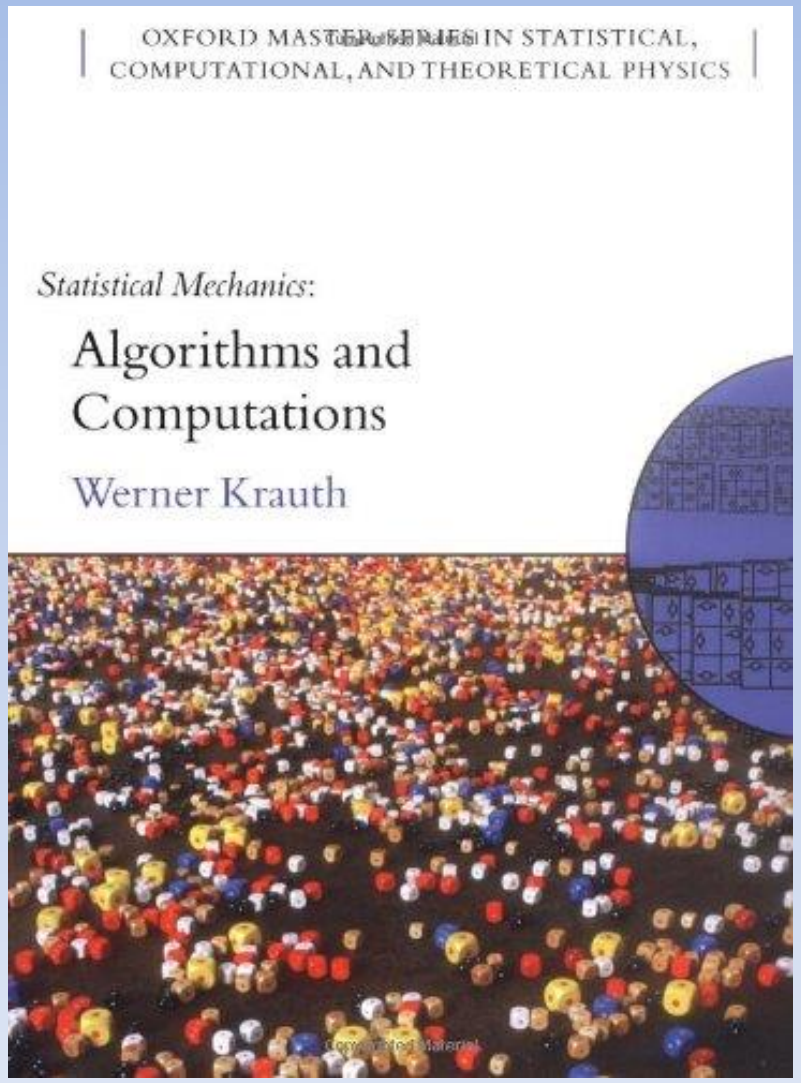
m is the mass of the particle.

∇^2 is the Laplacian operator, $\frac{\partial^2}{\partial x^2} + \frac{\partial}{\partial}$

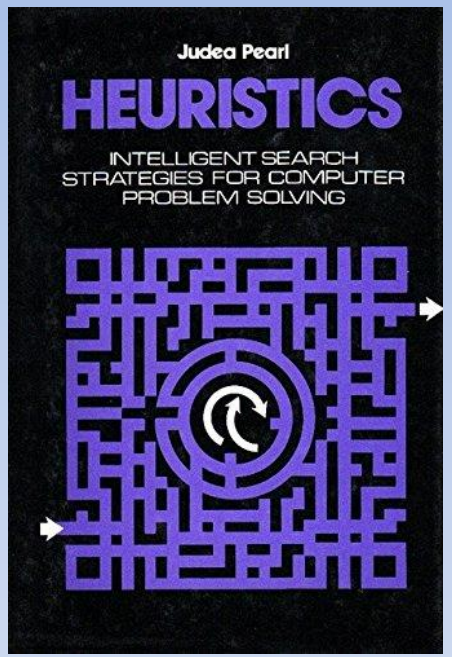
$V(\mathbf{r},t)$ is the potential energy influencin

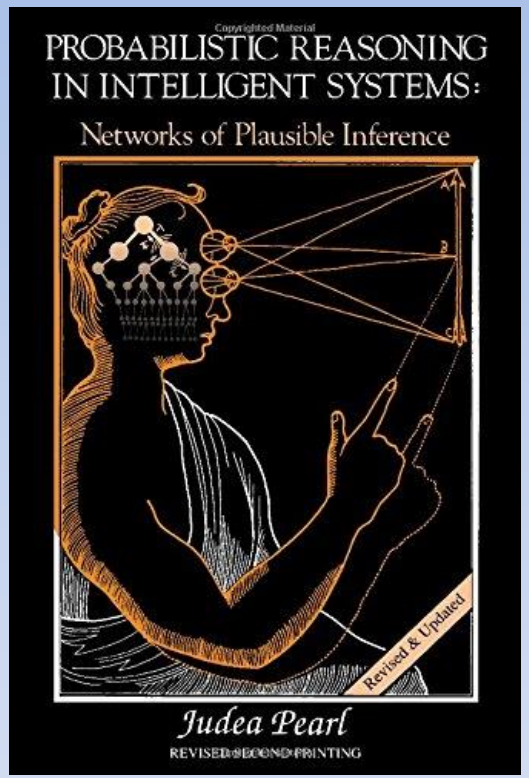
Uncertainty in Science
Classical Mechanics versus Statistical Mechanics

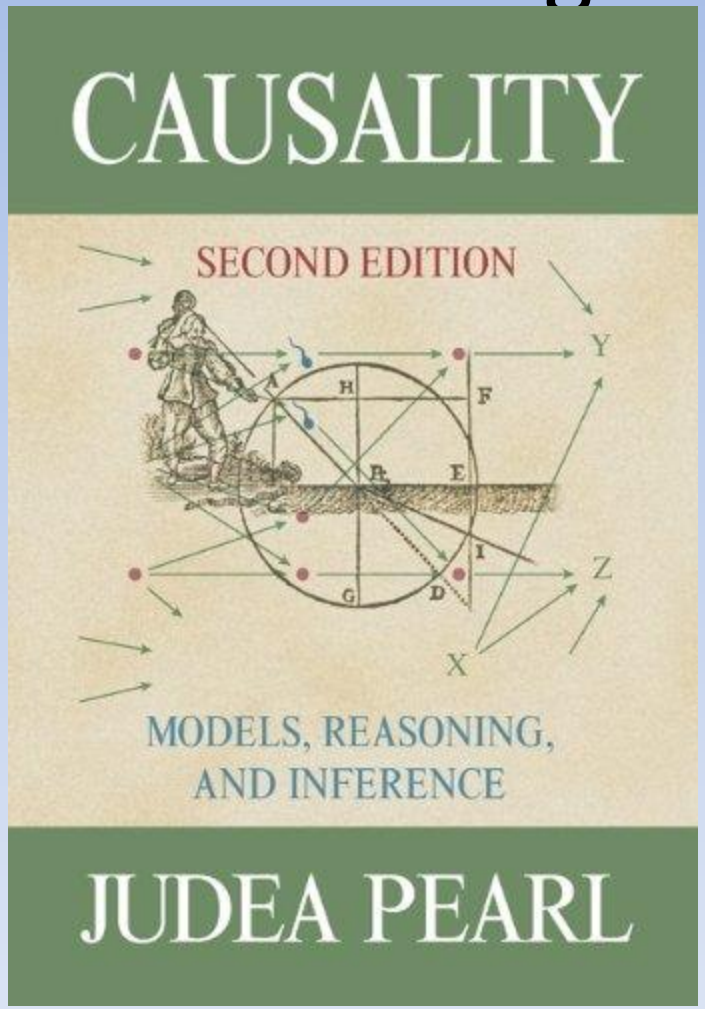




Judea Pearl
Heuristics -> Probabilistic Reasoning







- April 1984
- September 15, 1988
- September 14th, 2009

History of Probability (AI Text)

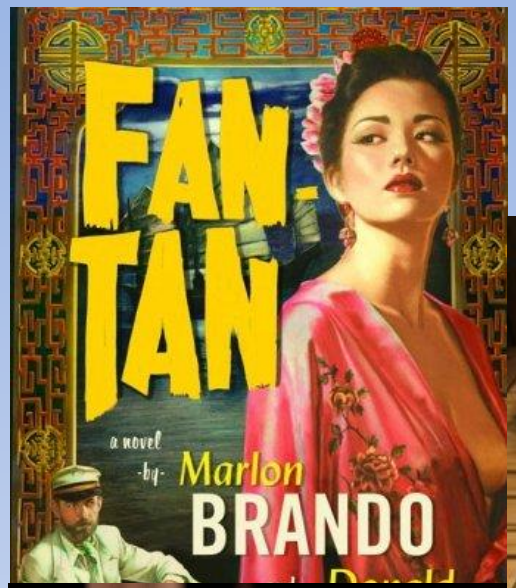
- In about 850 A.D. the Indian mathematician Mahaviracarya described how to arrange a set of bets that can't lose (what we now call a Dutch book).

Mahāvīra (mathematician)

From Wikipedia, the free encyclopedia

Mahāvīra (or **Mahaviracharya**, "Mahavira the Teacher") was a 9th-century Jain mathematician from Mysore, India.[1][2][3] He was the author of *Gaṇitasārasaṅgraha* (or *Ganita Sara Samgraha*, c. 850), which revised the Brāhmasphuṭasiddhānta.[1] He was patronised by the Rashtrakuta king Amoghavarsha.[4] He separated astrology from mathematics. It is the earliest Indian text entirely devoted to mathematics.[5] He expounded on the same subjects on which Aryabhata and Brahmagupta contended, but he expressed them more clearly. His work is a highly syncopated approach to algebra and the emphasis in much of his text is on developing the techniques necessary to solve algebraic problems.[6] He is highly respected among Indian mathematicians, because of his establishment of terminology for concepts such as equilateral, and isosceles triangle; rhombus; circle and semicircle.[7] Mahāvīra's eminence spread in all South India and his books proved inspirational to other mathematicians in Southern India.[8] It was translated into Telugu language by Pavuluri Mallana as *Saar Sangraha Ganitam*.[9]

Uncertainty w/ Gambling Games



Gambling Concepts

- Games of Chance
  - Poker
  - Roulette
  - Fan Tan
- Odds
  - Sports Book
  - Horse Racing

1565 Girolamo Cardano

- In Europe, the first significant systematic analyses were produced by Girolamo Cardano around 1565, although publication was posthumous (1663).

- Gambling Motivated

Meaning and Probability Theory

- 20 to 1 in the 5th
  - What does 20 to 1 Mean?
  - Where does 20 come from?

Joints & Marginals

		Intelligence		
		low	high	
Grade	A	0.07	0.18	0.25
	B	0.28	0.09	
	C	0.35	0.03	
			0.3	1.0

- P(Intelligence=high) = ?

More Problems…

		Intelligence		
		low	high	
Grade	A	0.07	0.18	0.25
	B	0.28	0.09	
	C	0.35	0.03	
			0.3	1.0

- P(Intelligence=high) = ?

INTRACTABILITY

Computational Thinking!!! Algorithms & Data Structures

- Representation
  - Bayesian Networks
- Algorithms
  - Exact Inference
  - Approximate Inference

PLAN
The Bayesian Network (Chapter 14)

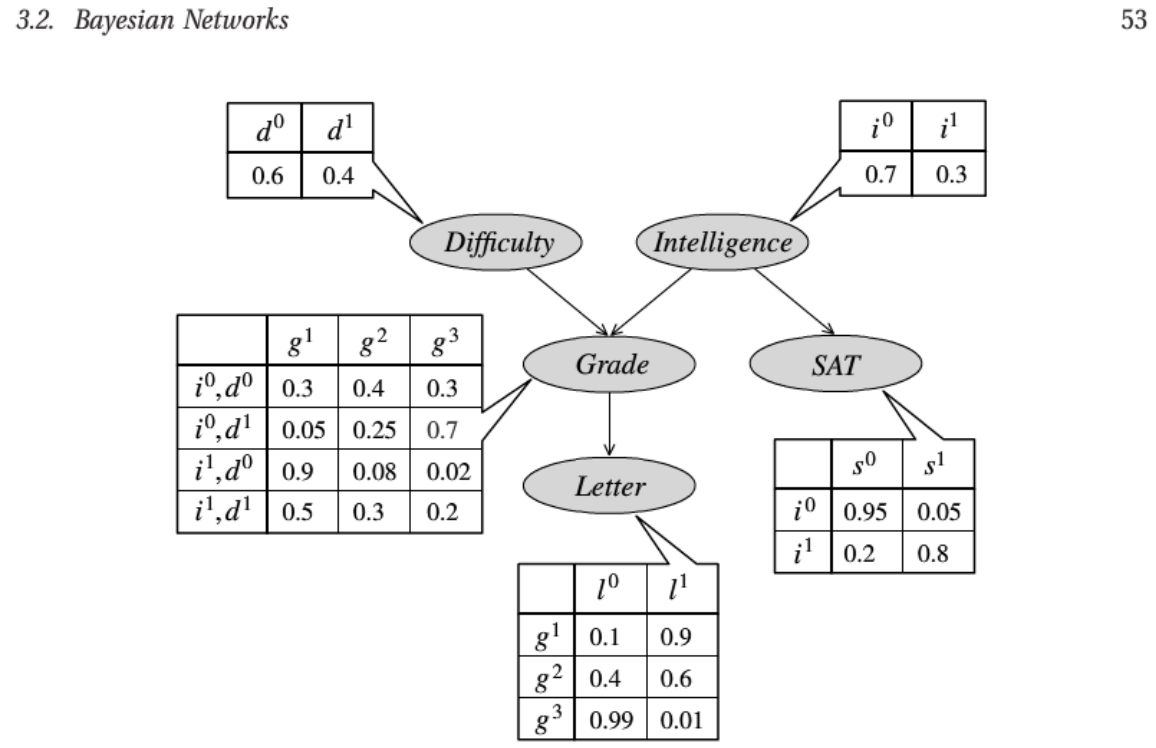
- Exploit Islands of Tractability in High Dimensional Space Probability Distributions
  - Worst Case is Intractable
  - Real World Frequently NOT Worst Case
  - Efficiently Exploit properties in Real World Probability Distributions to induce Tractability

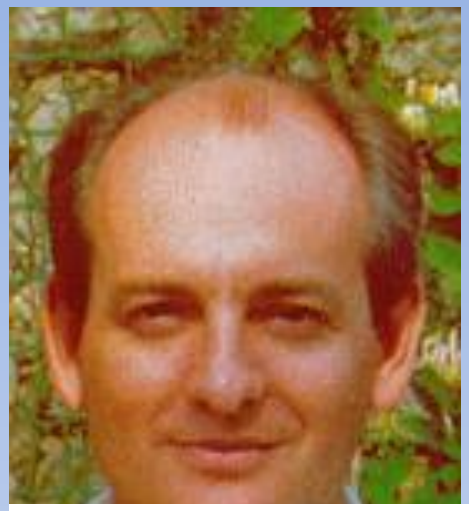
- Primary Tool:

INDEPENDENCE!

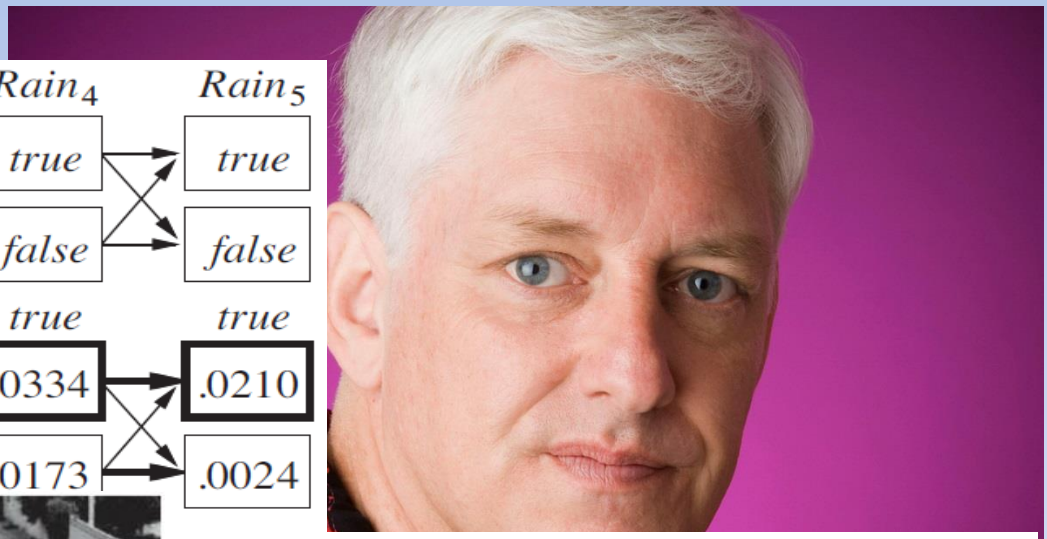
Bayesian Networks : (CPD's)

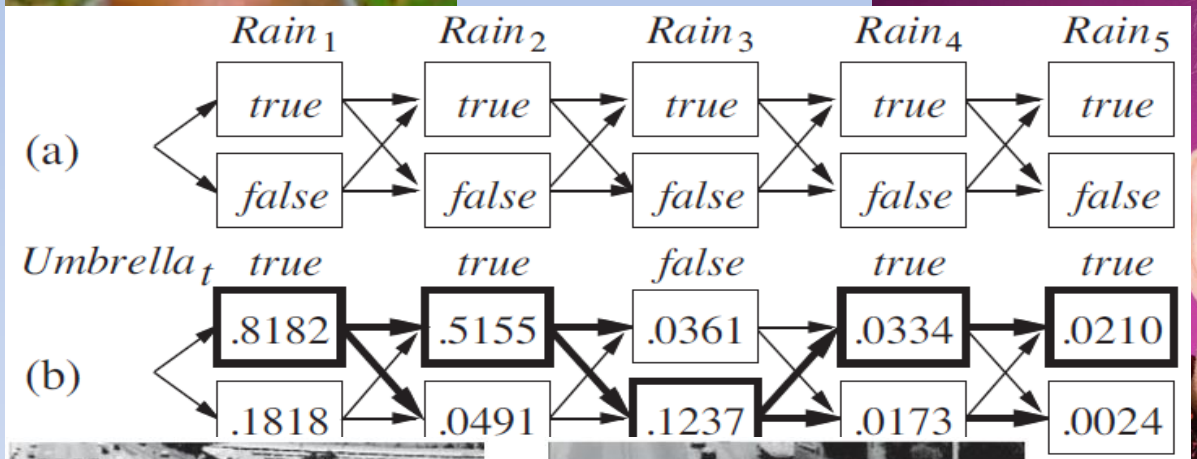
- Each variable is associated with a conditional probability distribution (CPD) that specifies a distribution CPD over the values of X given each possible joint assignment of values to its parents in the model.
- For a node with no parents, the CPD is conditioned on the empty set of variables.





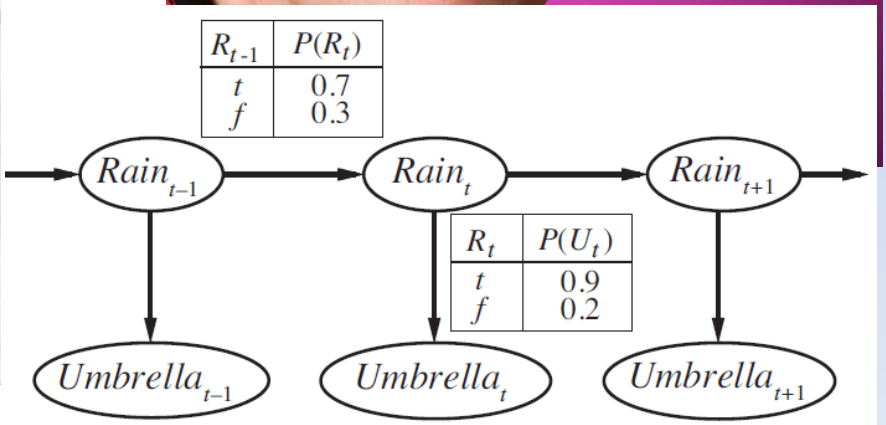
Chapter 15: Probabilistic Reasoning over Time





(a)

(b)



Where's Ghost?
Ghostbusters (Assign 4)!

Final

- Chapters 1-7, 13, 14, 15

Grading

Period	Presentations/Examinations/Assignments	Points
Various	Presentations/In-Class	100
	Python (Assignment 1)	100
	Problem Solving w/ State-Space Search (Assignment 2)	100
	Problem Solving in Games (Assignment 3)	100
	Probabilistic Reasoning (Assignment 4)	100
	Midterm	200
	Final	300