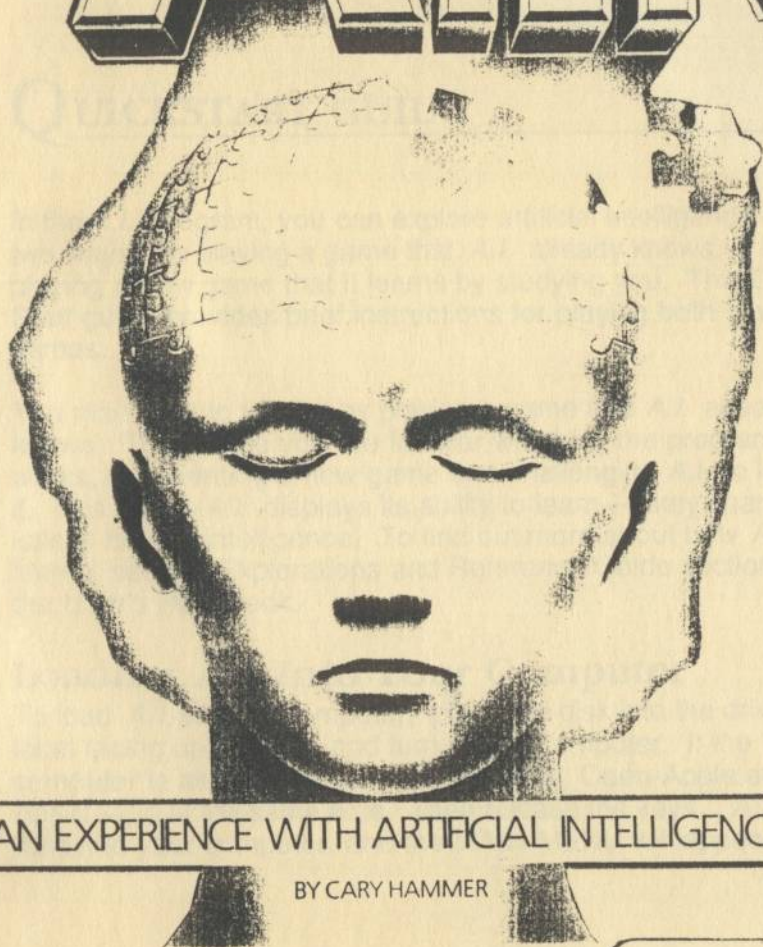


AI



AN EXPERIENCE WITH ARTIFICIAL INTELLIGENCE

BY CARY HAMMER

Apple IIe (128K),
IIc, IIGS

Scanned by cvxmelody

<http://www.cvxmelody.net/AppleUsersGroupSydneyAppleIIDiskCollection.htm>



QUICKSTART GUIDE

In the *A.I.* program, you can explore artificial intelligence in two ways: by playing a game that *A.I.* already knows, or by playing a new game that it learns by studying you. This Quick Start guide provides brief instructions for playing both types of games.

You may want to start by playing a game that *A.I.* already knows. Then, once you are familiar with how the program works, try inventing a new game and challenging *A.I.* to learn it. This is how *A.I.* displays its ability to learn—a key characteristic of human intelligence. To find out more about how *A.I.* learns, see the Explorations and Reference Guide sections of the User's Handbook.

Loading *A.I.* Into Your Computer

To load *A.I.* into the computer, place the disk into the drive, label facing up and out, and turn on the computer. If the computer is already on, press the Control, Open-Apple and Reset keys at the same time. Then release the keys. *A.I.* will load into your computer, and the MAIN MENU will appear.

Playing a Game *A.I.* Already Knows

When *A.I.* learns or is programmed to play a game, it stores that knowledge in an electronic "brain." These brains are in turn stored on "Brain Bank" disks. Side 2 of your program disk is such a Brain Bank disk. It contains seven brains. All of these brains play different games. Some play games very well, while others play poorly.

To play one of these brains, follow this procedure:

1. Select Brain Bank from the MAIN MENU.
2. From the Brain Bank screen, select Load a Brain.
3. Use the arrow keys to indicate the brain you want to play against, and then press Return. For a sample game, choose Challenge1, Challenge3, or Fullsandwich from the Brain Bank.
4. When the brain has been loaded, press Esc to return to the MAIN MENU.
5. From the MAIN MENU, select Start Game. The game screen will appear.
6. You and *A.I.* take turns placing pieces on empty squares, just as in tic-tac-toe. The goal of the game is to complete a certain pattern. Each brain plays a game with a different winning pattern. The Challenge3 brain plays a game with *three* different patterns. If you want to try to figure out the patterns by watching *A.I.*'s strategy, skip to step 7: If you want to see the winning patterns, press the B key to do a "Brain Scan." You can do a Brain Scan anytime it's your turn.
7. On your turn, use the arrow keys to place the cursor where you want to move. Then press the Space Bar. Your piece will be placed, and Play will be highlighted on the menu. To tell *A.I.* to move, press Return. *A.I.* will move, then it will be your turn again. Repeat Step 7 to continue playing the game. *A.I.* will tell you when the game has ended.

Playing a New Game Against *A.I.*

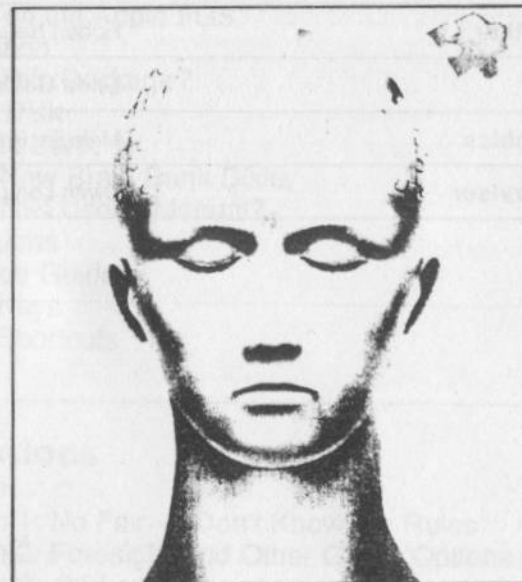
If you want to watch *A.I.* learn a new game, you must first invent the game. You do this by deciding on a winning pattern. The goal of the game will be to create that winning pattern. A winning pattern can consist of up to six different pieces, and it may be a row, column, diagonal, triangle, square, or any other shape that you can create on the game board. Your game may have as many as four different winning patterns.

To play a game that you've invented, follow this procedure:

1. Turn the computer off. Place the *A.I.* disk into the computer and turn it back on.
2. Select Start Game from the MAIN MENU.
3. You and *A.I.* take turns placing pieces on empty squares, just as in tic-tac-toe. The goal is to complete the winning pattern. On your turn, use the arrow keys to place the cursor on the square where you want to move. Then press the Space Bar. When you do so, your piece will be placed, and Play will be highlighted on the PLAY MENU. To tell *A.I.* to move, press Return. *A.I.* will move, then it will be your turn again.
4. Repeat Step 3 until you or *A.I.* plays a winning move. At that point, do not continue the game by pressing Return. Instead, use the arrow keys to indicate End Game on the menu, and then press Return. A new menu will appear. Choose one of the options from this menu to tell *A.I.* who has won the game.
5. Play the game as often as you like. The more times you play, the better *A.I.* will get at playing the game. You can press B to do a "Brain Scan" of *A.I.* anytime it's your move, to see how well *A.I.* is learning the game. If *A.I.* does not appear to be learning the game after you play 10 times or so, you may have chosen a type of game *A.I.* can't learn. Refer to The Games *A.I.* Can Play section in the Reference Guide for more information.

6. The knowledge *A.I.* gains about the game you play is stored in a brain. If you want *A.I.* to remember your game, you must store (save) the brain when you finish playing. To do so, press Esc until you return to the MAIN MENU. Then select the Brain Bank option from the menu. When the Brain Bank screen appears, select Save a Brain, and follow the instructions that will appear at the bottom of the screen.


To learn more about *A.I.*, go through the Explorations section of the User's Handbook. Then, if you have questions about any of *A.I.*'s features or capabilities, turn to the Reference Guide.



AN EXPERIENCE WITH ARTIFICIAL INTELLIGENCE


BY

CARY HAMMER

 **Scholastic**

Program Design and Programming	Cary Hammer
Publisher	Scholastic Software 730 Broadway New York, NY 10003
President	Richard Khaleel
Director of Technology	Walter J. Koetke
Director of Editorial Development	Philip H. Miller
Senior Project Editor	Robert Neumann
Handbook Author	Robert Neumann
Art Director	Linda Gottfried
Program Graphics	Malcolm Kirton
Testing Supervisor	Owen Long



 Scholastic Inc. New York Toronto London Auckland Sydney

Notice: This work is fully covered by the Copyright Law of the U.S. (Title 17 of the U.S. Code) and the Universal Copyright Convention. Unauthorized copying is strictly prohibited.

Software copyright © 1987 by Cary Hammer. All rights reserved. User's Handbook copyright © 1987 by Scholastic Inc. Printed in U.S.A. Published by Scholastic Inc.

ISBN: 0-590-98100-5 12 11 10 9 8 7 6 5 4 3 2 1 3 7 8 9 0 1 2 3

CONTENTS

Introduction	7
What Is Artificial Intelligence?	8
A.I. and Artificial Intelligence	9
Where Do You Go from Here?	11
About the Author	11
Getting Started	12
Equipment You Need	12
Loading A.I. Into the Computer	12
Using A.I. on the Apple IIGS	13
Sound Option	13
What's in This Package?	13
The A.I. Disk	13
The Brain Bank	13
Making New Brain Bank Disks	14
What's in This User's Manual?	14
Explorations	14
Reference Guide	15
Important Keys	16
Keypress Shortcuts	16
Icons	17
Explorations	18
Introduction	18
Exploration 1: No Fair—I Don't Know the Rules	21
Exploration 2: Foresight and Other Game Options	26
Exploration 3: A.I. Learns	31
Exploration 4: A.I. Learns Advanced Games	37
Exploration 5: Programming a Brain	41
Exploration 6: Using AutoRater to Activate A.I.'s Strategy	46
Exploration 7: Re-programming a Brain	49
Exploration 8: Programming a New Brain	52
Exploration 9: Programming Games That Include the Opponent's Pieces	56
Exploration 10: Advanced Programming	60

Reference Guide	70
START GAME	72
Play	73
Options	74
Foresight	74
Whose Turn	75
Learning	75
Grid Size	77
Brain Scan	77
Ratings	78
End Game	79
Take Back	80
Clear Board	80
Setup Board	80
PROGRAM A BRAIN	81
On-screen Tutorial	82
Set Goal #	82
Define Goal	82
Rate Moves	83
View Ratings	86
Grid Size	86
Auto Rater	86
CLEAR BRAIN	87
BRAIN BANK	88
Save a Brain	89
Load a Brain	90
Erase a Brain	90
Make a New Brain Bank	90
QUICKSTART GUIDE	91
Loading <i>A.I.</i> Into Your Computer	91
Playing a Game <i>A.I.</i> Already Knows	91
Playing a New Game Against <i>A.I.</i>	92

GAMES <i>A.I.</i> CAN PLAY	94
Winning Patterns	94
More Rules About Winning Patterns	96
<i>A.I.</i>'S ELECTRONIC BRAINS	98
HOW <i>A.I.</i> WORKS	100
Pattern Recognition: What Are the Winning Patterns?	101
What <i>A.I.</i> Does When a Game Has Ended	101
Truncation	102
Game Theory: Rating Moves Within Given Positions	102
Rules for Rating Moves	103
Categories of Moves	104
Negative Ratings	105
Board Evaluation Algorithm	106
<i>A.I.</i> 's Internal Rating Scale	108
Foresight	108
Ratings	109
BIBLIOGRAPHY	110



I NTRODUCTION

Suppose that you've created a board game and challenged a friend to play. But your friend is busy, and suggests you play her computer instead.

"But I just made this game up," you protest. "Your computer won't know the rules."

"Don't worry," your friend says. "It'll figure them out fast enough."

Ten minutes later you're playing the computer. You win easily several times running. "This chiphead's a chump," you call to your friend. But after the eighth or ninth game, the "chiphead" says:

"Sorry, I win."

You look at the board. "Caught me napping," you mutter. You continue to play. And you lose again. And soon you realize that not only has the computer figured out the rules of the game, but also an excellent strategy. The chump has become a champ. "It was an easy game, anyway," you tell your friend.

The next day you sit opposite the computer with a much more complicated game in mind. You've thought up *four* separate ways this game can be won. Once again you start the session by whipping your binary opponent. "This human's just too much for you, eh, chiphead?"

Then it happens.

"Sorry, I win," the computer says.

You examine the board and find that the computer has in fact achieved a winning position. "Clever," you admit. "Very clever." And you play again. . . and again . . . and again. But you can't deny what your eyes are telling you. The computer has figured out this game, too, despite all the different ways there are to win.

"It's an eerie feeling," you confess to your friend, "watching the computer figure out what's going on. It's like there's an intelligence inside."

"In a way there is," your friend says. "Check out the disk drive."

You pluck out a disk from the drive and read the name: *A.I., an Experience with Artificial Intelligence*.

"Artificial Intelligence," you repeat. "HMMMMMM..."

What is Artificial Intelligence?

Artificial intelligence is the branch of computer science concerned with teaching machines to do activities that we normally think require human intelligence. Playing games, as *A.I.* does, is an excellent example.

Researchers in different fields of artificial intelligence also teach computers to translate language, understand speech, summarize text, "see" patterns, or help professionals such as doctors and geologists make decisions.

When a machine or program is as successful at one of these human-like activities as *A.I.* is, it is said to have artificial intelligence. To date, no single machine or software program has come close to displaying *all* of the different types of artificial intelligence. Nor do researchers expect to see such machines in the near future. However, computers and robots displaying one or two types of artificial intelligence are already at work in offices, factories, and research laboratories. And many more "artificially intelligent" machines will join them in coming years, as scientists develop new software programs and faster, more powerful computer technology. (Detailed information about Artificial Intelligence can be found in the sources listed in the Bibliography at the end of the Reference Guide.)

A.I. and Artificial Intelligence

Your *A.I.* program draws from two fields of artificial intelligence. First, it uses pattern recognition techniques to compare and contrast winning positions until it finds what they have in common, thereby coming to "see" the goals of each game you make up. *A.I.* performs this compare and contrast procedure through techniques similar to those used by robots as they examine parts on an assembly line.

Once *A.I.* knows the winning patterns, it turns to a different field of artificial intelligence research: *game theory*. It uses game theory to devise strategies for reaching the winning patterns before you do.

You'll learn about these techniques and others not just as a witness to a demonstration, but as a hands-on researcher. That's how you'll feel when you master the basics of how *A.I.* works, and attempt to teach it your own game-playing strategies. Then you'll know what artificial intelligence is really all about—not a magic wand that turns a computer into a human, but an art and a science that attempts to make the most of what technology offers.

Along the way, you might also want to consider the sorts of questions scientists ask as they do artificial intelligence research. For example:

- What does it mean "to be intelligent?" What are the differences between artificial intelligence (at least as we now know it) and human intelligence?
- Is it possible for us to create programs that make computers smarter than we are?
- Do computers really "think," or do they just carry out instructions in a mindless way?
- Can a computer have imagination?
- Should we try to program computers to mimic the same processes that humans follow when they think? Or should we develop new strategies that are more appropriate to the way that computers work—as long as the result is the same?

As you begin your exploration of *A.I.*, here's one more question to consider: The *A.I.* program can usually beat you at your own games. Does that mean that *A.I.* is smarter than you?

Where Do You Go From Here?

Once you've read this Introduction, turn to the Getting Started section. That section will tell you what's in your *A.I.* package, what's in the different sections of this manual, and how you can get started using the *A.I.* program.

After you've read the Getting Started instructions, turn to the Explorations section of the manual. The explorations are step-by-step tutorials that show you how to use and learn from the *A.I.* program. Completing the explorations is the best way to learn how to use *A.I.*

After each exploration, take the time to experiment. If questions come up, turn to the Reference Guide section of this manual. The Reference Guide provides complete explanations of *A.I.*'s many features and functions.

About the Author

Cary Hammer is a freelance software designer who mainly develops educational products. He subsists mostly on bagels and Chinese food, and lives in the narrowest apartment in New York City.

He would like to give special thanks for their comments and suggestions to Christopher Chance, Mark Z. Malamud and Beatrice Bowers.

Comments about *A.I.* should be sent to the author at Scholastic Software, 730 Broadway, New York, New York 10003.



GETTING STARTED

This Getting Started section will acquaint you with your *A.I.* package and the different sections of this manual. Once you read the Getting Started section, you should introduce yourself to *A.I.* by completing the tutorials in the Explorations section.

Equipment You Need

- An Apple //e, IIc, or IIGS with 128K
- One disk drive

Loading *A.I.* Into the Computer

Follow these steps to load *A.I.* into your computer:

1. Insert the *A.I.* disk, label facing up and out, into the drive. Close the disk drive door.
2. Turn on the monitor or TV, and the computer. If the computer is already turned on, hold down and then release the Control, Open-Apple and Reset keys.
3. The disk drive will make some working noises as *A.I.* loads into the computer.

When the program finishes loading, the MAIN MENU will appear.

Using *A.I.* on the Apple IIGS

If you are running *A.I.* on the Apple IIGS, first use the Control Panel (Control, Option, Reset) to set the computer to fast mode. At this speed, *A.I.* will think and move more quickly than it does on other Apple II computers. The practical benefit of this increased speed is that you can tell *A.I.* to look further ahead when it plans its moves. See Foresight in the Reference Guide for more information about the program's "look ahead" feature.

Sound Option

You can turn sound on or off at any time by pressing Control-S.

What's in This Package?

Your package contains this user's manual and a double-sided *A.I.* disk.

The *A.I.* Disk

The *A.I.* disk is double-sided. The program is on Side 1, and a "Brain Bank" (see below) is on Side 2.

The Brain Bank

Side 2 of your disk contains a Brain Bank. The Brain Bank is a data disk that *A.I.* uses to store its knowledge about different games. You will use the Brain Bank when you do the tutorials in the Explorations section of this handbook.

The Brain Bank is not copy protected. If you have a two-drive system, copy the Brain Bank side onto another disk. Place the program disk in Drive 1 and the Brain Bank in Drive 2. If you have a one-drive system, *A.I.* will tell you when to switch disks.

Making New Brain Bank Disks

When *A.I.* learns a new game, you will often want it to save what it learns on your own Brain Bank disk. In Exploration 3, you will learn how to make Brain Bank disks. This information is also in the Reference Guide section titled, Make a New Brain Bank.

What's in This User's Manual?

This manual contains a series of 10 tutorials, called "explorations," that will introduce you to each facet of *A.I.*, and a Reference Guide that describes the program's features in detail. Both of these sections are described below.

Explorations

The Exploration section contains 10 explorations — or tutorials — that will introduce you to *A.I.* The first four explorations will teach you how to play games that *A.I.* already knows, and how to create your own games to see how well *A.I.* learns them. You'll learn about the various game features that regulate *A.I.*'s skills, and how to peer inside *A.I.*'s "brains" to monitor *A.I.*'s progress and emerging strategy. The final six explorations will show you how to program *A.I.*'s brains with your own strategies.

Each exploration contains simple examples that illustrate a different side of *A.I.* How deeply you explore each aspect of *A.I.* is up to you. Questions you may want to keep in mind include: What can *A.I.* learn? What can't it learn, and why? Can it learn better than you can? Why or why not? Can you program it to play games it can't learn by playing them? Can you improve upon *A.I.*'s game-playing strategies?

Even *A.I.*'s creators don't know everything it can do. Perhaps you will find a class of games *A.I.* can play that its creators didn't anticipate, or maybe you will develop playing strategies superior to the ones *A.I.* employs.

Reference Guide

When you want specific information about any of the game-playing options, programming features, or workings of the program, consult the Reference Guide.

The first part of the Reference Guide is divided into four sections that describe the four options listed on the program's MAIN MENU. You'll find information on what each game-playing option or feature does, and how and when to use it.

In the second half of the Reference Guide, you'll find information on the following:

- Games *A.I.* can learn.
- Games *A.I.* cannot learn, but that you can program it to play.
- The artificial intelligence techniques *A.I.* employs to recognize winning patterns and to devise game-playing strategies.
- The differences between the various types of brains used by *A.I.*

You'll also find a Quickstart Guide that contains procedures for loading the program, playing games *A.I.* already knows, and playing new games against *A.I.* All this information is contained in more detailed form in the Explorations, along with explanations of how *A.I.* works. Use the Quickstart Guide to refresh your memory, not to learn the program.

Important Keys

The prompts on the bottom of the screen will always tell you what options you have and which keys to press. In general, you will use the arrow keys, Space Bar, Return, and Esc keys in the manner described below.






Key	When To Use It
Arrow keys	Use the arrow keys to move the cursor around the game board, or to indicate menu selections.
Space Bar	Use the Space Bar to place a piece at the cursor. In Setup and programming modes, you also use the Space Bar to change a piece or remove it from the board.
Return	Use Return to select a menu choice. When you program a brain, use Return to indicate a move you wish to rate.
Esc	Use Esc to return to the previous menu, to switch to the menu when it is your turn during a game, or to finish adjusting a game option.
Control-S	Turn sound on or off.

Keypress Shortcuts

All game-playing options that can be changed or chosen from the menus can also be changed by keypress shortcuts. Simply press the key for the first letter of the option (F for Foresight, L for Learning, etc.)

Icons

The status of most game-playing options is indicated by icons:

Icon	What It Means
 	These are the playing pieces on the game board. The Sphere represents the player's move; the pyramid represents A.I.'s move.
	The Ratings icon indicates whether A.I. will display, in numerical form, its evaluation of each move it considers. When there is a slash across it, Ratings is <i>off</i> .
	The eye represents Foresight. The meter next to the "eye" indicates how many moves A.I. looks ahead on its turn. When Foresight is left at 0, A.I. does not look ahead at all. When Foresight is 1, A.I. looks at all possible replies to each move it considers. When Foresight is set to 2, A.I. considers all possible replies to those moves, too. Each time Foresight is raised one level, A.I. looks ahead one more move.
	The graduation hat indicates whether A.I.'s active brain is learning. It also appears on the Brain Bank screen, where it indicates whether the brain was programmed (<i>no</i> learning can take place) or learned (more learning <i>can</i> take place.)



EXPLORATIONS

This section includes 10 "explorations." Each exploration is a step-by-step tutorial that introduces you to a different facet of *A.I.* You'll begin by learning to play games *A.I.* already knows. Then you'll move on to creating your own games and challenging *A.I.* to learn them. Finally, if you want to delve more deeply into *A.I.*, you'll learn to modify *A.I.*'s game-playing strategies or provide your own.

When *A.I.* learns how to play a game, it stores its knowledge in an electronic "brain." Each brain can play one game. In each of the 10 explorations in this section, you will work with one or more brains. Some are on your Brain Bank (Side 2 of your *A.I.* disk). Others will be created by *A.I.* as it plays games against you during the explorations.

Explorations 1 to 4

In the first four explorations, you'll learn how to play games against *A.I.*

In Explorations 1 and 2, you'll play games *A.I.* already knows, and you'll try to guess the winning patterns. That will give you an idea of the complexity of the challenge that *A.I.* faces when it tries to learn games. You will also learn about the game-playing features you control. These include Foresight, which regulates how far ahead *A.I.* looks when planning a move; Ratings and Brain Scan, which offer you a look at how *A.I.* "thinks"; and other options that let you choose the size of the board, who goes first, and even the board position.

In Explorations 3 and 4, you'll create your own games and test *A.I.*'s learning capabilities. In these Explorations, you'll learn the difference between brains that were programmed to play games, and brains that learned through playing games. You'll also learn more about what sort of games *A.I.* can or cannot learn.

Explorations 5 to 10

The first four explorations demonstrated that *A.I.* can learn the goals of a game and then devise a strategy to suit those goals. In Explorations 5 to 10, you'll dig deeper into *A.I.*'s strategy-making mechanisms, and you'll have a chance to substitute your own strategies for *A.I.*'s.

But before you can teach strategy to *A.I.*, it must know the goal(s) for the game you're teaching. In Exploration 5, you'll learn how to sidestep the learning process and simply show, or "program," the goals into *A.I.* You'll play games against this programmed brain to see how *A.I.* plays when it knows the goals but has no strategy.

In Exploration 6, you'll tell *A.I.* to use its built-in strategies. It will work backwards from the goals to rate moves that might help it achieve those goals. In Exploration 7, you'll study *A.I.*'s strategies more closely. You'll make some changes to see how they affect *A.I.*'s playing skill. In Exploration 8, you'll learn to program *A.I.* with a strategy all your own.

In Exploration 9, you'll try to figure out why the standard strategies seem to fail for games whose goals include an opponent's piece or pieces. You'll also learn how to program these games. Finally, in Exploration 10, you'll learn some clever programming tricks. You'll learn new ways to use the goals, and in so doing, you'll also learn about a class of games *A.I.* cannot learn through competition, but that you can *program* it to play.

As you go through the explorations, you'll probably have questions about different features of the program. For answers to these questions, look in the appropriate sections of the Reference Guide.



EXPLORATION 1

A.I. can be taught an almost limitless number of games. It stores what it learns about each game in a different "electronic brain." Seven of these electronic brains are in the Brain Bank on Side 2 of your *A.I.* disk. In this exploration, you'll play one of them. Your mission is to figure out the goal of the game that this brain plays, and then to try to beat it at its own game.

Game Rules

You play games against *A.I.* on a board that looks like a tic-tac-toe grid—except that you can change the size of the grid. The grid that you see when you first play *A.I.* is six by six, the largest allowable size.

Here are the rules for making moves and winning games.

- Players alternate making moves.
- The player on the move must place a piece in an *empty* square.
- A game can have from one to four "goals," or winning patterns. The first player to achieve one of these goals is the winner.

An example of a winning goal is "four pieces in a row." The first player to place four pieces in a row wins the game.

As you try to figure out the goal of the game in this exploration, think about your own thought processes. What strategy are you using to deduce the goals? How would you tell someone else to go about it? Later, when you learn how *A.I.* figures out the goals of your games, you may want to compare notes. Do you and *A.I.* think alike?

Loading *A.I.* into the Computer

Place the *A.I.* disk into the disk drive, label up and out. If your computer is off, simply turn it on. If your computer is already on, hold down and then release the Control, Open-Apple and Reset keys at the same time. In a moment, the MAIN MENU should appear. Follow these steps whenever this manual asks you to load *A.I.* into the computer.



Figure 1. The MAIN MENU.

Loading a Brain

Before you can play a game, you must load a brain into your computer. Follow these steps:

1. The MAIN MENU should be on the screen. Select Brain Bank from the menu.
2. A Brain Bank screen appears.



Figure 2. The Brain Bank screen.

3. From the menu, select Load a Brain. The names of the brains on your Brain Bank disk will appear.

NOTE: If you have only one disk drive, or if you have two and have not placed a copy of the Brain Bank in the second drive, A.I. will ask you to insert a Brain Bank disk. Turn the program disk over and place it back in the drive, or insert a copy of the Brain Bank disk into either drive.

4. Use the arrow keys to select the brain named Challenge 1. Then press Return. *A.I.* will load the Challenge 1 brain.
5. When the disk drive stops whirring, press Esc to return to the MAIN MENU.

NOTE: Follow steps 1 through 5 whenever you want to begin play against an existing brain. Substitute the name of the brain you want to play for Challenge 1.

Starting a Game

Once you've loaded a brain, you're ready to play a game. Follow these instructions:

1. From the MAIN MENU, select Start Game. A.I.'s game board appears. Notice that the name of A.I.'s current brain is displayed across the electronic brain slot towards the bottom right of your screen.



Figure 3. The Game board.

A cursor is blinking on the board, telling you it's your move. Use the arrow keys to direct the cursor to the square where you want to move. Press the Space Bar to place a piece there. A sphere will appear. The sphere is the piece that marks your moves.

3. After you move, Play is highlighted on the menu. Now it's A.I.'s turn to place a piece on the board. Press Return to tell A.I. to play its move. After A.I. places its piece (a pyramid), the cursor will reappear. Now it is again your turn to move.

4. Repeat steps 2 and 3 until A.I. tells you the game is over. Study the final position. Can you guess the goal of this game? Press Return to start a new game.
5. After a few games, you may notice that the same pattern appears somewhere on the board each time A.I. says it has won. See if you can create that pattern with your own pieces as you play a game. If you do create the pattern and it is in fact the goal of the game, A.I. will tell you you've won. Congratulations! If you're wrong, you'll have to keep trying.

Brain Scan

There's an easy way to find out the goal of the game—just press the B key (for Brain Scan) whenever it's your turn. The game board will be replaced by four quadrants. One or more of the quadrants will display A.I.'s guess at the goal (or goals) of the game. A.I.'s pieces resemble small computer screens. The player's pieces look like someone's head and shoulders.

After you study the pattern, press Esc. The game board returns to the screen, and you may continue to play.

The Challenge1 brain plays a simple game. It was programmed to play the game at an intermediate skill level, not an expert one. In later explorations, you'll learn about more complex games, and you'll encounter brains that are virtually unbeatable at those games.



EXPLORATION 2

Before you turn the tables on *A.I.* to see how well it learns, you should explore some of *A.I.*'s game-playing features. In this exploration, you'll change some options to see how each affects the game. Some of the game options are on the same menu as the Play option. This is the PLAY MENU. Another list of options appears when you select Options from the PLAY MENU. This is the OPTIONS MENU.

Making Challenge 1 the Active Brain

For this exploration, like Exploration 1, you will need the Challenge1 brain in the computer as the "active brain." If your computer is on and you've just completed Exploration 1, then the Challenge1 brain is still the active brain. Press `Esc` until the MAIN MENU is on the screen.

If you're starting a new session, load *A.I.* into the computer. Then load the Challenge1 brain from the Brain Bank. (If you need help, see Exploration 1.)

Exploring the Options

With the Challenge1 brain in the computer, you're ready to explore some of the different game options.

1. Select Start Game from the MAIN MENU.
2. The game board appears. *A.I.* blinks the cursor, telling you it's your turn. But don't move yet. Instead, press `Esc` to change some game options. The cursor disappears and Play is highlighted on the menu. Use the arrow keys to highlight Options on the menu. Press `Return`. The OPTIONS MENU appears.

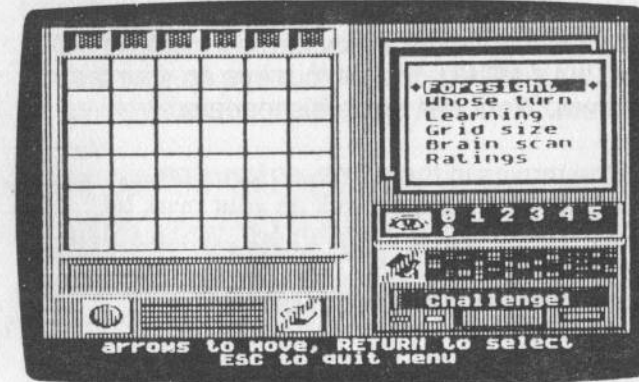


Figure 4. The OPTIONS MENU.

3. You can decide who will begin a game by selecting Whose Turn from the OPTIONS MENU. Do so now.

Look at the sphere on the bottom left of your screen. This sphere indicates it's your move. On *A.I.*'s turn, it becomes a pyramid. Press `Return` to watch it change. Press it again, and it changes back. During the game, it changes automatically after each player moves. Press `Return` to set it at the pyramid, so *A.I.* makes the game's first move.

NOTE: You can change whose move it is any time during the game.

4. Now select the Grid size option from the menu. Press any arrow key as often as necessary to adjust the size of the game board. The board can be any size from 3 x 3 to 6 x 6. When you finish, press Esc.

NOTE: You can only change Grid size at the start of a game.

5. We're finished with the OPTIONS MENU, so press Esc to go back to the PLAY MENU. (The Esc key always takes you back to where you were.) Make sure Play is highlighted, and press Return.

Take That Back!

Play two or three moves against *A.I.* After one of your moves, do not select Play. Instead, use the arrow keys to highlight Take Back from the PLAY MENU. Keep your eye on your last move, and press Return. Your last move disappears.

A.I. remembers all the moves in the game, so you can continue to take back moves, *A.I.*'s as well as your own, by pressing Return while Take Back is highlighted. When you're ready to continue the game, select Play. The game resumes with the player whose move was last taken back.

A.I. Looks Ahead

After your next move, do not select Play. Instead, select Options once again. From the OPTIONS MENU, select Foresight. Notice the meter in the middle right of the screen, next to the eye. The meter indicates the level of Foresight, or how far ahead *A.I.* looks when planning a move. When Foresight is set to 0, *A.I.* does not look ahead at all. When Foresight is raised to 1, *A.I.* analyzes all possible replies to each move it considers. Each time Foresight is raised another notch, *A.I.* analyzes one more move ahead.

The higher the Foresight, the better—but the more slowly—*A.I.* will play. Raising Foresight one notch slows *A.I.* down by a factor of six to 10.

Use the arrow keys to adjust the Foresight meter to 1 or 2, but no higher. For now, you want *A.I.* to move quickly. Once you have set Foresight to 1 or 2, press Esc. Now press Esc again to return to the PLAY MENU. Select Play from the menu and continue with the game.

NOTE: Foresight can be changed anytime.

As you play, you may notice two objects moving across the board while *A.I.* plans its move. The cursor shows the move *A.I.* is considering. The other object indicates the move *A.I.* considers best so far. When Foresight is set to 0, *A.I.* moves so quickly that it only displays the "best move" indicator.

Keypress Shortcuts

You may select any game-playing option on the PLAY MENU or the OPTIONS MENU by pressing the first letter of the option during your turn. For example, press T for Take Back, C for Clear Board, or O for the OPTIONS MENU itself.

On Your Own

Play several games to experiment with the different game options. (At this point, the End Game and Learning options won't work. You'll learn about them in the next exploration.) You may change game options by going through the menu in the way just described, or by pressing the key for the first letter of the option. For example, you can change the Whose Turn option by pressing **W**. Or you can change the Foresight option by pressing **F**, then adjusting the Foresight meter.

Try the following:

- Take back a move, then raise Foresight to see if *A.I.* plays differently.
- Select Setup Board from the PLAY MENU. Use the arrow keys and Space Bar to place pieces wherever you like. (The Space Bar is an all-purpose key. Use it to place your piece, to change your piece to your opponent's piece or the reverse, or to erase a piece.)

When you have set up the board, press Esc to finish. Then use the Whose Turn option to give *A.I.* the move. See what move *A.I.* makes.

Extra Challenges

Before you continue to the next exploration, play a different game with *A.I.* Press Esc until the MAIN MENU appears. Go into the Brain Bank and select Load a Brain. Choose, Challenge3, so named because the game it plays has three goals. (An example of a popular board game that has several goals is tic-tac-toe, which has four: a row, a column, and two diagonals.)

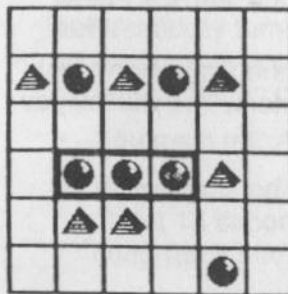
Can you figure out the goals for Challenge3? Hint: One of the goals includes an opponent's piece. You'll find that the Challenge3 brain plays a more complicated game than the Challenge1 brain. But it, too, was programmed to play at an intermediate, not an expert, level.



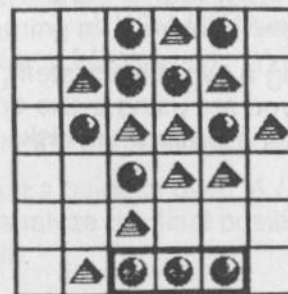
EXPLORATION 3

You've seen how tough it is to figure out the goals of a game. Now it's time to see how well *A.I.* does it. You'll start by teaching *A.I.* a simple game. (We're not trying to make life easier for *A.I.*—we just want you to learn the ropes. In Exploration 4, you'll challenge *A.I.* with more complicated games.)

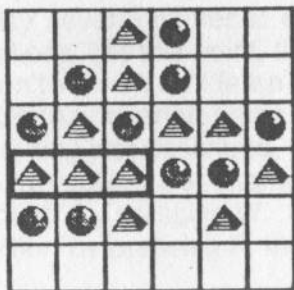
The goal of the game is to get a row of three pieces anywhere on the board. *A.I.* recognizes patterns, not exact board positions. This means that, once it understands that a row is a winning pattern, *A.I.* will consider a row anywhere on the board to be a winning pattern. To *A.I.*, then, all three positions shown below contain the same winning goal. In Figures 5 (a) and (b), *A.I.*'s opponent has won. In Figure 5 (c), *A.I.* has won.



(a)



(b)



(c)

Figure 5 Three in a row wins anywhere on the board.

Making a Brain Bank Disk

When you finish this exploration, you will save the *A.I.* brain that has learned the game you are about to play. To save this brain, you must first make a Brain Bank disk. You'll need a blank disk, or one whose contents you don't mind erasing.

1. You should be at the MAIN MENU. If you're not, press Esc until you are. (If you don't remember how to load *A.I.* into the computer, refer to Exploration 1.)
2. Choose Brain Bank from the MAIN MENU.
3. Choose Make a New Brain Bank from the BRAIN BANK MENU.
4. Follow the instructions that appear on the screen.
5. Once you have created the Brain Bank disk, press Esc to return to the MAIN MENU.

If you are using a two-drive system, place the Brain Bank disk in Drive 2. If you are using a one-drive system, *A.I.* will tell you when it needs the Brain Bank disk.

Blank Brains

A.I. must use a blank brain to learn a new game. When you load *A.I.* into the computer, it automatically selects a blank brain, so you can begin immediately to teach it a new game.

However, when you decide to teach *A.I.* a new game in the middle of a playing session, you must first tell *A.I.* to use a blank brain. This is true even if the brain's name, displayed across the electronic brain slot on the bottom right of your screen, is "unnamed." (Such a brain has undergone some learning, but has not yet been saved.)

Select Clear Brain from the MAIN MENU to tell *A.I.* to use a blank brain.

Learning

How good is *A.I.* at learning games it has never seen? Here's your chance to begin finding that out. Before you start, make sure that *A.I.* is using a blank brain, as described above. Then follow these instructions:

1. From the MAIN MENU, select Start Game.
2. Notice the graduation cap symbol on the bottom of the screen. This icon indicates whether "Learning" is on or off. When you played the Challenge1 brain in earlier explorations, the icon was slashed to indicate Learning was off. Learning was off because the Challenge1 brain already knew how to play a game. In this exploration, you want Learning on, so *A.I.* can learn a new game. *A.I.* has automatically turned Learning on for you. It knows to turn Learning on when you play a blank brain. When Learning is on, two things change:
 - You must tell *A.I.* when the game is over, and who won.
 - When you indicate that a game is over, *A.I.* will take about 15 seconds to analyze the final position and integrate it into its brain.

3. Play a game against *A.I.* Remember, the goal of this test game is to get three pieces in a row anywhere on the game board. When you make the winning move, do not select Play. Instead, select End Game from the menu. Another menu appears that offers three choices: *A.I.* Won, Player Won, or Tie Game. Choose one of these options to tell *A.I.* who won the game. (In this case, it will probably be you, the player.)

NOTE: If by mistake you select Play instead of End Game, use Take Back or Setup Board to restore the winning position. Then select End Game.

Begin a new game. Some time during the game, select Brain Scan from the OPTIONS MENU (or just press B). You will see *A.I.*'s first guess for the goal of the game.

4. Play several more games, telling *A.I.* who won after each. At first you will win easily, because *A.I.* doesn't know the goal of the game. But keep playing, and watch how *A.I.* gradually zeroes in on the goal. Whenever you like, peer inside *A.I.*'s electronic brain by selecting Brain Scan (or pressing B). Because *A.I.* assumes there may be as many as four goals, the brain scan may show more than one winning goal.

After each game, *A.I.* compares the final position with others of the same game, to refine its knowledge of the winning goals. To do this, *A.I.* uses simplified pattern recognition techniques commonly used to help computers and robots "see." (See the How *A.I.* Works section in the Reference Guide for more information.) It's as if *A.I.*'s "eyes" come a little more into focus after each game. If *A.I.* thinks it has found a new winning goal, it automatically develops a strategy to lead it toward the goal and to block the opponent from reaching the goal.

Ratings

The Ratings feature offers a different sort of look at *A.I.*'s insides. *A.I.* will display a number that rates each move it considers. To make its evaluations, *A.I.* uses techniques drawn from "game theory," another realm of artificial intelligence. (For more information about how *A.I.* evaluates moves, see the How *A.I.* Works section of the Reference Guide.)

1. To turn Ratings on, go to the OPTIONS MENU and highlight Ratings. Look for the icon of a gauge. The slash indicates that Ratings is off. Press Return, and the slash disappears. Ratings is now on. You can turn Ratings on and off by selecting Ratings, or by pressing R. For now, turn it on.

NOTE: You can turn Ratings on or off anytime during the game.

2. Press Esc to return to the PLAY MENU. Make sure Play is highlighted, and press Return. On *A.I.*'s turn, numbers will appear on the game board. These numbers indicate how *A.I.* has rated each move. The numbers are scaled from 1 to 99. The higher the number, the better the move — according to *A.I.*

NOTE: A.I. will play more slowly when Ratings is on. The difference is more noticeable at higher Foresight levels.

Saving a Brain

After you play for a while, you should save the brain for this game. Follow these instructions:

1. Return to the MAIN MENU by pressing Esc from the PLAY MENU.
2. Select Brain Bank from the MAIN MENU.

3. On the Brain Bank screen, select Save A Brain from the menu. A list of brains in the Brain Bank appears. Select a "cartridge slot" to store your brain. On a new Brain Bank disk, all the slots are blank. Press Return to select the first slot. If you use a Brain Bank disk that already has brains stored, use the arrow keys to select a blank slot. Press Return.
4. Type a name for your brain. Use a name that describes the game and how well A.I. has learned it. For example, if A.I. has not learned the game too well, you might name this brain RowNovice. "Row" tells you that the winning pattern is a row, and "Novice" tells you that the brain is not very experienced.

NOTE: It is a good idea to save a brain periodically as it learns a new game. This can prevent trouble if you accidentally declare the wrong winner of a game. If, for example, you tell A.I. that it won a game when in fact you did, it would become confused about the goals of the game. A.I. is capable of clearing up this confusion after many more games, but it's much quicker to select Clear Brain from the MAIN MENU, and then load in the most recently saved version of the brain you're playing.

Now that the brain is stored, you can return to the Brain Bank anytime and load it back into A.I. When you do so, the brain will pick up learning where it left off.



EXPLORATION 4

There is almost no limit to the number of games you can create with the A.I. program. You can play games with simple goals, such as a row, column, or diagonal. Or you can play games in which the goal is more complex, such as a cross, diamond, or triangle. However, the most interesting games are the ones in which the opponent's pieces are part of the goal, as shown in Figure 6.

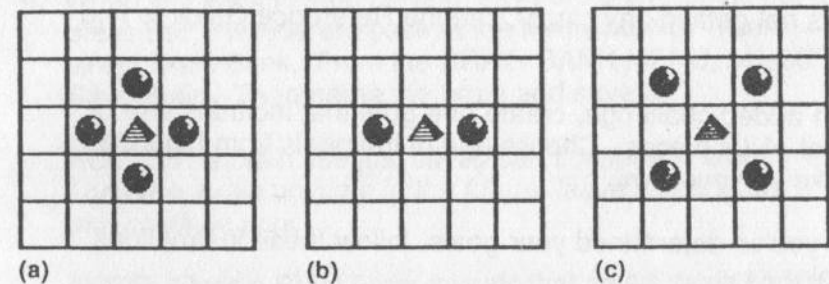


Figure 6. Goals that involve the position of a loser's piece as well as the winner's pieces.

A.I., like all computers, is very finicky, so you must be precise in defining the goal of your game. Look at the final positions in Figures 6(a) and (c). You might create a game saying to yourself, "The goal is to surround an opponent's piece." The patterns in Figure 6(a) and 6(c) satisfy your requirement. But A.I. sees these as two different goals. In one case, the encirclement takes place on the rows and columns of the surrounded piece. In the other case, it takes place on the diagonals.

Remember, also, that A.I. sees patterns, not board positions. If you intend a goal to win only on one part of the board, you're out of luck. A.I.'s motto is: If a pattern wins *somewhere* on the board, it wins *everywhere*.

Playing a Game with Two or More Goals

In this exploration, you'll create a game with at least two goals. Before you choose the goals, it is recommended that you read the Winning Patterns section in the Reference Guide. In that section, you'll learn how A.I. defines a goal, which goals it can learn, and which it can't. Of course, you can also plunge right into this exploration. But if A.I. does not learn your game after 10 or 15 plays, and if Brain Scans suggest that it's not getting any closer, see the Reference Guide to find out why.

For an added challenge, create one goal that includes one or more of A.I.'s pieces. Choose one of the goals from Figure 6, or make up your own.

Once you've determined your goals, follow these instructions to begin:

1. If A.I. is not currently using a blank brain, select Clear Brain from the MAIN MENU.

2. Choose a grid size and then begin play. For the first few games, leave Foresight at 0. This is the fastest setting, and A.I. won't know enough about the new game to make much use of a higher Foresight, anyway. Use Brain Scan whenever you like, and keep Ratings on. When A.I. starts to catch on, raise Foresight to 1 or 2 to see how well A.I. plays. You may raise Foresight even higher towards the end of a game, when A.I. has fewer empty squares to consider, allowing it to move more quickly.
3. Play as many games as you like. Remember to tell A.I. when a game is over. If you continue play past that point, use Take Back or Setup Board to restore the position as it was just after the winning move was played.

NOTE: If you or A.I. make a move which wins for both sides, declare the game a tie.

A.I. will probably take longer to learn this game than the first one you created. That's because it has to learn two different goals. Also, goals that require combinations of pieces from both players are harder for A.I. to learn than goals requiring only the winner's pieces.

4. When you want to stop playing, save the brain. To do so, press Esc until you are back at the MAIN MENU. Then select Brain Bank. From the BRAIN BANK MENU, select Save Brain. Then name the brain and save it.

NOTE: The brain you just saved can continue to learn. Load the brain anytime in the future, leave learning on, and continue play.

NOTE: Once a Brain Scan reveals that an A.I. brain knows the goals of a game, it is a good idea to turn Learning off. Then you won't have to wait for A.I. to study the final position after each game.

About the Brain Bank

Here is some important information about Brain Banks:

- Each Brain Bank disk can hold 16 brains. If you want to save more, create a new Brain Bank disk by choosing MAKE A NEW BRAIN from the BRAIN BANK MENU.
- The name of a brain cannot exceed 13 letters.
- The name of a brain should tell you what game it plays, how well it plays, and perhaps even who played against it. For example, TTTfairJOE might mean the game of tic-tac-toe at a fair skill level, played against JOE. BOXGreatSUE might mean the game of BOX at a high skill level, played by Sue.
- If you select Load a Brain from the Brain Bank, you will see the Learning icon. As you use the arrow keys to highlight the names of different brains, the Learning icon indicates whether that brain can learn or not. When the icon is slashed, that brain *cannot* learn. When the icon is not slashed, the brain *can* learn. The brains you saved from Explorations 3 and 4 can learn. But the Challenge brains cannot. These brains did not learn by actually playing, but by being programmed. You'll learn about these "programmed" brains in later explorations.



EXPLORATION 5

In this exploration, you'll learn to program winning goals directly into an *A.I.* brain. This will make life easier for this electronic brain when you play against it. The brain will *know* the goals, rather than having to guess at them as you play the game.

The MAIN MENU should be on the screen. If it isn't, press Esc until it appears. The active brain should be empty. If it is not, select Clear Brain from the MAIN MENU.

On-Screen Tutorial

You can ask *A.I.* to briefly demonstrate the programming process. To do so, press H when the programming screen appears after Step 1 in the section below.

Showing *A.I.* the Goals of a Game

Here is how you show *A.I.* the goals of a game, using tic-tac-toe as an example. (If you prefer, you can create your own game that has at least two winning patterns. However, the figures and instructions in this exploration refer to tic-tac-toe.)

1. From the MAIN MENU, select Program Brain. The Programming screen appears. You may recognize the Goals display from doing Brain Scans.

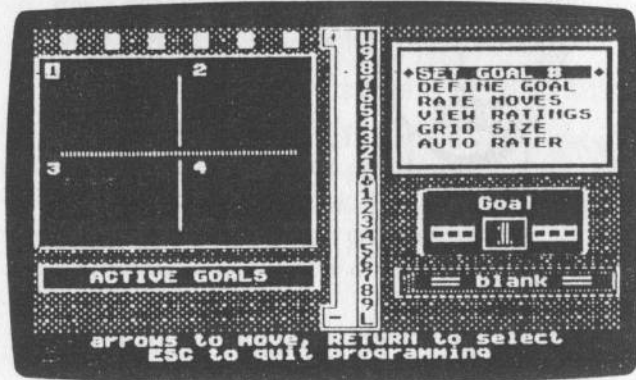


Figure 7. The Programming screen.

2. Adjust the grid size, if necessary, by selecting the Grid Size option from the PROGRAMMING MENU. If you're going to teach *A.I.* how to play tic-tac-toe, set the grid size to 3 x 3.
3. Make sure the goal indicator is set to 1. If it is, continue with step 4. If it is not, select Set Goal # from the PROGRAMMING MENU, and use the arrow keys to set the goal indicator to 1. Then press Esc.
4. Choose Define Goal from the menu.
5. Select one of the four tic-tac-toe winning positions that you will program into *A.I.* All four winning positions for tic-tac-toe are shown below.

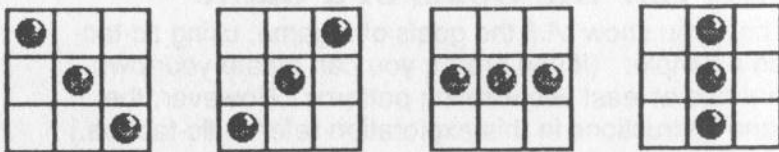


Figure 8. Winning Patterns in tic-tac-toe.

Use the arrow keys to move the cursor, and press the Space Bar to place pieces. The pieces you place will be larger versions of the computer pieces you've seen in Brain Scans.

Notice that if you press the Space Bar when the cursor is on a square occupied by a computer piece, that piece becomes a player's piece. If you press the Space Bar again, the player's piece will disappear. In this way, you can show *A.I.* a goal using only the arrow keys and the Space Bar.

Make sure that the pieces you place are computer pieces! (If you placed player pieces instead, *A.I.* would think that one goal of the game is to force the opponent into achieving that pattern.) When you finish, press Esc.

The goal should appear in the first quadrant of the Goals display. Don't be surprised if it appears in a different part of the quadrant than you expected. For example, a row may appear on top, even though you programmed it in the middle. Remember that *A.I.* recognizes patterns, not positions.

6. Select the Set Goal # option from the PROGRAMMING MENU. Then use the arrow keys to set the goal indicator to 2. Press Esc once it's set.
7. Select Define Goal, and place the pieces for the second goal of tic-tac-toe.
8. Now show *A.I.* goals 3 and 4. First select Set Goal# to move the goal indicator to the next goal. Then select Define Goal to show *A.I.* the goal. If your game has fewer than four goals, stop after the last goal.

Here is how the finished screen should look once you've programmed A.I. to play tic-tac-toe.

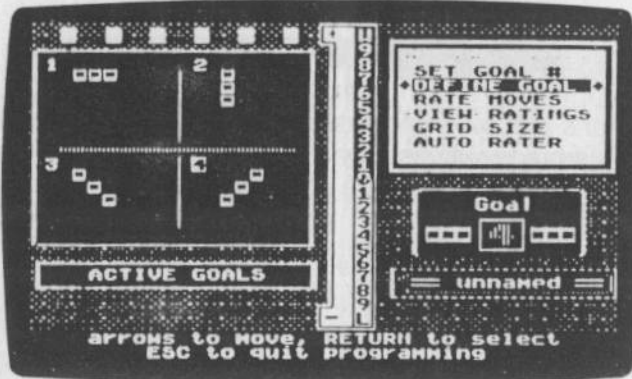


Figure 9. Tic-tac-toe goals programmed into A.I.

Programming goals into A.I. is not as exciting as watching A.I. learn them through play. But it's a handy shortcut when you want to work with another aspect of A.I.'s artificial intelligence—its strategy mechanisms.

Normally, you would give A.I. a strategy as soon as you show it the goals. You'll do that in the next exploration. For now, why not take advantage of the fact that A.I. has no strategy and beat it a few times? To do this, return to the MAIN MENU by pressing Esc. Then select Start Game from the menu. Set Foresight to 0, and play some games.

Without a strategy and at Foresight 0, A.I. plays blindly. If you turn Ratings on, you will see that A.I. can't tell the difference between moves, except for the winning move. Raise Foresight to 1 or 2 and let A.I. move first, and it will play slightly better. It still won't have a strategy, but it will be able to "see" further ahead towards a goal.

NOTE: Learning is off, so A.I. is in control and will tell you when the game is over. A.I. won't let you turn Learning on. The brain you're playing against has been programmed, and programmed brains cannot learn from playing.

Saving the Brain

Save this brain. You'll use it in the next exploration, so name it Explorer 6. If you don't remember how to save a brain, see the instructions in Exploration 3.



EXPLORATION 6

For *A.I.* to play a game well, it is not enough for it to know the goals. It must also have a strategy.

When *A.I.* learns a game by playing it against you, it develops a new strategy each time it studies a final position and finds what it thinks is a winning goal. In this exploration, you'll learn how you can tell *A.I.* to do that for goals that you have programmed for it. You'll be using the Auto Rater feature to have *A.I.* set its strategy automatically. In the next exploration, you'll set the strategy yourself.

Using Auto Rater

Here's how you use the Auto Rater option on the Program Brain screen to have *A.I.* automatically rate different moves.

1. Load in the brain you saved in Exploration 5. Select Program Brain from the MAIN MENU. Make sure the Goal indicator is set to 1.
2. Select Auto Rater from the PROGRAMMING MENU on the Program Brain screen. *A.I.* will devise a strategy for Goal 1 by rating various moves it can make in different situations. When *A.I.* plays this game against you, it will use these ratings to decide which move to make.

Seeing the Ratings *A.I.* Created

To see the ratings that *A.I.* created, follow these instructions:

1. Select View Ratings. The arrow will go all the way to the W at the top of the ratings scale. Press Return to see the ratings.
2. *A.I.* displays "formations" of pieces one at a time. Each formation is in the shape of the goal. In each formation, one computer piece is white (or unshaded on monochrome monitors.) This piece represents the move whose rating appears on the rating scale alongside the board. The rating indicates the value *A.I.* has placed on making that move when the other pieces are in the formation shown. In the figure below, for example, placing the third piece in a diagonal is a W for Win, the highest rating. Placing a second piece is worth an 8 on the ratings scale.

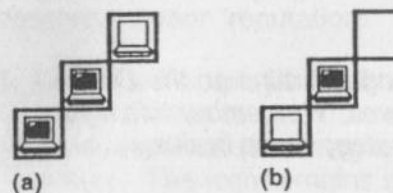


Figure 10: (a) The move is rated W, for win. (b) The move is rated 8.

For a description of why *A.I.* rates moves only as they relate to the shape of the goals, and for information on how *A.I.* uses the ratings to decide on a move, refer to the How *A.I.* Works section in the Reference Guide.

3. Continue to look through the ratings. Notice that different moves often give rise to the same position, as shown in Figure 11. Since *A.I.* rates moves, not positions, you'll find separate ratings for *all* of these moves.

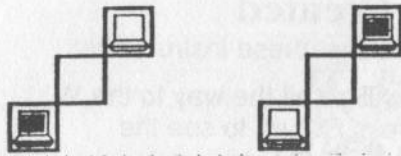


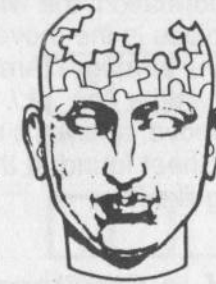
Figure 11. Both moves are rated 8.

After you look at the ratings, tell *A.I.* to develop a strategy for the other winning patterns. First select Set Goal # to change the Goal indicator. Then select Auto Rater.

When you finish, play a few games against this brain. You'll find that it plays very well indeed, even if Foresight is set to 0. For certain games played on small boards, such as tic-tac-toe, it will probably be unbeatable.

Before you proceed to the next exploration, you may want to create some more games and use Auto Rater to see how well *A.I.* plays them.

NOTE: The Challenge1 and Challenge3 brains on the Brain Bank disk were altered to play less well. Remember this if you use View Ratings to examine the strategy those brains employ.



EXPLORATION 7

In this exploration, you'll tinker with *A.I.*'s strategies to change the way a brain plays. You'll work on a brain that was poorly programmed, changing the ratings of moves that you think deserve a better "reputation."

1. Load the Poorplay brain from the Brain Bank. As you press the arrow key to step from one slot to the next, watch the graduation hat (Learning icon) between the stacks of brains. The icon remains slashed, indicating that the brain was programmed, and no more learning can take place. All the brains on your Brain Bank disk were programmed.
2. The Poorplay brain plays a game in which the winning pattern is a row of three pieces—but it doesn't play it very well. Play a few games against this brain. You'll find that the brain makes some smart moves, but also some very dumb moves. Can you figure out what's gone wrong with its strategy?
3. Select Program Brain from the MAIN MENU.
4. Select View Ratings.

- Look through the ratings *A.I.* has assigned various moves. Remember that in each formation, the white (unshaded in monochrome monitors) piece is the move *A.I.* rated. Can you see a problem with the strategy? Are there moves that deserve a better rating than the one *A.I.* gave? If so, write down the formation, the move, and *A.I.*'s rating. Use the Programming Template Sheet found at the end of this exploration, or something like it.

Tinkering with *A.I.*'s Strategy

Here are the steps you should take to adjust the strategy the Poorplay brain is using.

- Choose Rate Moves from the menu on the Programming screen. For each move whose rating you want to change, follow this procedure:
 - Set up the formation of pieces. Use the arrow keys to show where a piece belongs, and press the Space Bar to place, change, or remove a piece. Do *not* place the move you want to rate.
 - When the other pieces are in place, indicate the move you will rate by placing the cursor over the square and pressing Return. Do *not* use negative ratings. You will learn when to use negative ratings in Exploration 9.
 - Use the arrow keys to adjust the move's rating on the ratings scale. When you finish, press Return.

Repeat steps A through C until you have made all necessary changes.

- When you finish, play some games against this reprogrammed brain. Does it play better? If not, try some more reprogramming.

The How *A.I.* Works section in the Reference Guide describes the general procedure *A.I.* uses to rate moves for each game. But before you read it, you may want to continue along to Exploration 8, to see how well you can program a brand new brain. Maybe you'll come up with strategies similar to *A.I.*'s.

Programming Template Sheet

--	--	--

Rating:

--	--	--

Rating:

--	--	--

Rating:

--	--	--

Rating:

--	--	--

Rating:

--	--	--

Rating:

--	--	--

Rating:

--	--	--

Rating:

--	--	--

Rating:

--	--	--

Rating:

--	--	--

Rating:

--	--	--

Rating:

--	--	--

Rating:

--	--	--

Rating:

--	--	--

Rating:

--	--	--

Rating:

--	--	--

Rating:

--	--	--

Rating:

--	--	--

Rating:

--	--	--

Rating:

--	--	--

Rating:

--	--	--

Rating:

--	--	--

Rating:

--	--	--

Rating:

--	--	--

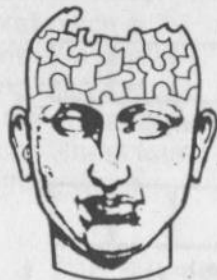
Rating:

--	--	--

Rating:

--	--	--

Rating:



EXPLORATION 8

In this exploration, you'll teach *A.I.* a new game called Triangle. First you'll show it the goal. Then you'll program your own strategy. The goal is simple: to make a triangle such as the one below.

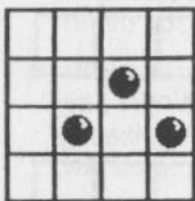
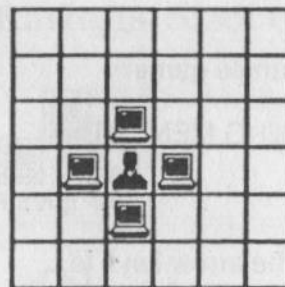
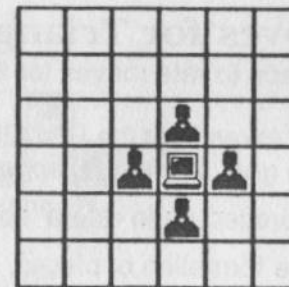


Figure 12. The winning goal for Triangle.

When you show *A.I.* a goal, *A.I.*'s pieces must be the winner's pieces. To see why, look at Figure 13.



(a)



(b)

Figure 13: a) The goal is to trap an opponent's piece. b) The goal is to be trapped by the opponent.

The pattern at the left tells *A.I.* that the goal is to trap the opponent's piece. But the pattern at the right tells *A.I.* that the goal is to be trapped by the *opponent*. That's why you must be careful to use *A.I.*'s pieces as the winning pieces.

Ratings Sheet

At the end of this exploration, you'll find the ratings that *A.I.* would assign if you asked it to develop its own strategy for Triangle. Enter all these formations and ratings into *A.I.*, just to get the feel of programming a game. If you prefer, create your own game and try to program it. Whether you use Triangle or create your own game, remember that *A.I.* rates moves, not patterns. For most games, you must rate *all* moves that give rise to a given pattern, not just one of them.

Clear Brain

To start this exploration, select Clear Brain from the MAIN MENU. Then select Program Brain from the MAIN MENU.

Showing *A.I.* the Goal of Triangle

Select Define Goal from the PROGRAMMING MENU on the Program Brain screen. A game board appears. Place the triangle pattern (or your own goal) anywhere on the board. Use arrow keys to move the cursor, and press Space Bar to place, change, or erase a piece. When you finish, press Esc.

Rating Moves for Triangle

Follow these steps to rate moves for the Triangle game:

1. Select Rate Moves from the PROGRAMMING MENU. The shape of the goal, a triangle, appears.
2. Follow this procedure to rate a move:
 - A. Set up the formation of pieces. Use the arrow keys to show where a piece belongs, and press the Space Bar to place, change, or remove a piece. Do *not* place the move you want to rate.
 - B. When the other pieces are in place, indicate the move you will rate by placing the cursor over the square and pressing Return.
 - C. Use the arrow keys to rate the move on the ratings scale. When you finish, press Return.

Repeat steps A through C until you have rated all the moves.

Note that all moves are automatically rated 0 until you change them. Therefore, moves you do not rate are actually rated 0.

Note, also, that *A.I.* only rates moves that *A.I.* itself can make, but it applies those ratings to its opponent's moves, too. For example, if you rate a move *A.I.* can make as a 9, it also considers that move to be a 9 when made by its opponent. In this way, *A.I.* knows it has lost a game when its opponent makes a move that it has rated a W.

Congratulations! You've just programmed a brain. If you like, play some games against this brain.

On Your Own

You should now take some time to design, program, and play some of your own games. In Exploration 9, you'll learn to program games whose goals include your opponent's pieces as well as your own. If you wish, try to program such a game now, before you go on to see how *A.I.* handles it.

Ratings Sheet



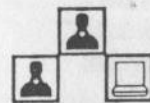
Rating: W



Rating: W



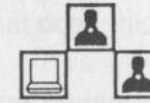
Rating: W



Rating: 9



Rating: 9



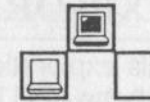
Rating: 9



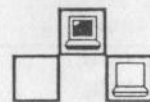
Rating: 8



Rating: 8



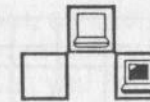
Rating: 8



Rating: 8



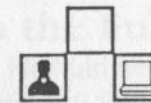
Rating: 8



Rating: 8



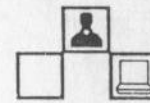
Rating: 7



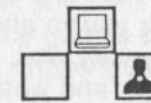
Rating: 7



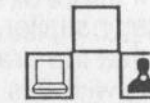
Rating: 7



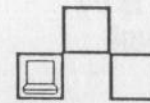
Rating: 7



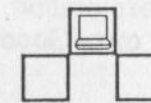
Rating: 7



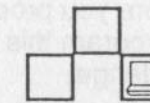
Rating: 7



Rating: 6



Rating: 6



Rating: 6



EXPLORATION 9

In this exploration, you'll look at the strategy for a game in which the goal includes an opponent's piece. The game is Sandwich. The goal is shown below.

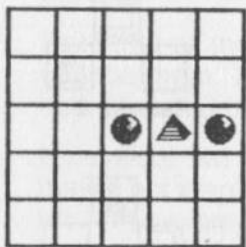


Figure 14. The goal is to place a piece to each side of an opponent's piece.

You'll play this game against a brain programmed with a strategy similar to other brains you've encountered. But you'll find that the brain doesn't play very well. You'll learn what's wrong with this brain's strategy, and what you have to do to improve it.

Before you proceed with this exploration, you may want to try to program this game on your own. Good luck—it's a tough challenge.

What's Wrong with the Halfsandwich Brain?

Follow these instructions to figure out what's wrong with the brain called Halfsandwich.

1. Go to the Brain Bank and load the Halfsandwich brain into A.I. Play some games against it with Foresight set at 0. You'll find it easy to win. In fact, you'll find that A.I. stumbles right into your winning patterns. What does this tell you about its programming?
2. Play again with Foresight set to 1. Now A.I. may be more careful about making losing moves. But if A.I. does place a piece between two of yours, you'll see that it doesn't know the game has ended. What else does this tell you about its programming?
3. Select Program a Brain from the MAIN MENU. Then select View Ratings from the PROGRAMMING MENU. The ratings should look familiar. A "W" rating is awarded to winning moves, a 9 to moves that prevent a win, an 8 to moves that bring the player one move away from victory, and so on. Continue looking at the moves. Can you figure out why A.I. plays so poorly?

Stepping Up to the Fullsandwich Brain

You've played against Halfsandwich, a poorly programmed game. Now let's play a brain that plays Sandwich much better. This brain is called Fullsandwich.

1. Go to the Brain Bank and load Fullsandwich into the computer. Play some games against it at Foresight 0. You should get a strong challenge.

- Select Program a Brain from the MAIN MENU. Then choose View Ratings from the PROGRAMMING MENU. The ratings from W to 0 will be identical to those you saw in Halfsandwich. Check them if you like. But notice that there are further ratings beginning at -5 and continuing through L. This may be the first time you've seen these "negative ratings." Games in which an opponent's piece is part of the goal must have negative ratings as well as positive ratings.

Here's why. Remember that ratings are awarded for any move that may be made within the shape of the goal. When a goal includes only your own pieces, any move you make within this goal shape either helps you towards the goal, prevents the opponent from achieving it, or, at worst, accomplishes nothing. See Figure 15, below.

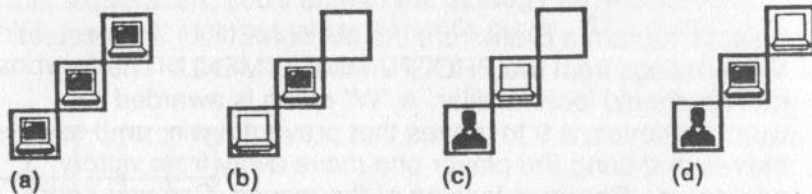


Figure 15. (a) A goal. (b) A move that leads towards the goal. (c) A move that blocks the opponent from achieving the goal. (d) A pointless move which neither blocks the opponent nor advances towards the goal.

However, in games where the opponent's pieces are included in the goal, new types of moves are possible, as shown in Figure 16 below. You can make moves that lose instantly, and you can make moves that spoil your chances of winning. *A.I.* regards these moves as "actively bad," and rates them with negative numbers. More exact descriptions of how *A.I.* does this may be found in the How *A.I.* Works section of the Reference Guide.

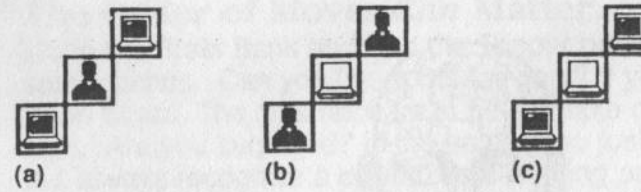


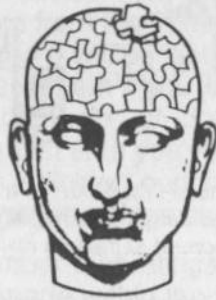
Figure 16: (a) The goal. (b) A move that deserves a negative rating because it loses instantly. (c) A move that deserves a negative rating because it spoils an opportunity for the opponent to make a losing move.

- Games with goals that include your opponent's pieces as well as your own are the most challenging. Try programming your own. Refer to the ratings used by the Fullsandwich brain for clues as to how to rate certain moves in your game.

On Your Own

You've now learned almost everything you need to know to experiment with *A.I.* on your own. You can experiment to see what games *A.I.* can learn, how quickly it learns, and how well it will play. You've learned how to show *A.I.* the goals of a game, and how to have it generate its own strategy. You've also learned how to program your own strategy. And you've learned how to do all of this for a variety of games.

Before you continue to Exploration 10, take time to invent, play, and program several games with goals that include an opponent's piece. These are the most complicated games and involve the most interesting strategies.



EXPLORATION 10

In this section, you will learn two advanced programming techniques. You will learn to program *A.I.* to play games that it cannot learn through over-the-board play with an opponent. These games include:

- Games whose goals include empty squares.
- Games in which the order of moves is important in achieving the goals.
- Games with more than four goals.

You will also learn a new way to program strategies into *A.I.* In both cases—programming new games and programming new strategies—you will be taking advantage of the fact that goals and winning patterns need not be the same.

You'll start by exploring Topper, one of the brains on the Brain Bank disk.

The Order of Moves *Can* Matter

Go to the Brain Bank and load the Topper brain. Then play some games. Can you figure out the goal? If you can't, do a Brain Scan. The goal appears to be a simple column three high. Are you surprised? In the games you just played, did *A.I.* always recognize a column as a winning pattern? Probably not. To find out why, choose Program Brain from the MAIN MENU. Then select the View Ratings option from the PROGRAMMING MENU.

Notice which moves are rated W, and which aren't. Only one move is rated W—the move that places the top piece in the column. Placing the piece in the middle or at bottom is rated a 0. This tells you that the column is a winning pattern *only* when the top piece is the final one played.

A.I. was programmed to play Topper. When *A.I.* examines final positions, it looks only at the pattern, not the sequence of moves. Consequently, it cannot learn Topper through actual play, because the pattern itself does not completely describe the winning goal.

NOTE: A.I. does remember the final move of each game, but for a different purpose. See the How A.I. Works section of the Reference Guide.

How an *A.I.* Brain was "Doctored" to Play Topper

When *A.I.* finds a winning goal through over-the-board play, or when you program a goal, *A.I.* immediately assigns a W to all moves that achieve this goal. But the programmer of the Topper brain cleverly used the Rate Moves feature to remove the W rating from moves that completed a column in the middle or at bottom. This maneuver helped *A.I.* play a game it could not learn through play. The programmer continued to use the Rate Moves feature to rate moves in accordance with the single winning pattern. The programmer could not use the Auto Rater feature, which would restore the W ratings and rate moves in the standard manner.

Continue using View Ratings to see how the programmer of this game rated various moves.

Goals *Can* Include Empty Squares

Load the Space brain from your Brain Bank disk. This brain demonstrates another class of games *A.I.* can be programmed to play, but cannot learn—games in which empty squares are part of the goal. Play a few games. Can you figure out the winning pattern? Hint: it involves a triangle.

After you play some games, do a Brain Scan. The winning pattern appears to consist of a triangle. But you probably found that *A.I.* did not recognize this four-piece triangle as a winning pattern. To see why, go into the programming mode and select View Ratings.

Look at the moves rated W. In each case, the square below the top of the triangle is empty. In effect, that means that the winning goal is a three-piece triangle in which the square below the top piece *must* be empty. (See Figure 17.) Look through the ratings and you will see that moves into that square are considered very bad, unless they are blocking moves.

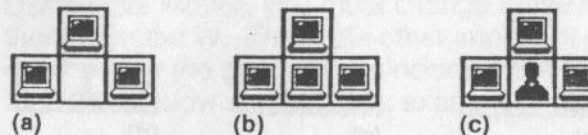


Figure 17: (a) A winning goal. The square below the top of the triangle must be empty. (b) and (c) These patterns do not win, because the square is occupied.

If you attempted to have *A.I.* learn this game through play, it would learn the importance of the triangle, but it would assume that *anything* can be in the space below the top of the triangle. *A.I.* would make the same assumption if, in programming mode, you defined the goal as including only the three squares of the triangle, as in Figure 18 below. Whether you then chose Auto Rater or Rate Moves, *A.I.* would display a goal shape with only the three squares. There would be no way to show that the fourth square must be empty.



Figure 18. The winning goal for *Space* cannot be defined with only three squares, as shown here, because *A.I.* would have no way of knowing the importance of the empty square.

In Figure 19 below, the ratings shown hold regardless of what's in the square below the top of the triangle.

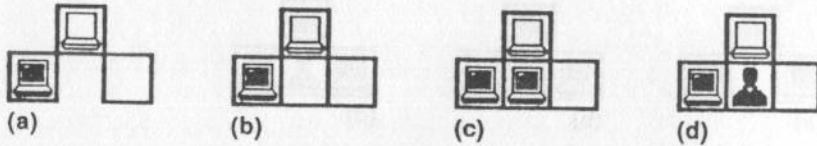


Figure 19: (a) For the goal shown in Figure 18, Auto Rater would rate this move an 8. In *Space*, however, the move only merits an 8 in (b), where the bottom square is empty. In (c) and (d) it is occupied, so the move does not deserve an 8.

Defining an Empty Square

To have *A.I.* recognize the importance of the square below the top of the triangle, you must include it in the goal. Choose Define Goal from the PROGRAMMING MENU and place four pieces; three in the triangle and one where the empty square should be. The extra piece can be either a computer piece or a player piece.

A.I. will assign a "W" to all moves that bring about this position, when in fact none are winning moves. That's because the goal must have an empty square below the top of the triangle. See Figure 20 below.

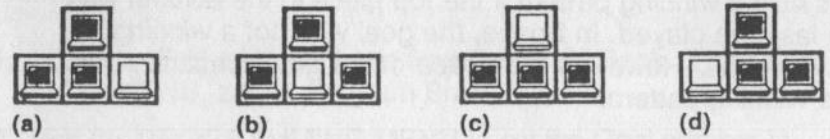


Figure 20. *A.I.* rates these moves W, because it thinks that the *Space* goal includes pieces in all four squares. These ratings must be changed.

Using Rate Moves, you must change these ratings—"unhook" them from the W. Then rate other important moves that can occur within the goal shape, including the true winning moves. Figure 21 below shows some examples.

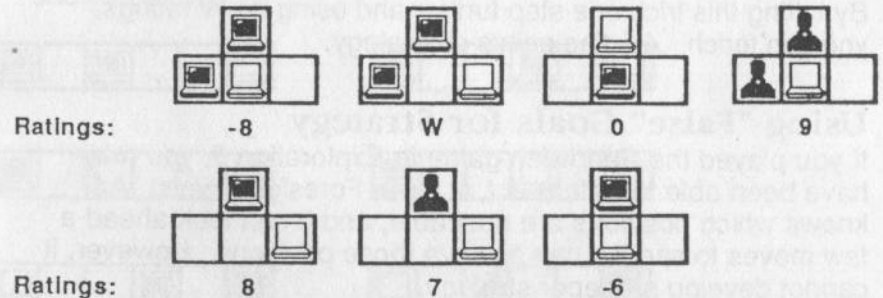


Figure 21. The rating for each move appears below the formation. Negative ratings are assigned to moves which worsen the position.

Goals vs. Winning Patterns

Topper and Space show that moves that complete a goal do not necessarily win. You will understand *A.I.* better if you realize that a goal is just a convenient shape that can be used to teach *A.I.* winning patterns.

In earlier explorations, the goal and the winning pattern were the same. That's how *A.I.* thinks when it learns a game through play, and that's the easiest way to start programming *A.I.* But by playing Topper and Space, you've seen that the goals can be used more creatively, as a way to teach *A.I.* games it could never learn through play. In Topper, the goal was only a winning pattern if the top piece in the column was the last one played. In Space, the goal was not a winning pattern at all. However, the *shape* of the goal contained the true winning pattern.

Because *A.I.* automatically assigns *W* ratings to all moves that complete a goal, you have to change these ratings when you use goals this way. In Topper, the *W* ratings were removed from moves that completed a column in the middle or at the bottom. In Space, the *W* was assigned only for moves that completed the triangle when the space below the top was blank.

By taking this trick one step further and using no *W* ratings, you can teach *A.I.* fine points of strategy.

Using "False" Goals for Strategy

If you played the Sandwich game in Exploration 9, you may have been able to defeat *A.I.* at lower Foresight levels. *A.I.* knows which positions are desirable, and it can look ahead a few moves to see if it can achieve those positions. However, it cannot develop a deeper strategy.

When playing Sandwich, suppose you decide it is desirable to place three pieces in each row: one on each side of the board, and one separated by at least one square from each end piece, as in Figure 22 below. In such a position, any opponent's move onto that row will create a win for you. You can teach this strategy to *A.I.* by setting up a "false" goal.



Figure 22. A desirable formation for the Sandwich game.

Here is the procedure for setting up a "false" goal, using the Sandwich strategy as an example:

1. Load the Fullsandwich brain from the Brain Bank. From the MAIN MENU, select Program Brain.
2. Use the PROGRAMMING MENU to set the Goal number to 2, and select Define Goal.
3. Place six pieces in a row.
4. Remember that *A.I.* automatically assigns a *W* to every move that brings about a goal. Use the Rate Moves option to set those ratings to 0.
5. Continue using the Rate Moves option to rate the moves shown below.

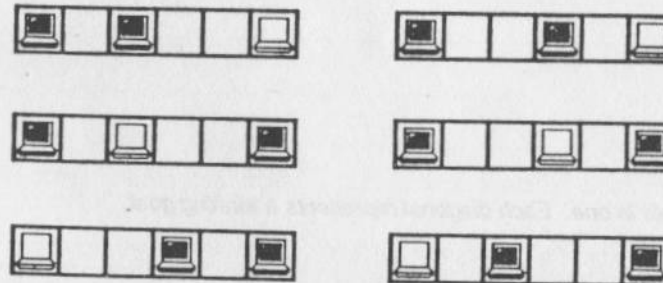


Figure 23. Moves that bring about a desirable formation for the Sandwich game.

If you want to go one step further, rate the moves that create the formations shown in Figure 23. A few examples are shown below.

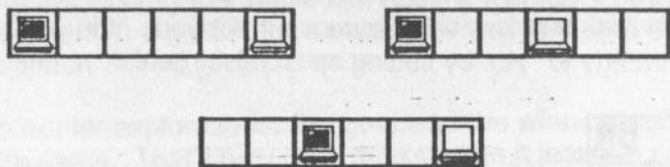


Figure 24. Moves that may be rated for the Sandwich game.

In its calculations, A.I. will examine moves leading toward the false goal as well as moves that lead toward the real goal. However, because you "unhooked" the W rating from moves that bring about Goal 2, A.I. knows that this goal is not a winning pattern. By using this trick, you can distinguish between winning patterns and goals, and you can indicate specific move situations A.I. should seek or avoid.

Two Goals in One

Another trick is to create a goal that actually includes two winning patterns. This is useful if you want to program a game with five goals. If you combine two of them, you can teach A.I. the game. For example, you could show A.I. that two different diagonals are both winning patterns by using the Show Goal option to create the "X" pattern below.



Figure 25. Two goals in one. Each diagonal represents a winning goal.

After you used the Show Goal option, you would use Rate Moves to rate the winning moves and other moves as well. Many ratings are required for each winning pattern. You must, for example, show A.I. that three pieces in one diagonal wins for all possible combinations of pieces in the other diagonal. Some examples appear below.

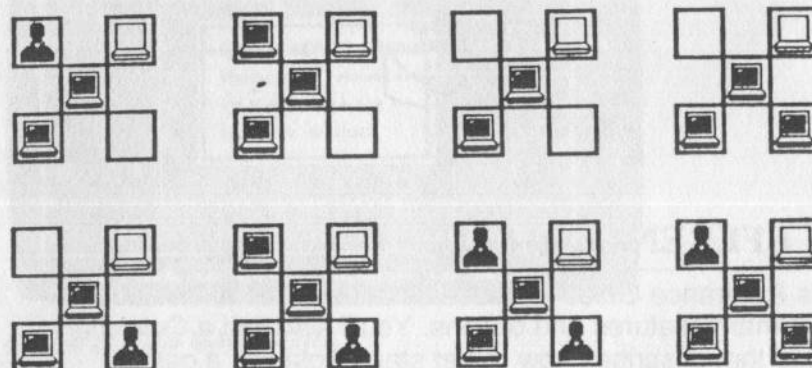


Figure 26. These moves are all rated a W. The moves are the same, but occur in different formations, so each must be rated separately.



REFERENCE GUIDE

This Reference Guide includes detailed information about *A.I.*'s many features and options. You'll also find a Quickstart Guide that describes how to get started playing a game against *A. I.*

The information in the Reference Guide assumes that you have already completed the Explorations section of this manual. Go through the explorations to learn about the *A.I.* program. Then refer to this Reference Guide when you need more information about any of the program's features.

How This Reference Guide Is Organized

Once you load *A.I.* into your computer, you'll see the MAIN MENU:

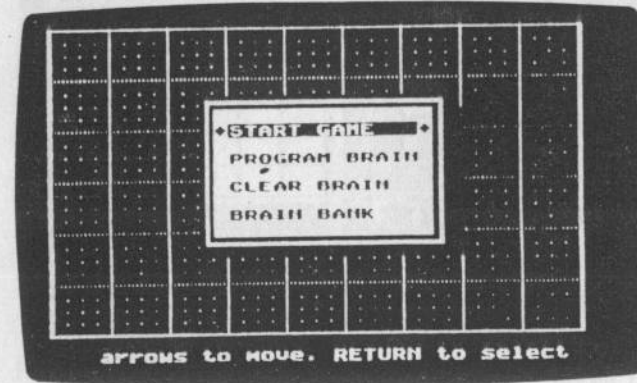
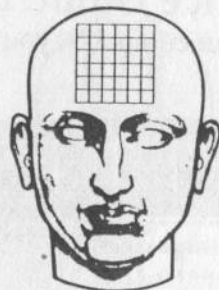


Figure 27. *The MAIN MENU.*

The Reference Guide begins by describing each option on the MAIN MENU and what happens when you choose one of those options. This is followed by the Quick Start Guide, a section that describes the different games that *A.I.* can play, and a section that brings you inside *A.I.* to explore how it learns.



START GAME

Choose this option to begin a new game against *A.I.* *A.I.* will use whatever brain is active. If you have just loaded *A.I.* into your computer, the active brain is blank.

Before you use this option, make sure the proper brain is active. If it is not, go into the Brain Bank and load the brain you want.

Once you select the Start Game option from the MAIN MENU, the game screen will appear.



Figure 28. *The Game board.*

On the game screen, you will see the game grid and the PLAY MENU. The different options on the PLAY MENU are described in the sections that follow.

Play

The Play option on the PLAY MENU tells *A.I.* to begin or continue a game. When you select it, the player whose turn it is can make a move.

Use Play in the following instances:

- When *A.I.* is supposed to make a game's opening move.
- After you move, to tell *A.I.* to move.
- After you change game options such as Foresight or Take Back.

Options

Selecting Options from the PLAY MENU calls up a new menu containing the following game options: Foresight, Whose Turn, Learning, Grid size, Brain Scan, and Ratings. All of these options may also be selected by pressing the key for the first letter of the option. Each is described below.



Figure 29. The OPTIONS MENU.

Foresight

Foresight governs how far A.I. looks ahead. You can choose this option to change Foresight any time during a game.

When Foresight is set at 0, A.I. simply looks for the move that has the highest rating. When Foresight is set at 1, A.I. considers all possible replies to every move it examines. Each time Foresight is raised one notch, A.I. looks ahead one more half move. When Foresight is set to 5, for example, A.I. looks 5 half moves ahead.

Raising Foresight a notch makes A.I. move six to 10 times slower. Unless you're unusually patient, set Foresight no higher than 2 when beginning a game on a board 5x5 or 6x6. As the game progresses, raising Foresight will not slow things as much, because A.I. will have fewer possible moves to consider.

To see how a higher Foresight level can improve A.I.'s playing skill, try this:

1. Take back a move that A.I. has played on a low Foresight level.
2. Raise the Foresight level.
3. See if A.I. chooses a different move.

This works best in a game where A.I. already knows at least one of the winning patterns.

Whose Turn

This option lets you change the player who is supposed to make the next move. When you press Return, the player on the move changes. An icon below the playing board indicates whose move it is. The icon is a pyramid when it is A.I.'s turn and a sphere when it is your turn. Use this option to change who makes the game's opening move.

Learning

When Learning is on and you tell A.I. that the game is over, it will compare the final position of pieces on the board with other patterns stored in the active brain. A.I. looks for similarities among these patterns to help it determine the actual winning patterns. A.I. changes its game-playing strategy each time it changes its opinion about what is, or what is not, a winning pattern.

The Learning state is indicated by the mortarboard (graduation cap) icon on the game screen. When there is a slash through the mortarboard, Learning is off.

NOTE: Programmed brains can only be played with Learning off.

You can turn Learning on or off anytime during a game, until you tell A.I. who won a game. After that, the board is cleared and it is too late.

When A.I. plays a game against you, it learns the winning positions, but it does not copy your strategy. (You can, however, *program* a strategy into A.I. See Explorations 7 - 10.) Therefore, once A.I. has learned all the winning patterns for the game you're playing, you should turn Learning off. That way, you don't have to wait after each game for A.I. to study the final position.

NOTE: For more information about how A.I. learns, see the How A.I. Works section of this Reference Guide.

Even when A.I. knows the goals, a Brain Scan may reveal that it continues to store other patterns that are specific instances of the true goal, as in Figure 30. This does not significantly hurt its playing skill.

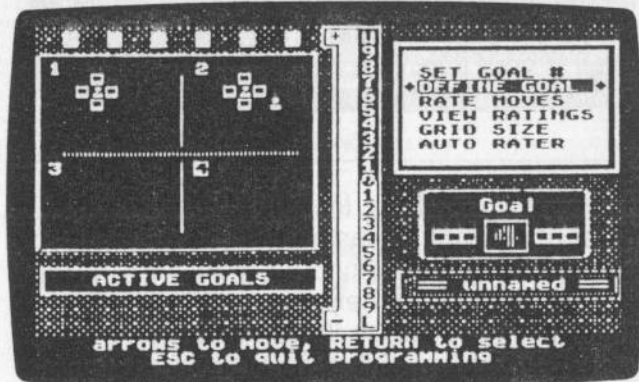


Figure 30. Goal #1 is correct. Goal #2 is a subset of that goal.

When Learning is On

When Learning is on, you must tell A.I. when a contest is over and who won. Once you do this, A.I. will spend 5 to 15 seconds to compare the final positions with others in its memory. If you quit before a game is over, or if you do not declare the result, no learning takes place.

When Learning is Off

When Learning is off, A.I. will tell you when a game has ended, and who has won. Whether A.I. will be right depends on the programming or learning that has gone on. When Learning is off, A.I. will not study the final position, so no learning will take place.

Grid Size

This option allows you to select the board size. The board remains square, but it can range from 3x3 to 6x6 squares.

You can play a game on any size board. You can even teach A.I. a game on one board size and then play it on another. Keep in mind, though, that certain games designed on small boards become very easy to win on large boards for the player who moves first. Tic-tac-toe is such an example. A.I. interprets the winning patterns in tic-tac-toe as three in a row, column, or diagonal, not as a complete row, column, or diagonal. If you wanted the winning pattern to be six in a row, column, or diagonal, that would be a different game. You may adjust grid size before, but not during, a game.

Brain Scan

Brain Scan takes an inside look at A.I.'s brain. It displays four quadrants, as shown below.



Figure 31. A Brain Scan reveals three winning goals.

Each quadrant represents a goal. The more playing experience a brain has, the closer these goals will be to the true goals.

NOTE: Learning brains always assume there are up to four goals.

Select Brain Scan to see how close A.I. is to determining the goals of the game you have created. If you want to hurry A.I.'s learning process on games that include more than one goal, attempt to win by achieving patterns that A.I. has not yet learned. The more it sees the goal, the quicker it learns.

Ratings

With the Ratings option turned on, numbers appear on the game board squares to indicate how highly A.I. "scores" each possible move. A.I. will also take more time to move. With Ratings off, the numbers do not appear.

The Ratings numbers range from 1 to 99. They do not have an absolute value that is valid from move to move or game to game. Instead, they are relative values that show you how A.I. compares its available moves at any given point in a game.

Whether Ratings is on or off is indicated by the gauge icon below the board and to the right. If there is a slash through the gauge, Ratings is off. If there is no slash, Ratings is on.

End Game

When you play against a learning brain, selecting the End Game option on the PLAY MENU allows you to tell A.I. the game is over and announce the result. When you select the End Game option, a list of three choices appears.

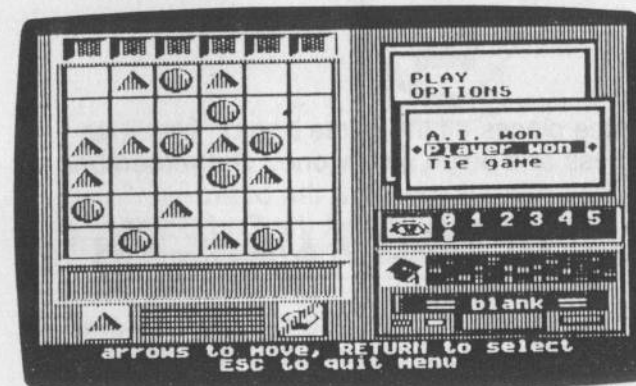


Figure 32. The END GAME MENU.

Pick one of these options to tell A.I. who won, or if the game was a draw. Once you indicate who won, A.I. studies the final position for 5 to 15 seconds and adds what it learns to the active brain. Remember that if a move is played that "wins" for both sides, you should declare the outcome as a tie.

NOTE: If you fail to notify A.I. that a game is over—by pressing Play instead, for example—use the Take Back option from the PLAY MENU to restore the winning position. Then select the End Game option. Also, if you select Game Over by mistake, you can return to the PLAY MENU with no harm done by pressing Esc.

NOTE: If you want to stop a game and not declare a result, do not select End Game. Instead, choose Clear Board to begin a new game, or press Esc to return to the MAIN MENU. The Clear Board option is explained below.

Take Back

This option allows you to take back your move or *A.I.*'s move. You can take back a move whenever it's your turn, or before you select Play on *A.I.*'s turn. You can take back as many moves as you wish, because *A.I.* remembers the order in which they were made.

Clear Board

This option takes all the pieces off the game board. Use it to stop a game in progress and begin a new one. But remember that no learning will take place if you clear the board. Learning only takes place when you select the End Game option from the PLAY MENU, and then tell *A.I.* the result of a game.

Setup Board

This option allows you to create a position by placing your pieces and *A.I.*'s pieces anywhere on the board. Once you choose the Setup Board option, use the arrow keys to move about the board, and press the Space Bar as often as necessary to place, change, or remove a piece.

You may want to use this option in any of the following instances, and perhaps others as well:

- To set up a position to see what move *A.I.* will make.
- To change *A.I.*'s move.
- To set up the same position several times and play different moves each time, to see which works best.



PROGRAM A BRAIN

Select the Program Brain option from the MAIN MENU if you want to program *A.I.* to play a game, rather than have *A.I.* learn the game through playing it. When you select Program Brain, the Programming screen appears.

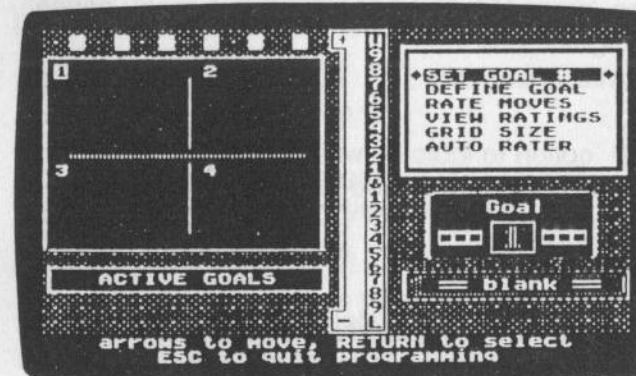


Figure 33. The Programming screen.

A.I.'s electronic brains can be programmed to play one game each. Each game can have up to four goals. You can also program a brain with a strategy for reaching each goal.

Any brain may be programmed. You can program a blank brain, or one that has learned from direct play or has already been programmed. If you program a learning brain, it becomes a programmed brain.

NOTE: Programmed brains can play games that learning brains can't. (See Exploration 10.)

Programming a brain requires two steps. First, you show *A.I.* the goals. Then you give *A.I.* a strategy for achieving those goals.

When giving *A.I.* a strategy, you have two choices. You can either tell *A.I.* to develop a strategy for achieving these goals, or you can teach it your own strategy. You teach *A.I.* a strategy by rating moves that lead toward a particular goal, or that prevent the opponent from doing so.

On-Screen Tutorial

You can ask *A.I.* to briefly demonstrate the programming process. To do so, press H instead of choosing a menu option on the programming screen.

Set Goal

Use the Set Goal# option to indicate which goal you want to work with. The Goal indicator tells you the current Goal.

Define Goal

You can show *A.I.* up to four goals for a game. Normally, each goal represents a winning pattern. (For examples of how a goal may be *different* from a winning pattern, see Exploration 10.)

To program a new goal into *A.I.*, or to change an existing goal, make sure the Goal indicator is set to the proper number. If it is not, set it using Set Goal #. When you enter goals, you can use any board size, but the size you use is the one *A.I.* will start with when you play this game.

When you select Define Goal, the board appears.

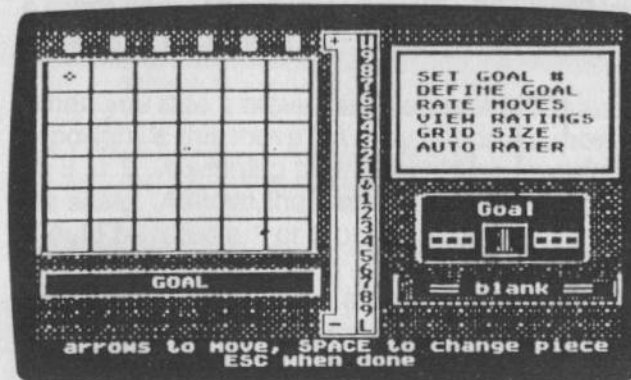


Figure 34. The Define Goal board.

If you are changing an existing goal, first remove the pieces from the old goal. Use the arrow keys to move the cursor. Then press the Space Bar to place or change the piece located under the cursor. A goal may have as many as six pieces. When you finish, press Esc.

When you change a goal, *A.I.* erases any strategy you programmed for the old goal.

NOTE: If you program a goal or change ratings in a learning brain, it becomes a programmed brain.

Rate Moves

Use the Rate Moves option to provide *A.I.* with a strategy for achieving a goal.

First, make sure the Goal indicator is set to the right goal. For example, if you are trying to rate moves for the first goal in a three-goal game, the goal indicator should be set to 1. If it is not, select Set Goal # to set it.

When you select Rate Moves, you will be shown an empty arrangement of squares in the same shape of the goal. An example is shown in Figure 35. The goal is displayed on the left, and the goal shape on the right.

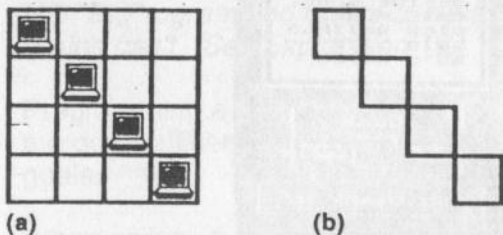


Figure 35: (a) A winning goal programmed into A.I. (b) The goal shape used to rate moves.

Use the arrow keys and Space Bar to set up a position within the goal shape. For the goal shown in Figure 35, a sample position is shown in Figure 36(a).

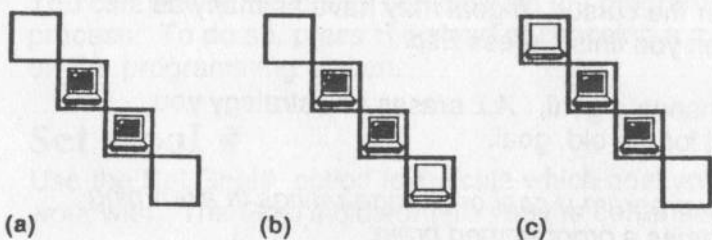


Figure 36: (a) A "formation" of pieces. (b) and (c) Moves to be rated within the formation.

Place the move you want to rate by moving the cursor to the square and pressing Return. Two different examples are shown in Figures 36(b) and (c).

After you indicate the move you wish to rate, use the arrow keys to move the arrow next to the ratings meter. Then choose a rating from -10 to 10. (The 10 rating is W, for Win. The -10 rating is L, for Lose.)

When you rate a move, ask yourself this question: How important is this move? A very important move might be worth an 8 or 9. A winning move should be awarded a W, the top of the scale. A move that barely, if at all, improves your position, should be rated a 1 or 2 or left unrated.

Moves that are not rated are automatically set to 0. If you want to delete a move rating, set it to 0.

Moves that worsen your position should be given negative ratings. Such moves generally occur only in games in which the goals include placement of the opponent's pieces. In such games, it is possible to make a move that loses immediately. These moves should be rated L, for Lose, as shown in Figure 37(b) below.

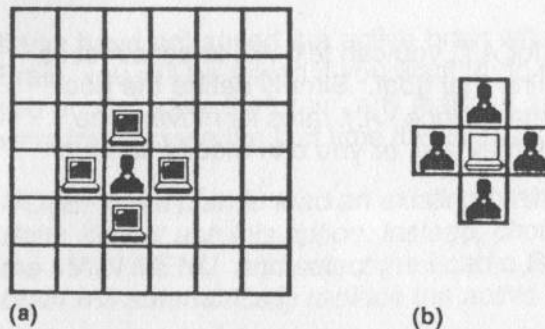


Figure 37: (a) A goal. (b) This move should be rated L, because it immediately loses.

To change a rating, follow the same procedure. The existing rating will appear when you indicate the move you wish to rate. Use the arrow keys to set the new rating.

View Ratings

When you select this option, *A.I.* will show you the ratings for moves related to the current goal. You can let *A.I.* start by looking for winning moves, or you can select another point at which to begin. *A.I.* will show moves by descending the ratings ladder one step at a time.

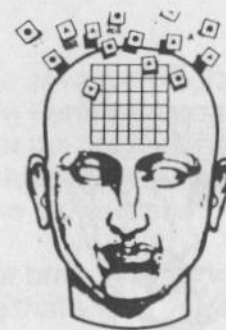
A.I. will skip over the 0 rating. All moves left unrated by *A.I.* when it did its own programming, or by you when you programmed *A.I.*, are rated 0. See the How *A.I.* Works section at the end of this Reference Guide for more information.

Grid Size

Use this option to change the size of the board grid. The size you choose is the size *A.I.* will assume you want to play the game on. You may, however, change the grid size of a game before you begin playing.

Auto Rater

After you define a goal for *A.I.*, you can tell *A.I.* to construct its own strategy for achieving that goal. Simply define the goal, and then select Auto Rater. Once *A.I.* rates its moves, you can leave the ratings as they are, or you can modify some or all of them.



CLEAR BRAIN

The Clear Brain option on the MAIN MENU instructs *A.I.* to dispose of the active brain and replace it with a blank one. A blank brain is a blank slate. It has neither learned nor been programmed to play any game. You should always use a blank brain when you want to program or play a new game.

If you have not saved the active brain when you select Clear Brain, you will be asked if you want to do so. Remember that if you do *not* save the brain, any learning or programming that went into it since the last time it was saved will be lost.

NOTE: If you plan to load an existing brain from the Brain Bank, do not use this option. Instead, choose Brain Bank from the MAIN MENU and select the Load a Brain option. The new brain will automatically replace the active brain.



BRAIN BANK

The Brain Bank option allows you to manage A.I.'s collection of programmed brains and learning brains. You should visit the Brain Bank whenever you want to save or load a brain, or when you need to create a new Brain Bank (data disk).

Once you choose the Brain Bank option from the MAIN MENU, a new menu will appear.

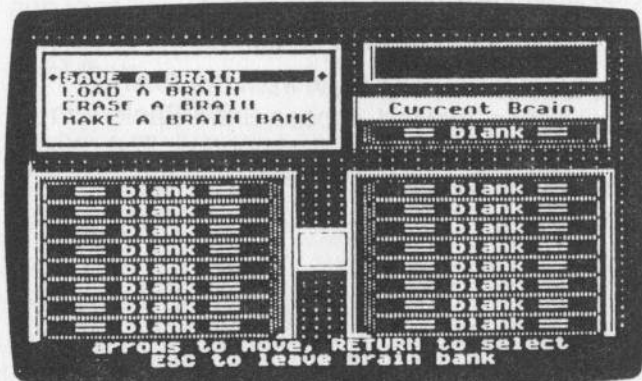


Figure 38. The Brain Bank Screen.

Save a Brain

Choose this option to save the programming or learning of the active brain in a Brain Bank for use at a later time. A.I. will ask you to select a slot for the brain. If you select the slot from which you originally loaded the brain, then the new version of the brain replaces the old version.

If you try to save your brain in a slot with a different brain, A.I. will ask if you're sure that's what you want to do. If you continue, the brain you save replaces the brain in that slot, but takes on its name.

If you choose a blank slot in which to store your brain, you will be asked to name your brain, even if your brain already has a name. This allows you to save a new version of a brain and not lose the older version. You might, for example, want to store several brains that all play the same game, but at different points in their learning.

Give the brain a name of up to 13 letters. The name should be descriptive. For example, TTTJOHNEXPERT might stand for a brain that plays tic-tac-toe expertly, and that was programmed by John.

Remember that if you do not save the contents of the active brain, it will be lost when you select Clear Brain or Load Brain from the MAIN MENU, or when you turn off the computer.

Load a Brain

Select this option when you want to tell *A.I.* to load a specified brain into its memory. *A.I.* will show you the list of brains in the Brain Bank currently in the disk drive. Use the arrow keys to highlight the brain you want. The learning icon in the center of the screen indicates whether the highlighted brain can learn. Learning brains can learn; programmed brains cannot. Press Return to select the highlighted brain.

Remember that each brain can play only one game. If you start to play a new game against a learning brain that knows a different game, *A.I.* will play very poorly, attempting to win by the rules of the old game. In time, however, it will change its learning and learn the new game. But a programmed brain cannot learn. Instead, it will continue to play for the patterns of the game it knows.

Erase a Brain

Use this option to remove a brain from a Brain Bank. Be sure that you really do not need the brain anymore. Once it's erased, you can't get it back.

Make a New Brain Bank

Choose this option to create a new data disk, or Brain Bank, to store *A.I.*'s electronic brains. Make sure that a blank disk, or a disk with contents you don't need, is in the drive.

Create new Brain Bank disks as you need them. Each Brain Bank can hold 16 brains.



QUICKSTART GUIDE

Loading *A.I.* Into Your Computer

To load *A.I.* into the computer, place the disk into the drive, label facing up and out, and turn on the computer. If the computer is already on, press the Control, Open-Apple and Reset keys at the same time. Then release the keys. *A.I.* will load into your computer, and the MAIN MENU will appear.

Playing a Game *A.I.* Already Knows

When *A.I.* learns or is programmed to play a game, it stores that knowledge in an electronic "brain." These brains are in turn stored on "Brain Bank" disks. Side 2 of your program disk is such a Brain Bank disk. It contains seven brains. All of these brains play different games. Some play games very well, while others play poorly.

To play one of these brains, follow this procedure:

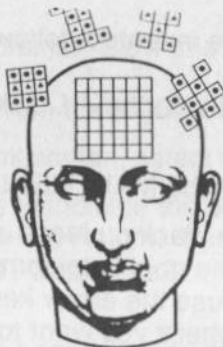
1. Select Brain Bank from the MAIN MENU.
2. From the Brain Bank screen, select Load a Brain.
3. Use the arrow keys to indicate the brain you want to play against, and then press Return. For a sample game, choose Challenge1, Challenge3, or Fullsandwich from the Brain Bank.
4. When the brain has been loaded, press Esc to return to the MAIN MENU.
5. From the MAIN MENU, select Start Game. The game screen will appear.
6. You and *A.I.* take turns placing pieces on empty squares, just as in tic-tac-toe. The goal of the game is to complete a certain pattern. Each brain plays a game with a different winning pattern. The Challenge3 brain plays a game with *three* different patterns. If you want to try to figure out the patterns by watching *A.I.*'s strategy, skip to step 7. If you want to see the winning patterns, press the B key to do a "Brain Scan." You can do a Brain Scan anytime it's your turn.
7. On your turn, use the arrow keys to place the cursor where you want to move. Then press the Space Bar. Your piece will be placed, and Play will be highlighted on the menu. To tell *A.I.* to move, press Return. *A.I.* will move, then it will be your turn again. Repeat Step 7 to continue playing the game. *A.I.* will tell you when the game has ended.

Playing a New Game Against *A.I.*

If you want to watch *A.I.* learn a new game, you must first invent the game. You do this by deciding on a winning pattern. The goal of the game will be to create that winning pattern. A winning pattern can consist of up to six different pieces, and it may be a row, column, diagonal, triangle, square, or any other shape that you can create on the game board. Your game may have as many as four different winning patterns.

To play a game that you've invented, follow this procedure:

1. Turn the computer off. Place the *A.I.* disk into the computer and turn it back on.
2. Select Start Game from the MAIN MENU.
3. You and *A.I.* take turns placing pieces on empty squares, just as in tic-tac-toe. The goal is to complete the winning pattern. On your turn, use the arrow keys to place the cursor on the square where you want to move. Then press the Space Bar. When you do so, your piece will be placed, and Play will be highlighted on the PLAY MENU. To tell *A.I.* to move, press Return. *A.I.* will move, then it will be your turn again.
4. Repeat Step 3 until you or *A.I.* plays a winning move. At that point, do not continue the game by pressing Return. Instead, use the arrow keys to indicate End Game on the menu, and then press Return. A new menu will appear. Choose one of the options from this menu to tell *A.I.* who has won the game.
5. Play the game as often as you like. The more times you play, the better *A.I.* will get at playing the game. You can press B to do a "Brain Scan" of *A.I.* anytime it's your move, to see how well *A.I.* is learning the game. If *A.I.* does not appear to be learning the game after you play 10 times or so, you may have chosen a type of game *A.I.* can't learn. Refer to The Games *A.I.* Can Play section in the Reference Guide for more information.
6. The knowledge *A.I.* gains about the game you play is stored in a brain. If you want *A.I.* to remember your game, you must store (save) the brain when you finish playing. To do so, press Esc until you return to the MAIN MENU. Then select the Brain Bank option from the menu. When the Brain Bank screen appears, select Save a Brain, and follow the instructions that will appear at the bottom of the screen.



GAMES A.I. CAN PLAY

A.I. plays games that follow these rules:

- A.I. and its opponent alternate making moves.
- A player making a move must place a piece on an unoccupied square.
- The first player to achieve a winning pattern (see below) is the victor. If one move creates a winning pattern for both players, the contest is a draw.

Winning Patterns

Games are won by the first player who completes a "winning pattern." A game may have up to four different winning patterns (or "goals"). A winning pattern can consist of any specific arrangement of as many as six pieces.

Here are some examples of different games with different winning patterns:

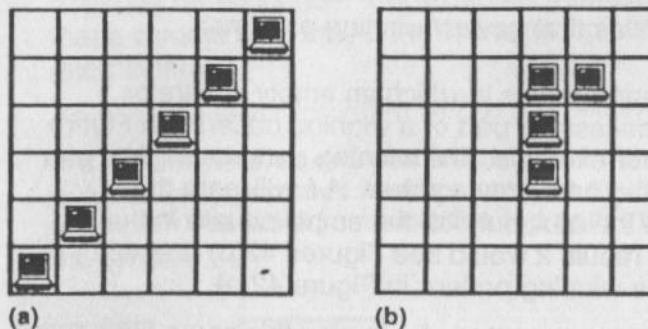


Figure 39. A game with two winning goals.

The game in Figure 39 can be won in two ways: by a diagonal going from bottom left to top right, or by an L shape three pieces tall with a "neighbor" to the right of the top piece.

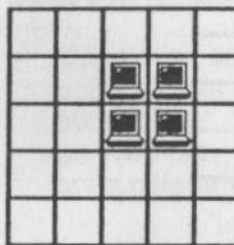


Figure 40. A game with one winning goal.

The game in Figure 40 has only one winning pattern, a square of four pieces all touching each other.

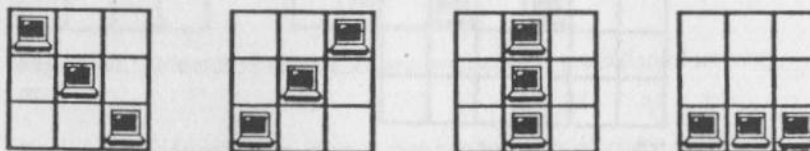


Figure 41. A game with the maximum of four winning goals: tic-tac-toe.

The game in Figure 41 has the maximum of 4 winning positions: a row, column, and both diagonals. You may recognize it as tic-tac-toe.

More Rules about Winning Patterns

Here are some rules that govern winning patterns:

- *A.I.* cannot learn games in which an empty square or squares is a necessary part of a winning pattern. In Figure 42(a) below, for example, the winning pattern consists of a box surrounding an empty square. *A.I.* will learn the importance of the box, but not the empty square in the middle. As a result, it would see Figures 42(b) and 42(c) as fulfilling the winning pattern in Figure 42(a).

You can, however, program *A.I.* to play this game, and others like it.

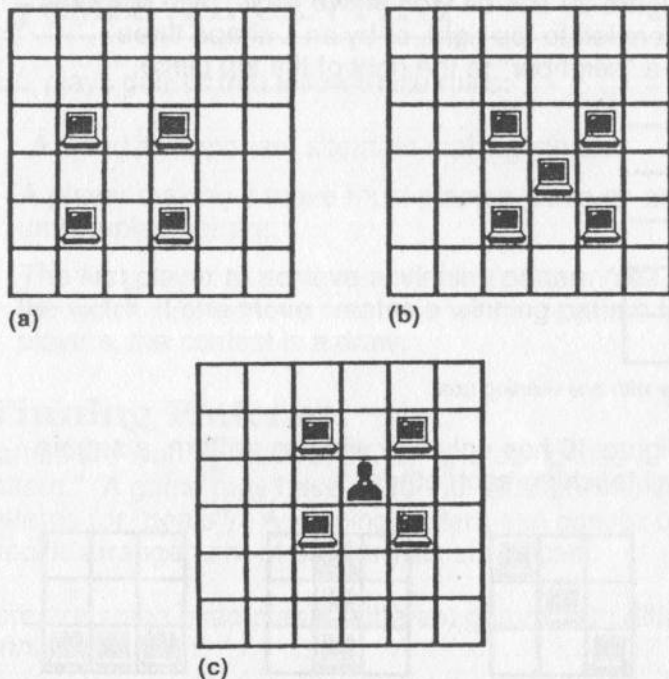


Figure 42: *A.I.* cannot learn (but can be programmed) to recognize winning patterns containing empty squares. If *A.I.* learns that the position in (a) is a winning goal, it will assume that (b) and (c) also represent winning goals.



- *A.I.* cannot learn games in which the winning pattern depends on sequence. For example, you cannot specify that a column wins only if the uppermost piece is the last one placed.
 You can, however, program *A.I.* to play this type of game.
- If a pattern wins on one part of the board, *A.I.* automatically assumes it wins anywhere on the board. For example, *A.I.* would consider the two patterns below to be the same.

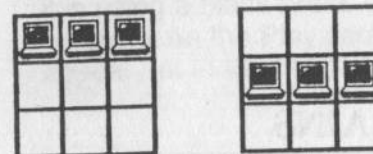


Figure 43. If a pattern wins somewhere on the grid, it must win everywhere on the grid. *A.I.* does not distinguish between these two patterns.

- Orientation counts. A diagonal going from top right to bottom left is not the same as one going from top left to bottom right. Also, a row is not the same as a column. As an example, *A.I.* would consider the two L patterns below to be different winning patterns.

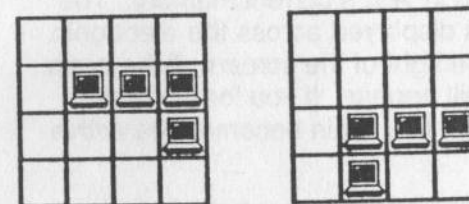


Figure 44. Orientation counts. These two patterns are different winning goals.

For more information about programming games with winning patterns that *A.I.* can't learn on its own, see Exploration 10.



A.I.'S ELECTRONIC BRAINS

A.I. acquires knowledge in two ways: by learning as it plays against an opponent, or by being programmed. In both cases, A.I. stores its knowledge in "electronic brains." Each electronic brain can learn one game (one set of winning patterns). You can save these brains on a Brain Bank disk and use them again anytime. Here is some important information about A.I.'s electronic brains:

- The *active* brain is the brain in A.I.'s current memory. The name of the active brain is displayed across the electronic brain slot toward the bottom right of the screen. If the brain is new, the word "Blank" will appear. If you load a brain from a Brain Bank data disk, that brain becomes the active brain.
- Any brain may be programmed or reprogrammed at any time. However, once a brain undergoes any sort of programming, it cannot learn from actual play. It can be changed only through further reprogramming. A.I. automatically turns Learning *off* when you play such a brain, and will announce when it thinks it has won or lost, basing its judgment on the goals programmed into it.

- Brains that have not been programmed can learn through direct play on the game board. Such brains can continue to refine their knowledge about the goals of a game, and can even learn new ones, up to a maximum of four goals per game. When you play this type of brain, A.I. will turn Learning *on*. When Learning is on, you must tell A.I. when a game has ended and who has won.
- Blank brains are unused brains. They have not been programmed, and they have not learned by playing games against you. When you teach A.I. a new game, make sure it is using a blank brain. Check the name on the electronic brain slot on the Play screen. If the word Blank does not appear, go to the MAIN MENU and select Clear Brain.



HOW A.I. WORKS

In learning to play board games, *A.I.* draws upon two fields of artificial intelligence: pattern recognition and game theory. In this section, you'll read how *A.I.* uses techniques from each of these fields to help it first recognize winning patterns, and then to develop a strategy for achieving those winning patterns on the game board.

NOTE: The information in this section assumes that you have some experience playing games against A.I. and programming A.I.'s "brains." To learn how to play games against A.I., complete Explorations 1 to 4 in the Exploration section of this User's Guide. To learn how to program A.I.'s brains, complete Explorations 5 to 10.

Pattern Recognition: What Are the Winning Patterns?

When learning is on, *A.I.* finds winning patterns by studying the final positions in two seemingly contradictory ways. *A.I.* compares the final position with previous final positions to find what patterns, if any, they have in common. But because *A.I.* expects four different winning patterns to exist for each game, it also looks for final positions distinctly different from previous ones. Such positions may contain goals that *A.I.* has not yet "seen."

What *A.I.* Does When a Game Has Ended

When the player tells *A.I.* that a game has ended and who has won, *A.I.* stores the final position and the winning move. It then follows the steps described below to refine its knowledge of the game's goals.

1. *A.I.* temporarily removes the winning move from the board. There should now be no winning pattern on the board. *A.I.* looks to see if any of its existing goals appear on the board. If they do, *A.I.* throws them out. In Figure 45 below, *A.I.* realizes that one of its goals must be wrong, and it rejects that goal.

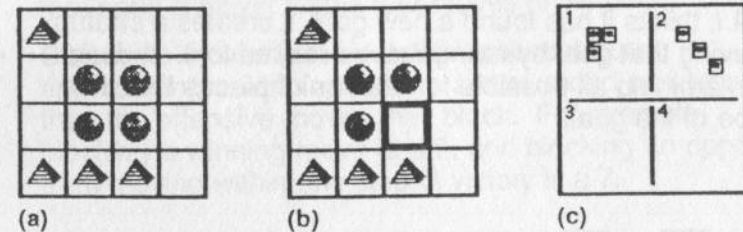


Figure 45: (a) *A.I.* has just been defeated. (b) The position before *A.I.*'s opponent played the winning move. There should be no winning patterns in this position. (c) The patterns that *A.I.* believes are winning goals. When *A.I.* finds Goal #1 in (b), it no longer considers it a winning goal.

2. After discarding any such goals, *A.I.* returns the winning move to the board to restore the final position.

- If *A.I.* has recognized fewer than four goals, it adds the final position to its list of goals for this game. If it already has four goals, however, *A.I.* temporarily considers the final position as a fifth goal. Then it compares all five goals with each other, looking for the two that are most similar. It finds the pattern these two goals have in common, and substitutes that pattern for the goals themselves. Because it is replacing two goals with one, it is back to the limit of four.
- If *A.I.* has modified an existing goal or added a new one, it also creates a table of ratings for moves related to that goal. See the Game Theory section in this Reference Guide for more details.

Because *A.I.* can never be sure it knows the goals of a game, it will repeat the above steps after any game for which Learning was on. Therefore, if a Brain Scan reveals that *A.I.* has in fact figured out the goals of a game, you can turn Learning off to save time.

To learn more about how *A.I.* identifies and stores goals, see Exploration 3: *A.I.* Learns.

Game Theory: Rating Moves Within Given Positions

When *A.I.* thinks it has found a new goal, it creates a strategy for achieving that goal by rating moves related to it. It does this by examining all possible formations of pieces that fit in the shape of the goal.

In Figure 46(a) below, a winning goal is displayed. Figure 46(b) shows the "shape" of the goal. Figures 46(c) and 46(d) are examples of the many formations that can occur in that shape. The unshaded pieces represent the move that *A.I.* will rate.

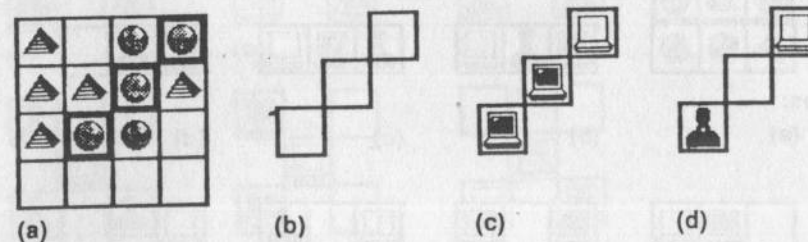


Figure 46: (a) The outlined squares contain the winning goal. (b) The shape of this goal. (c) and (d) Sample formations and moves within the goal shape.

Rules for Rating Moves

A.I. follows these rules in assigning ratings:

- A winning move receives a W.
- A move that brings the player within one move of victory is rated an 8. A 6 is awarded to any move that brings a player two moves from victory, a 4 to any move that brings a player within three moves of winning, and a 2 to moves that bring a player within four moves of winning.
- Defensive moves that block the opponent from approaching a winning pattern are awarded one less point than the offensive moves they block. For example, blocking a winning move is a 9, and blocking an opponent from moving within one step of victory is a 7.

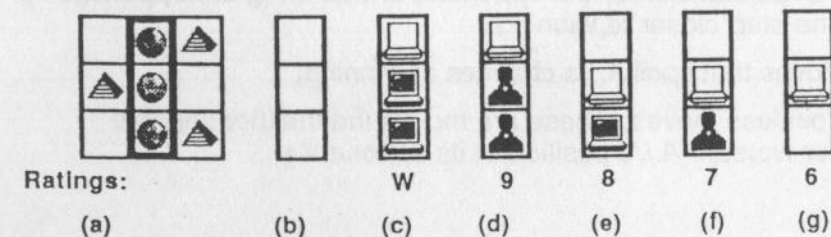


Figure 47: (a) The winning goal for a game. (b) The goal shape. (c) - (g) Formations, moves, and move ratings.

The situation is more complicated if a winning goal includes opponent's pieces as well as your own, as in Figure 48 below.

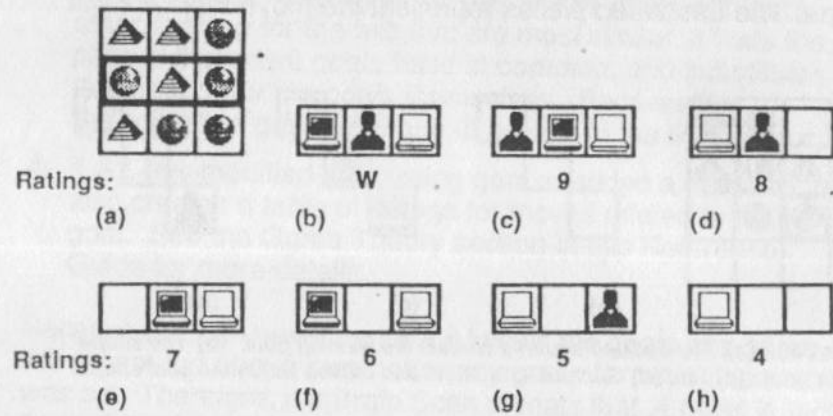


Figure 48: (a) A position containing a winning goal. (b) - (e) A.I. rates moves made when the potential loser's piece (the middle piece) is in place. (f) - (h) A.I. rates moves without the potential loser's piece.

In this case, A.I. evaluates an opponent's piece as equal to two of its own pieces.

Categories of Moves

A.I. classifies moves that it wants to rate in one of five categories:

1. Moves that win for A.I., or bring it one step closer to winning.
2. Moves that block the opponent from winning, or from coming one step closer to winning.
3. Moves that win for the opponent, or that bring the opponent one step closer to winning.
4. Moves that spoil A.I.'s chances of winning.
5. Pointless moves. These are moves that neither improve nor worsen A.I.'s position or its opponent's.

Examples of each move are shown below.

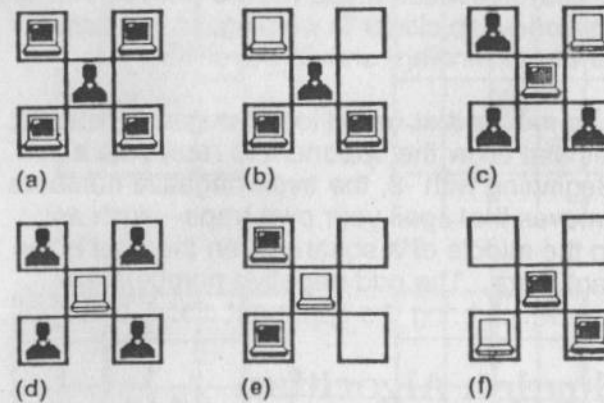


Figure 49: (a) A winning goal. (b) A move that leads towards the goal. (c) A blocking move. (d) A losing move. (e) A move which spoils the player's chances for winning. (f) A pointless move.

Negative Ratings

For games with a goal that includes the placement of the loser's pieces as well as the winner's, A.I. assigns negative ratings as well as positive ones. The moves assigned negative ratings are moves from categories 3 and 4 described in the previous section. These negative ratings are necessary because moves from category 3 (moves that win for the opponent, or that bring the opponent one step closer to winning) and category 4 (moves that spoil A.I.'s chances of winning) worsen a player's position within a goal shape. In other words, A.I. doesn't want to make a move that moves you closer to winning, or that reduces its chances of winning. That's why it rates any move that falls into category 3 or 4 with a negative number.

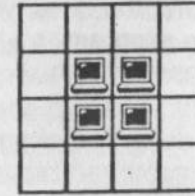
By contrast, moves from categories 1 and 2 never receive negative ratings. That's because these moves (moves that win for A.I., that bring it one step closer to winning, or that help block an opponent from winning) should never hurt A.I.

A.I. assigns an L to moves that would lose the game instantly, and a -9 to moves that allow the opponent to reply with a winning move. Beginning with -8, the even negative numbers are assigned to moves that spoil your own traps—such as putting a piece in the middle of a square when the goal is to trap your opponent there. The odd negative numbers are assigned to moves which bring the opponent closer to a goal.

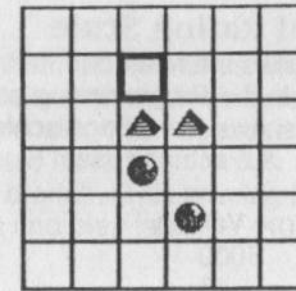
Board Evaluation Algorithm

As you are playing a game, here is how A.I. evaluates a move. First it places the shape of the first goal over the empty square that it's thinking of moving to. (See Figure 50.) Then A.I. looks up the rating of that move. A.I. repeats this for each possible positioning of the goal shape over the empty square. It totals up these ratings to arrive at a rating for moving into that square for that particular goal. It then repeats that procedure for each goal, and sums the rating for all goals.

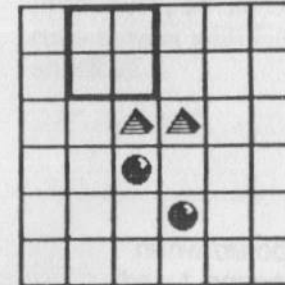
A.I. repeats that procedure for all empty squares. The square that emerges with the highest value is the square A.I. will move into.



(a)

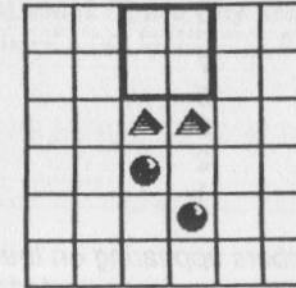


(b)



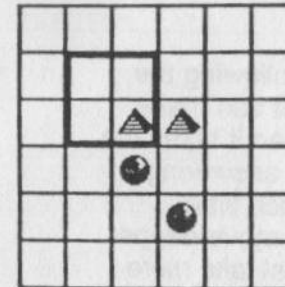
(c)

4



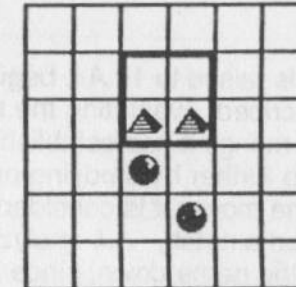
(d)

4



(e)

6



(f)

8

Figure 50: (a) A winning goal. (b) A.I. is evaluating a potential move into the outlined square. (c) - (f) Four overlays of the goal shape on the square. The numbers represent ratings for moving into the square in the particular goal-shape orientation. A.I. repeats the procedure for other game goals and totals all ratings to arrive at a final rating for the square. After evaluating all empty squares, A.I. selects the one with the highest rating.

A.I.'s Internal Rating Scale

The ratings *A.I.* uses are related to, but different from, the ones you see and use in the Programming section of the program. The table below shows the ratings scale seen in the program, and the numbers *A.I.* actually uses.

Ratings	True Values
W	5000
9	48
8	32
7	18
6	9
5	7
4	5
3	3
2	2
1	1

NOTE: The numbers appearing on the game board when Ratings is turned on have been scaled to fit between 1 and 100.

Foresight

When Foresight is raised to 1, *A.I.* begins by following the process just described, evaluating the moves it can make according to the ratings it has established. Then it takes the process one step further by studying possible answering moves to all of the moves it is considering. Each time Foresight is raised a notch, *A.I.* analyzes one move deeper. This also slows the game down, since *A.I.* must take more time to ponder each move.

Ratings

When the Ratings option is off, *A.I.* performs what is known as Alpha Beta pruning. This is a method in which a branch of moves is eliminated from consideration because it is clear that one player or the other would have been foolish to make the move that gave rise to the branch. This pruning helps *A.I.* avoid needless calculation and play faster.

When Ratings is turned on, *A.I.* displays values for each move it can make. Alpha Beta pruning interferes with this procedure, so *A.I.* does not use it. That's why the program runs slower with Ratings on—it's not taking the Alpha Beta shortcut.

BIBLIOGRAPHY

Here are some books and articles that provide more information about artificial intelligence and the debate over whether machines can really think like human beings.

Asimov, Isaac, "No Solution", *American Way*, July 8-21, 1986.

Davis, Dwight, "Artificial Intelligence Enters the Mainstream", *High Technology*, July 1986.

Dreyfus, Hubert L., *What Computers Can't Do: The Limits of Artificial Intelligence*, Harper & Row, 1972.

Dreyfus attacks what he considers are inflated claims made by the 'artificial intelligentsia' of the AI community. His attack provoked many indignant responses.

Feigenbaum, Edward A. and McCorduck, Pamela, *The Fifth Generation -- Artificial Intelligence and Japan's Challenge to the World*, Addison-Wesley Publishing Co., 1983.

The authors advocate a national initiative to prevent Japan from overtaking the United States in information technology and AI. A somewhat biased but interesting overview of the AI industry in the United States and Japan.

Hartnell, Tim, *Exploring Artificial Intelligence on Your Apple II*, Bantam Books, 1985.

A collection of BASIC programs illustrating the basics of AI.

Harvey, Brian, *Computer Science Logo Style: Intermediate Programming*, MIT Press, 1985.

Harvey presents Logo as a powerful tool of computer science. Readable and philosophical. More appropriate for experienced Logo programmers than beginners.

Haugeland, John, *Artificial Intelligence: The Very Idea*, MIT Press, 1986.

A discussion of AI's philosophical underpinnings written for the laymen but appropriate for philosophers. Brings perspective to the notion that human thinking and machine computing can be the same.

Hoffman, Paul, "The Next Leap In Computers", *The New York Times Magazine*, December 7, 1986.

Hofstadter, Douglas R., *Metamagical Themas: Questing for the Essence of Mind and Pattern*, Basic Books, 1985.

A collection of Hofstadter's 26 columns from *Scientific American* with seven new essays, all connected with additional comments, observations and concerns. Explores how people perceive, think and create.

Hutchison, Michael, *MegaBrain*, Beech Tree Books, 1986.

A summary of current efforts to understand and artificially stimulate the brain, especially in areas where results challenge traditional beliefs.

James, Mike, *Artificial Intelligence in BASIC*, Butterworth & Co., 1984.

Contains sample BASIC programs to demonstrate AI concepts.

Johnson, George, *Machinery of the Mind*, Times Books, 1986.

A journalistic introduction to AI, with stories of AI pioneers and their programs, explanations of current AI issues and problems, an introduction to AI programming languages, and an examination of philosophical questions raised by AI.

Levy, David, *Computer Gamesmanship*, Century Publishing (London), 1983.

Levy is an international chess master. Excellent for those interested in the creation and structure of game programs.

McCorduck, Pamela, *Machines Who Think*, W. H. Freeman, 1979.

An important historical work based on McCorduck's interviews with AI pioneers, capturing their aspirations and fears as the author traces the beginnings of AI through current (1979) realities.

O'Shea, T. O. and Eisenstadt, M. (Editors), *Artificial Intelligence: Tools, Techniques, and Applications*, Harper & Row, 1984.

An anthology of 15 articles that provide an introduction to AI.

Poundstone, William, *The Recursive Universe*, Contemporary Books, 1985.

The book's subtitle, *Cosmic Complexity and the Limits of Scientific Knowledge*, tells only half the story. In alternating chapters, Poundstone presents the concepts of modern physics and the complexities of Conway's game of "Life," providing a host of thought provoking analogies for those exploring the possibility of machine intelligence.

Raphael, Bertram, *The Thinking Computer: Mind Inside Matter*, W. H. Freeman and Co., 1976.

Raphael explains how AI research addresses the goal of having computers perform significant intellectual tasks. Major topics include problem representation, search strategies, problem solving, natural language, perception, robots, and new applications of AI.

Rose Frank, "The Black Knight of AI", *Science 85*, March 1985.

Schlesinger, Arthur M., Jr., "The Challenge of Change", *The New York Times Magazine*, July 27, 1986.

Turkle, Sherry, *The Second Self: Computers and the Human Spirit*, Simon & Schuster, 1984.

Investigates the effect of computers on people. Turkle suggests that the computer has provoked users to reconsider themselves with questions such as: What does it mean to think? What does it mean to be human?

Waldrop, M. Mitchell, "The Machinations of Thought", *Science 85*, March 1985.

Weizenbaum, Joseph, *Computer Power and Human Reason*, W.H. Freeman and Co., 1976.

A classic work that sounds an alert to potential dangers of computer use. Although the author's position is often extreme, his cautions should at least be considered by everyone interested in the application of technology.

Winston, Patrick H., *Artificial Intelligence*, 2nd edition, Addison-Wesley, 1977.

The classic college level textbook by the director of MIT's Artificial Intelligence Laboratory. Presents a clear explanation of the basic theoretical concepts of AI. Highly recommended for those able to invest the considerable time required.

A.I.

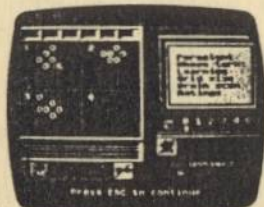
AN EXPERIENCE WITH ARTIFICIAL INTELLIGENCE

"Can a computer really think?"

Battle the power of a computerized master-mind...and discover for yourself the fundamental concepts of artificial intelligence.

Imagine designing a game to play with a computer that doesn't know the basic rules or how to win. Of course you are the winner of the first several games. But then, before your eyes, the computer learns and uses your strategies against you. You soon find it impossible to win. The computer has beaten you at your own game!

The secret of A.I.'s ability to continuously learn lies in the two major fields of artificial intelligence—Pattern Recognition and



Expert Systems. With A.I., you can actually access the "mind" of the computer to see first-hand how it uses these principles to learn from each game's final position.

And, A.I.'s unique BrainScan feature graphically displays each step of the "thought process" the computer uses to develop winning strategies.

You create an intelligence just like sophisticated researchers do.

Watching your computer learn strategies is just the beginning. You can then remove, disable or change those strategies in an attempt to outplay A.I. You can also program a set of rules into A.I. *before* you start a game. For example, you can program A.I. to look ahead 2 moves, 5 moves or as many moves as you like—just to see how that ability affects the progress of a game.

With A.I., the challenge is never-ending because the program is constantly changing, forever testing who's smarter—you or your computer.



The excitement keeps building as you create hundreds of games to save and challenge your friends.

A.I.'s easy-to-use Tutorial, User's Manual and Reference Guide start you off with almost a dozen different games which introduce you to each facet of A.I. Then you're on your way to creating hundreds of your own games which you can save and copy to challenge your friends.

One thing is for certain, after you've used A.I.—An Experience with Artificial Intelligence, you'll never look at your computer the same way again!

Contents: One double-sided disk and a complete User's Handbook.

Scholastic Inc.
2931 East McCarty St.
P.O. Box 7502
Jefferson City, MO 65102