MASSACHUSETTS INSTITUTE OF TECHNOLOGY

ARTIFICIAL INTELLIGENCE LABORATORY

AI Memo No. 191 updated June 1976

BIBLIOGRAPHY

Available publications may be ordered from:

Ms. Mary Castro
MIT AI Publications
MIT Artificial Intelligence Laboratory
545 Technology Square, Room 800E
Cambridge, Massachusetts 02139 USA

Titles preceded by asterisks are not available.

When ordering, specify MIT AI publication number and author, and enclose the correct amount of US currency. Prices of publications are listed as $x.xx, and include the cost of postage -- surface rate to North American countries and airmail printed matter rate to all other countries. The cost of other postal services is additional.

Reports assigned an "AD" number are available from the National Technical Information Service, Operations Division, Springfield, Virginia 22151 in both microfiche and paper copies.

#1-7 Concerned with early LISP development (see A.I. Memo 50).

#8 Recursive Functions of Symbolic Expressions and their Computation by Machine, John McCarthy, (in Computer Programming and Formal Systems, North-Holland Publishing Co., Amsterdam, 1963.

#9-14 Concerned with early LISP development (see A.I. Memo 50).

#15 SML-Examples of Proofs by Recursion Induction, John McCarthy.

#16 A Question-Answering Routine, A.V. Phillips.

#17 Programs with Common Sense, John McCarthy. Published in Proc. Math. of Thought Processes, HMSO, 1958. (in Semantic Information Processing)

#18 Some Results from a Pattern Recognition Program Using LISP, Louis Hodes.

#19 Concerned with early LISP development (see A.I. Memo 50).

#20 Puzzle Solving Program in LISP, John McCarthy.

#21 The Proofchecker, Paul Abrahams (complete version in Mathematical Algorithms, April 1966, Vol. 1, No. 2; July 1966, Vol. 1, No. 3.)

#22-26 Concerned with early LISP development (see A.I. Memo 50).

#27 Simplify, Tim Hart.

#28 Concerned with early LISP development (see A.I. Memo 50).

#29 Introduction to the Calculus of Knowledge, Bertram Raphael, November 1961.

#30 The Tree Prune (TP) Algorithm, Dec. 1961, Tim Hart & Daniel Edwards, Revised Oct. 1963 - The alpha-beta Heuristic. (See Slagle, J. and J.N. Dixon: JACM, Vol. 16, No. 2, April 1969, pp. 189-207.)

#31 A Basis for a Mathematical Theory of Computation, John McCarthy, Jan. 1962. Published in: Western Joint Computer Conference, 1961.

#32 On Efficient Ways of Evaluating Certain Recursive Functions, John McCarthy.

#33 Universality of (p=2) Tag Systems and a 4 Symbol 7 State Universal Turing Machines, Marvin Minsky (in Computation).

#34 A New Eval Function, John McCarthy.

#35 Concerned with early LISP development (see A.I. Memo 50).

#36 On the Problem of the Effective Definition of Random Sequence, Michael Levin, Marvin Minsky, Roland Silver.

#37 Some Identities Concerning the Function subst [x;y;z], Lewis Norton.

#38 Machine Understanding of Linguistic Information - A Survey and Proposal, Bertram Raphael. (In Semantic Information Processing)

¢39 Concerned with early LISP development (see A.I. Memo 50).

¢40 A Note on the Feasibility of Application of the Davis Putnam Proof Procedure to Elementary Number Theory, Donald Dawson.

¢41 A Chess Playing Program, Alan Kotok.

¢42 Proposed Research on Learning, Marvin Minsky.

¢43 Proposal for a General Learning Machine, Bertram Raphael. (In Semantic Information Processing)

¢44 A Simple Direct Proof of Post's Normal Form Theorem, Marvin Minsky. (In Computation)

¢45 A Question-Answerer for Algebra Word Problems, Daniel Bobrow. (In Semantic Information Processing)

¢46 A Heuristic Program to Solve Geometric Analogy Problems, T.G. Evans, Oct. 1962. (In Semantic Information Processing)

¢47 A Proposal to Investigate the Application of a Heuristic Theory of Tree-searching to a Chess-playing Program, Burton Bloom, February 1963.

¢48 Neural Nets and Theories of Memory, Marvin Minsky, March 1963.

¢49 Computer Representation of Semantic Information, Bertram Raphael, April 1963. (In Semantic Information Processing)

¢50 Concerned with early LISP development; More information is in AI Memos 98, 116, 190 and:

MACLISP Reference Manual, D. Moon et al, Project MAC; The LISP 1.5 Programming Manual, J. McCarthy et al, M.I.T. Press, Cambridge, Mass.; The Programming Language LISP, Berkeley Enterprises, Newton, Mass.; LISP 1.5 Primer, Clark Weissman, Prentice Hall, Englewood Cliffs, N.J.

¢51 METEOR: A LISP Interpreter for String Transformation, Daniel Bobrow, 1963.

¢52 Universality of Tag Systems with p=2, John Cocke & Marvin Minsky, April, 1963; (in Computation,)

¢53 ARGUS: Real-Time handwritten character-recognition system, Warren Teitelman, May 1963.

¢54 Proposal for a FAP Language Debugging Program, Joel Winett, June, 1963.

¢55 Primitive Recursion, Michael Levin, July, 1963.

¢56 A Proposal for a Geometry Theorem Proving Program, Tim Hart, September 1963.

¢57 MACRO Definitions for LISP, Tim Hart, October, 1963.

¢58 A LISP Garbage Collector Algorithm Using Serial Secondary Storage, Marvin Minsky, October, 1963 (MAC-M-129).

*59 Operation of a Semantic Question-Answering System, Bertram Raphael, November 1963. (In Semantic Information Processing,)

*60 Recent Improvements in DDT, D. Edwards, M. Minsky, November 1963.

*61 Mathscope: Part I: A Proposal for a Mathematical Manipulation-Display System, Marvin Minsky, November 1963. More advanced results in Ph.D. thesis of William Martin MAC-TR-36, January 1967. (AD-657-283).

*62 Derivator I. A Program for Visual Inspection of Solutions to First-Order Non-Linear Differential Equations, Marvin Minsky, Dec. 1963, MAC-M-124.

*63 Secondary Storage in LISP, Daniel Edwards, Dec. 1963, MAC-M-128.

*64 LISP Exercises, Tim Hart, Michael Levin, Jan. 1963 MAC-M-134. (In The Programming Language LISP, Berkeley Ent., Newton, MA)

*65 The Graphical Typewriter, A Versatile Remote Console Idea, MAC-M-135, Marvin Minsky.

*66 Natural Language Input for a Computer Problem Solving System, Daniel Bobrow, MAC-M-148. In Semantic Information Processing)

*67 Revised User's Version, Time Sharing LISP for CTSS, William Martin, Tim Hart. MAC-M-153.

*68 Syntax of the New Language, Michael Levin, May, 1963, MAC-M-158.

*69 New Language Storage Conventions, Michael Levin, May 1964, MAC-M-159.

*70 Hash-Coding Functions of a Complex Variable, William Martin, June 1964. (In Symbolic Mathematical Laboratory, MAC-TR-36) (AD-657-283).

*71 String Manipulation in the New Language, Daniel Bobrow, July 1964, MAC-M-176.

*72 Proposed Instructions on the GE 635 for List Processing and Push Down Stacks, Michael Levin Sept. 1964, MAC-M-183.

*73 Unrecognizable Sets of Numbers, Marvin Minsky, Seymour Papert, Nov. 1964. Revised Sept. 1965. (In JACM, Vol. 13, No. 2, April, 1966).

*74 CTSS LISP Notice-Supplement to A.I. Memo No. 67, T. Hart, Dec. 1964, MAC-M-206.

*75 Television Camera-to-Computer Adapters: PDP-6 Device 770, Marvin Minsky, Jan. 1965, MAC-M-215.

*76 The COMIT Feature in LISP II, Daniel Bobrow, Feb. 1965, MAC-M-219.

*77 Matter, Mind and Models, Marvin Minsky, March 1965. Published in 1965 IFIP Congress. (In Semantic Information Processing)

*78 Topics in Model Theory, Michael Levin, May 1965, MAC-M-240.

*79 PDP-6 LISP Input-Output for the Dataphone, William Martin, June 1965, MAC-

M-241.

☼80 PDP-6 LISP Input-Output for the Display, William Martin, June 1965, MAC-M-242.

☼81 PDP-6 TECO, Peter Samson, July 1965, MAC-M-250.

☼82 MAC PDP-6 DECtape File Structure, Peter Samson, July 1965, MAC-M-249.

☼83 Use of MACDMP, Peter Samson, July 1965, MAC-M-248. (See A.I. Memo 118).

☼84 EDIT and BREAK Functions for LISP, Warren Teitelman, MAC-M-264.

☼85 Syntax and Display of Mathematical Expressions, William Martin, July 1965. (In Symbolic Mathematical Laboratory, MAC-TR-36). (AD-657-283)

86 Design of the Hand, Marvin Minsky, August 1965, MAC-M-258 (see AI Memo 267) ($.90)

☼87 FLIP - A Format List Processor, Warren Teitelman, MAC-M-263. (See Bolt, Beranek and Newman Report No. 10, 15 July 1967. 50 Moulton St., Cambridge, Mass.)

☼88 MACTAP, A PDP-6 DECtape Handling Package, Peter Samson, Sept. 1965, MAC-M-267.

☼89 A Theory of Computer Instructions, Ward Douglas Maurer, Sept. 1965. (In Programming: An Introduction to Computer Languages and Techniques, Maurer, Holden-Day Inc., San Francisco).

☼89A Computer Experiments in Finite Algebra, W.D. Maurer, June 1965.

☼89B Computer Experiments in Finite Algebra II, W.D. Maurer, Dec. 1965.

90 MIDAS, Peter Samson, Oct. 1968. Revised version of MAC-M-279, Oct 1965 (see A.I. Memos 179, 214). ($1.70)

☼91 A Useful Algebraic Property of Robinson's Unification Algorithm, Tim Hart, Nov. 1965, MAC-M-285.

☼92 Topics in Model Theory, Michael Levin, Jan. 1966, MAC-M-294.

☼93 A New Version of CTSS LISP, Joel Moses, Robert Fenichel, MAC-M-296.

☼94 A New Machine-Learning Technique Applied to the Game of Checkers, Arnold Griffith, March 1966, MAC-M-299 (see Artificial Intelligence, Vol 5, No 2).

☼95 A Program Feature for CONVERT, Adolfo Guzman, Harold McIntosh, April 1966, MAC-M-305.

☼96 POLYBRICK: Adventures in the Domain of Parallelepipeds, Adolfo Guzman, May 1966. See MAC-TR-37. (AD-656-041).

☼97 Symbolic Integration, Joel Moses, June 1966.

☼97A Symbolic Integration, II, Joel Moses, Oct. 1966. See Symbolic Integration

MAC-TR-47. (AD-662-666).

☼98 PDP-6 LISP, Peter Samson, June 1966, MAC-M-313. See A.I. Memo 116.

☼99 CONVERT, Adolfo Guzman, Harold McIntosh, June 1966, MAC-M-316. Published in COMM, ACM 9, 8, Aug. 1966, pp. 604-615.

☼100 The Summer Vision Project, Seymour Papert, July 1966.

☼101 SIDES 21, Richard Greenblatt, Jack Holloway; described by Donald Sordillo, August 1966, MAC-M-320.

☼102 A Quick Look at Some of Our Programs, Gerald Sussman, Adolfo Guzman, July 1966 .

☼103 Additions to the Vision Library, John White, Aug. 1966.

☼104 Output to the PDP-6 Calcomp Plotter, Jack Holloway, Aug. 1966.

☼105 Modifications to PDP-6 Teletype Logic, Tom Knight, Aug. 1966.

☼106 An Input Macro for TECO, Donald Eastlake, Sept. 1966, MAC-M-324.

☼107 Music Playing on the PDP-6, Donald Sordillo, Aug. 1966, MAC-M-326.

☼108 A Primitive Control P Feature, Donald Eastlake, Oct. 1966, MAC-M-329.

☼109 SCPLOT BIN, Donald Sordillo, Oct. 1966, MAC-M-333 (see A.I.Memo 112)

☼110 Figure Boundary Description Routings for the PDP-6 Vision Project, John White, Sept. 1966, MAC-M-328.

☼111 Summer Vision Programs, Leslie Lamport, Oct. 1966, MAC-M-332.

☼112 CHAR PLOT, Donald Sordillo, Oct. 1966, MAC-M-334. (Replaced by A.I. Memo 125).

☼113 A Description of the CNTOUR Program, Larry Krakauer, Nov. 1966, MAC-M-335.

☼113A A Description of the CNTOUR Program, Revised, Larry Krakauer.

☼114 A Step by Step Computer Solution of Three Problems in Non-Numerical Analysis, William Martin, July 1966. (See Symbolic Mathematical Laboratory MAC-TR-36) (AD-657-283.

☼115 Program Memo) Peter Samson, Dec. 1966.

116 PDP-6 LISP (LISP 1.6), January 1967, Revised April 1967 (see A.I. Memo 190 and MACLISP Reference Manual, D. Moon et al, Project MAC). ($1.30)

This is a mosaic description of PDP-6 LISP, intended for readers familiar with the LISP 1.5 Programmer's Manual or who have used LISP on some other computer. Many of the features, such as the display, are subject to change. Thus, consult a PDP-6 systems programmer for any differences which may exist between LISP of Oct. 14, 1966 and present LISP on the system tape (sic).

*117 Memo, Russell Noftsker.

*118 PDP-6 Software Update, Thomas Knight, Jan. 1967; Revised June 1967, Donald Eastlake (System Program).

*119 A Primitive Recognizer of Figures in a Scene, Adolfo Guzman, Jan. 1967; see his Dissertation: Some Aspects of Pattern Recognition by Computer, MAC-TR-37. (AD-656 041).

*120 Vision Memo, Marvin Minsky, Feb. 1967.

121 Estimating Stereo Disparities, Marvin Minsky, Feb. 1967. ($.90)

An interesting practical and theoretical problem is putting bounds on how much computation one needs to find the stereo-disparity between two narrow-angle stereo scenes. By narrow angle I mean situations wherein the angle subtended by the eyes is a very few degrees; the kind of correlation-disparity method discussed here probably is not applicable to the wide-angle stereo we will usually use for scene-analysis in the Project.

122 Remarks on Correlation Tracking, Marvin Minsky, March 1967. ($.90)

123 Computer Tracking of Eye Motions, Marvin Minsky, Seymour Papert, March 1967 (see A.I. Memo 253). ($.90)

This memo explains why the Artificial Intelligence group is developing methods for on-line tracking of human eye movements. It also gives a brief resume of results to date and the next steps.

*124 Paradoxical Perception, Seymour Papert (never written).

*125 CHAR PLOT, Michael Speciner, March 1967 (see A.I. Memo 138).

126 A Quick Fail-Safe Procedure for Determining whether the GCD of 2 Polynomials is 1, Joel Moses, March 1967, MAC-M-345. ($.90)

Experiments by Collins have shown that given two polynomials chosen at random, the GCD has a high probability of being 1. Taking into account this probability and the cost of obtaining a GCD (some GCDs of polynomials of degree 5 in two or three variables can take on the order of a minute on the 7094), it appears that a quick method of determining whether the GCD is exactly 1 would be profitable. While no such method is known to exist, a fail-safe procedure has been found and is described here.

*127 Incorporating MIDAS Routines into PDP-6 LISP, Roland Silver, March 1967; revised in Nov. 1967 (127A).

*128 Hardware and Program Memo, Michael Beeler, March 1967.

*129 EUTERPE: A Computer Language for the Expression of Musical Ideas, Stephen Smoliar, April 1967 (see also A.I. Memo 141).

*130 A Miscellany of Convert Programming, Adolfo Guzman, Harold McIntosh, April 1967, MAC-M-346.

☼131 POLYSEG, Arnold Griffith, April 1967.

☼132 Additions to LAP, John White, July 1967 (See AI Memo 152.

☼133 A Glossary of Vision Terms, Russ Abbott, June 1967.

☼134 PSEG: Standardization of Data, Jim Bowring, June 1967.

☼135 Automata on a 2-Dimensional Tape, Manuel Blum, Carl Hewitt, July 1967.

☼136 Matrix Inversion in LISP, John White, July 1967.

☼137 PLANNER: A Language for Proving Theorems, Carl Hewitt, July 1967. Replaced by A.I. Memo 168, MAC-M-386, Oct. 1968. Revised June 1969.

☼138 The Calcomp Plotter as an Output Device in TS and LISP, Michael Speciner, July 1967.

☼139 Decomposition of a Visual Scene into Bodies, Adolfo Guzman, Sept. 1967, MAC-M-357.

☼140 Perceptrons and Pattern Recognition, Marvin Minsky, Seymour Papert, Sept. 1967. (In Perceptrons)

☼141 EUTERPE-LISP: A LISP System with Music Output, Stephen Smoliar, Sept. 1967. (see also A.I. Memo 243)

☼142 STRING, Peter Samson, Sept. 1967.

143 Stereo and Perspective Calculations, Marvin Minsky, Sept. 1967. ($1.30)

☼144 I/O Test, Michael Beeler, Sept. 1967.

☼145 A Fast-Parsing Scheme for Hand-Printed Mathematical Expressions, William Martin, Oct. 1967, MAC-M-360.

☼146 PICPAC: A PDP-6 Picture Package, Roland Silver, Oct. 1967.

☼147 A Multiple Procedure DDT, Thomas Knight, Jan. 1968.

147A DDT Reference Manual, Eric Osman, Sept. 1971. ($2.10)

This memo describes the version of DDT used as the command level of the AI Laboratory Time Sharing System (ITS). Besides the usual program control, examination, and modification features, this DDT provides many special utility commands. It also has the capability to control several programs for a user and to a single instruction continue mode and interrupt on read or write reference to a given memory location.

☼148 SUBM - A CONVERT Program for Constructing the Subset Machine Defined By a Transition System, Harold V. McIntosh, Jan. 1968.

☼149 REC/8 - A CONVERT Compiler of REC for the PDP-6, Harold V. McIntosh, Jan. 1968.

☼150 CGRU and CONG - CONVERT and LISP Programs to Find the Congruence

Relations of a Finite State Machine, Harold V. McIntosh, Jan. 1968.

¤151 Functional Abstraction in LISP and PLANNER, Carl Hewitt, Jan. 1968. (Replaced by AI TR-258.)

¤152 PDP-6 LAP, John White, Jan. 1968; replaced by A.I. Memo 190.

¤153 REEX - A CONVERT Program to Realize the McNaughton-Yamada Analysis Algorithm, Harold McIntosh, Jan. 1968.

154 The Artificial Intelligence of Hubert L. Dreyfus; A Budget of Fallacies, Seymour Papert, Jan. 1968. ($2.10)

This paper reviews the earlier writing of Hubert Dreyfus on Artificial Intelligence. It reveals many fallacies in Dreyfus' thinking.

155 A Left to Right then Right to Left Parsing Algorithm, William Martin, Feb. 1968. ($.90)

Determination of the minimum resources required to parse a language generated by a given context free grammar is an intriguing and yet unsolved problem. It seems plausible that any unambiguous context free grammar could be parsed in time proportional to the length, n, of each input string. Early has presented an algorithm which parses many grammars in time proportional to n, but requires n¤n on some. His work is an extension of Knuth's algorithm, which leads to a very efficient parse proportional to n of deterministic languages. This memo presents a different extension of Knuth's method. The algorithm is probably more efficient than Early's on certain grammars; it will fail completely on others. The essential idea may be interesting to those attacking the general problem.

¤156 Linear Decision and Learning Models, Marvin Minsky, March 1968 (see A.I. Memo 167).

¤157 Time-Sharing LISP for the PDP-6, John White, March 1968. Replaced by A.I. Memo 190.

¤158 SARGE: A Program for Drilling Students in Freshman Calculus Integration Problems, Joel Moses, March 1968, MAC-M-369.

¤159 Numerical Solution of Elliptic Boundary Value Problems by Spline Functions, Jayant Shah, April 1968, MAC-M-371.

160 Focusing, B.K.P. Horn, May 1968. ($1.70)

This memo describes a method of automatically focusing the image dissector camera. The same method can be used for distance measuring.

161 ITS 1.5 Reference Manual, D. Eastlake, R. Greenblatt, J. Holloway, T. Knight, S. Nelson, MAC-M-377. July 1969 (Revised form of ITS 1.4 Reference Manual, June 1968). (see AI Memos 147A, 238, 260A, 261A) ($2.60)

This reference manual consists of two parts. The first (sections 1 through 6) is intended for those who are either interested in the ITS 1.5 time sharing monitor for its own sake or who wish to write machine language programs to run under it. Some knowledge of PDP-6 (or PDP-10) machine language is useful in reading this part. The second part (sections 7, 8, and 9) describes three programs that run under ITS. The first program (DDT) is a modified machine language debugging program that also replaces the monitor command level (where the user is typing directly at the monitor) present in most time-sharing systems. The remaining two (PEEK and LOCK) are a status display and a miscellaneous utility program. It should be remembered that the A.I. Laboratory PDP-6 and PDP-10 installation is undergoing continuous software and hardware development which may rapidly outdate this manual.

¢162 Remarks on Visual Display and Console Systems, Marvin Minsky, June 1963, revised July 1968.

¢163 Holes, Patrick Winston, August 1968, revised April 1970.

¢164 Producing Memos Using TJ6, TECO and the Type 37 Teletype, Larry Krakauer, Sept. 1968. Replaced by Memo 164A.

¢164A The Text-Justifier TJ6, R. Greenblatt, B.K.P. Horn, L. Krakauer, June 1970. See AIM 358.

¢165 Description and Control of Manipulation by Computer-Controlled Arm, Jean-Yves Gresser, Sept. 1968.

¢166 Recognition of Topological Invariants by Modular Arrays, Terry Beyer, September 1968.

¢167 Linear Separation and Learning, Marvin Minsky and Seymour Papert, October 1968. (In Perceptrons)

¢168 PLANNER, Carl Hewitt, MAC-M-386, October 1968, revised June 1969. (Replaced by AI-TR-258.)

¢169 PEEK and LOCK, Donald Eastlake III, MAC-M-387, November 1968; replaced by revised A.I. Memo 161A, 260, 261.

¢170 WIRElist, John Holloway, January 1969.

171 Decomposition of a Visual Scene Into Three-Dimensional Bodies, Adolfo Guzman, January 1969. MAC-M-391.

¢172 Robot Utility Functions, Stewart Nelson, Michael Levitt, February 1969; replaced by Revised A.I. Memo 161A, July 1969.

173 A Heuristic Program That Constructs Decision Trees, Patrick Winston, March 1969. ($1.70)

Suppose there is a set of objects, (A,B,...,Z), and a set of tests, (T1,T2,...,TN). When a test is applied to an object, the result is either T or F. Assume the tests may vary in cost and the objects may vary in probability of occurrence. One then hopes that an unknown object may be identified by applying a sequence of tests. The appropriate test at any point in the sequence in general should depend on the results of previous tests. The problem is to construct a good test scheme using the test costs, the probabilities of occurence, and a table of test outcomes.

174 The Greenblatt Chess Program, Richard Greenblatt, Donald Eastlake III, Stephen Crocker, April 1969. ($.90)

Since mid-November 1966 a chess program has been under development at the Artificial Intelligence Laboratory of Project MAC at M.I.T. This paper describes the state of the program as of August 1967 and gives some of the details of the heuristics and algorithms employed.

¤175 On Optimum Recognition Error and Reject Trade-off, C.K. Chow, April 1969.

¤176 Discovering Good Regions for Teitelman's Character Recognition Scheme, Patrick Winston, May 1969.

¤177 Preprocessor for Programs Which Recognize Scenes, H.N. Mahabala, August 1969.

178 The Image Dissector Eyes, B.K.P. Horn, August 1969. ($1.70)

This is a collection of data on the construction, operation and performance of the two image dissector cameras. Some of this data is useful in deciding whether certain shortcomings are significant for a given application and if so, how to compensate for them.

¤179 The Arithmetic-Statement Pseudo-Ops: .I and .F, B.K.P. Horn, August 1969.

180 The Integration of a Class of Special Functions with the Risch Algorithm, Joel Moses, MAC-M-421. ($.90)

We indicate how to extend the Risch algorithm to handle a class of special functions defined in terms of integrals. Most of the integration machinery for this class of functions is similar to the machinery in the algorithm which handles logarithms. A program embodying much of the extended integration algorithm has been written. It was used to check a table of integrals and it succeeded in finding some misprints in it.

¤181 PROGRAMMAR: A Language for Writing Grammars, Terry Winograd, November 1969. (see A.I. Memo 282 & AI-TR-235)

¤182 Display Functions in LISP, Thomas Binford, 1970.

183 On Boundary Detection, Annette Herskovits and Thomas Binford, July 1970. ($2.10)

A description is given of how edges of prismatic objects appear through a television camera serving as visual input to a computer. Two types of edge-finding predicates are proposed and compared, one linear in intensity, the other non-linear. A statistical analysis of both is carried out, assuming input data distorted by a Gaussian noise. Both predicates have been implemented as edge-verifying procedures, i.e. procedures aiming at high sensitivity and limited to looking for edges when approximate location and direction are given. Both procedures have been tried on actual scenes. Of the two procedures, the non-linear one emerged as a satisfactory solution to line-verification because it performs well in spite of surface irregularities.

184 Parsing Key Word Grammars, William Martin, MAC-M-395. ($.90)

Key word grammars are defined to be the same as context free grammars, except that a production may specify a string of arbitrary symbols. These grammars define languages similar to those used in the programs CARPS and ELIZA. We show a method of implementing the LR(k) parsing algorithm for context free grammars which can be modified slightly in order to parse key word grammars. When this is done the algorithm can use many of the techniques used in the ELIZA parse. Therefore, the algorithm helps to show the relation between the classical parsers and key word parsers.

☼185 Proposal to ARPA for Research on Artificial Intelligence at MIT, 1970-1971, Marvin Minsky and Seymour Papert, December 1970.

☼186 Seymour Papert, LOGO report, Never written (see A.I. Memo 246).

187 Form and Content in Computer Science, Marvin Minsky, December 1969; ACM 'Turing Lecture' August 1969. ($1.30)

☼188 A Stability Test for Configurations of Blocks, Manuel Blum, Arnold Griffith, Bernard Neumann, February 1970.

☼189 Construction of Decision Trees, E. Roger Banks, February 1970.

☼190 Interim LISP Progress Report, John White, March 1970.

191 A. I. Bibliography, April 1970, updated later (FREE!)

☼192 Removing Shadows in a Scene, Richard Orban, August 1970.

☼193 Learning, Teaching and A.I., Marvin Minsky and Seymour Papert. Never written.

194 Movie Memo, Michael Beeler, April 1970. ($1.30)

This is intended as a brief explanation of how to use the Kodak movie camera in sync with a display.

☼195 INSIM1: A Computer Model of Simple Forms of Learning, Thomas L. Jones, April 1970.

☼196 Hypergeometric Functions in MATHLAB, Lewis Wilson, May 1970.)

☼197 A Simple Algorithm for Self-Replication, Terry Winograd, term paper written November 1967, released as A.I. Memo May 1970.

☼198 Cellular Automata, E. Roger Banks, June 1970.

199 The Function of FUNCTION in LISP, Joel Moses, June 1970. ($.90)

A problem common to many powerful programming languages arises when one has to determine what values to assign to free variables in functions. Different implementational approaches which attempt to solve the problem are considered. The discussion concentrates on LISP implementations and points out why most current LISP systems are not as general as the original LISP 1.5 system. Readers not familiar with LISP should be able to read this paper without difficulty since we have tried to couch the argument in ALGOL-like terms as much as possible.

◊200 1968-1969 Progress Report, Marvin Minsky and Seymour Papert.

◊201 Comparative Schematology, Michael Paterson and Carl Hewitt, November 1970.

◊202 Peter Samson's Music Processor, BIG, Michael Beeler, July 1970.

◊203 Micro-Planner Reference Manual, Gerald Sussman and Terry Winograd, July 1970.

203A Micro-Planner Reference Manual, (revised), Gerald Sussman, Terry Winograd, and Eugene Charniak, December 1971. ($1.30)

This is a manual for the use of the Micro Planner interpreter, which implements a subset of Carl Hewitt's language, PLANNER and is now available for use by the Artificial Intelligence Group.

◊204 Extending Guzman's SEE Program, Martin Rattner, July 1970 (M.I.T. B.S. Thesis June 1970).

◊205 Look-Ahead Strategies in One Person Games with Randomly Generated Game Trees, David S. Johnson, M.I.T. M.S. thesis August 1968, released as A.I. Memo July 1970.

◊206 The Vision Laboratory, Part One, Thomas O. Binford, July 1970.

◊207 More Comparative Schematology, Carl E. Hewitt, August 1970. (Replaced by AI TR-258.)

◊208 Teaching Procedures in Humans and Robots, Carl Hewitt. Paper presented April 1970, issued as A.I. Memo September 1970. (Replaced by AI TR-258.)

◊209 Digital Flight Simulation, David Silver, March 1971. Program listing available from line printer FLIGHT >, DSK DS;

210 A User's Guide to the A.I. Group LISCOM LISP Compiler, Jeffrey P. Golden, December 1970. ($.90)

The operation of a version of a fast arithmetic LISP compiler for ITS on the PDP-10 is discussed.

◊211 Equivalence Problems in a Model of Computation, Michael S. Paterson, Ph.D. thesis (Trinity College, Cambridge University), 1967; issued in November 1970.

◊212 Terry Winograd, Using the Language-Understanding System, Never written (see A.I. Memo 282).

◊213 The Computer as a Performing Instrument, Gordon Mumma and Stephen Smoliar. Presented as a MAC seminar February 1970, issued as A.I. Memo February 1971.

◊214 Linking Loader for MIDAS, Peter Samson, March 1971. Originally printed as MAC Memo in January 1966.

◊215 How to Get Onto the System, Mark Dowson, April 1971.

☆215A Instant TJ6, How to Get the System to Type Your Papers, M. Dowson, Sept. 1971.

☆216 Theories, Pretheories, and Finite State Transformations on Trees, Mitchell Wand, May 1971.

217 Computer Proofs of Limit Theorems, W.W. Bledsoe, Robert S. Boyer, W.H. Henneman, June 1971. ($.90)

Some relatively simple concepts have been developed which when incorporated into existing automatic theorem proving programs (including those using resolution), enable them to prove efficiently a number of the limit theorems of elementary calculus, including the theorem that differentiable functions are continuous. These concepts include:

(1) A limited theory of types, to designate whether a given variable belongs to a certain interval on the real line, (2) An algebraic simplification routine, (3) A routine for solving linear inequalities, applicable to all areas of analysis, and (4) A limit heuristic, designed especially for the limit theorems of calculus.

218 Information Theory and the Game of Jotto, Michael Beeler, August 1971. ($.90)

The operation and use of a PDP-10 program for playing the word game JOTTO are described. The program works by making the move which will give it the most information in the information-theoretic sense.

☆237 An Inquiry Into Algorithmic Complexity, Patrick E. O'Neil, September 1971.

238 ITS Status Report, Donald Eastlake, April 1972. ($2.10)

ITS is a time-shared operating system designed for the Artificial Intelligence Laboratory DEC PDP-10/PDP-6 installation and tailored to its special requirements. This status report described the design philosophy behind the ITS system, the hardware and software facilities of the system implemented with this philosophy, and some information on work currently in progress or desirable in the near future.

239 HAKMEM, M. Beeler, R.W. Gosper, R. Schroeppel, February 1972. ($2.60)

HAKMEM is a collection of 191 miscellaneous items, most of which are too short to justify documentation individually. Topics are related to computers, but range over number theory and several other branches of mathematics, games, programming techniques, and electronic circuits. Items come from A. I. work, outside contributors, and computer folklore.

"I keep HAKMEM in the bathroom, it's great reading." - Knuth

☆240 11SIM Reference Manual, Donald Eastlake, December 1971.

240A 11SIM Reference Manual, (revised), Donald Eastlake, February 1972. ($1.70)

A program that simulates a Digital Equipment Corporation PDP-11 computer and many of its peripherals on the A.I. Laboratory Time Sharing System (ITS) is described from a user's reference point of view. This simulator has a built in DDT-like command level which provides the user with the normal range of DDT facilities but also with several special debugging features built into the simulator. The DDT command language was implemented by Richard M. Stallman while the simulator was written by the author of this memo.

241 An A.I. Approach to English Morphemic Analysis, Terry Winograd, September 1971. ($.90)

This paper illustrates an approach toward understanding natural language through the techniques of artificial intelligence. It explores the structure of English word-endings both morpho-graphemically and semantically. It illustrates the use of procedures and semantic representations in relating the broad range of knowledge a language user brings to bear on understanding and utterance.

©243 Using the EUTERPE Music System, Stephen W. Smoliar, October 1971.

©244 Mark Dowson, Never written.

©245 Proposal to ARPA for Research on Artificial Intelligence at M.I.T., 1971-1972, Marvin Minsky and Seymour Papert, October 1971.

246 A Computer Laboratory for Elementary Schools, Seymour Papert, October 1971 (LOGO Memo No. 1). ($.90)

This is a research project on elementary education whose immediate objective is the development of new methods and materials for teaching in an environment of computers and computer-controlled devices. Longer term objectives are related to theories of cognitive processes and to conjectures about the possibility of producing much larger changes than are usually though possible in the expected intellectual achievement of children. This proposal is formulated in terms of the self-sufficient immediate objectives.

247 Teaching Children Thinking, Seymour Papert, October 1971 (LOGO Memo No. 2). ($1.30)

The purpose of this essay is to present a grander vision of an educational system in which technology is used not in the form of machines for processing children but as something the child himself will learn to manipulate, to extend, to apply to projects, thereby gaining a greater and more articulate mastery of the world, a sense of the power of applied knowledge and a self-confidently realistic image of himself as an intellectual agent. Stated more simply, I believe with Dewey, Montessori and Piaget that children learn by doing and by thinking about what they do. And so the fundamental ingredients of educational innovation must be better things to do and better ways to think about oneself doing these things.

248 Twenty Things To Do With A Computer, Seymour Papert and Cynthia Solomon, June 1971 (LOGO Memo No. 3). ($1.70)

249 Teaching Children to be Mathematicians VS. Teaching About Mathematics, Seymour Papert, July 1971 (LOGO Memo No. 4). ($1.30)

Being a mathematician is no more definable as knowing a set of mathematical facts than being a poet is definable as knowing a set of linguistic facts. Some modern math ed reformers will give this statement a too easy assent with the comment: Yes, they must understand, not merely know. But this misses the capital point that being a mathematician, again like being a poet, or a composer or an engineer, means doing, rather than knowing or understanding. This essay is an attempt to explore some ways in which one might be able to put children in a better position to do mathematics rather than merely to learn about it.

¢250 PLANNER Implementation Proposal to ARPA, 1971-1972, Carl Hewitt, December 1971.

¢251 Mini-Robot Proposal to ARPA, Marvin Minsky, January 1972.

¢252 Artificial Intelligence Progress Report, Marvin Minsky and Seymour Papert, January 1972. (Identical with pp. 129-224 of the 1971 Project MAC Progress Report VIII.) AD-754-820.

253 The Computer-Controlled Oculometer: A Prototype Interactive Eye Movement Tracking System, Matthew J. Hillsman, R. Wade Williams and John S. Roe, February 1972. (Originally published as a report to NASA in September 1970.) ($2.60)

The Oculometer electro-optic subsystem utilizes near-infrared light reflected specularly off the front surface of the subject's cornea and diffusely off the retina, producing a bright pupil with an overriding corneal highlight. An electro-optic scanning aperture vidi-sector within the unit, driven by a digital eye-tracking algorithm programmed into the PDP-6 computer, detects and tracks the centers of the corneal highlight and the bright pupil to give eye movement measurements. A computer-controlled, moving mirror head motion tracker directly coupled to the vidissector tracker permits the subject reasonable freedom of movement. Various applications of this system, which are suggested by the work reported here, include:

(a) using the eye as a control device,
(b) recording eye fixation and exploration patterns,
(c) game playing,
(d) training machines, and
(e) psycho-physiological testing and recording.

254 NIM: A Game-Playing Program, Seymour Papert and Cynthia Solomon, Feb. 1972 (LOGO Memo No. 5). ($.90)

This note illustrates some ideas about how to initiate beginning students into the art of planning and writing a program complex enough to be considered a project rather than an exercise on using the language or simple programming ideas. The project is to write a program to play a simple game (one-pile NIM of 21) as invincibly as possible. We developed the project for a class of seventh grade children we taught in 1968-69 at the Muzzy High School in Lexington, Mass. (This work was supported by NSF Contract No. NSF-C 558 to Bolt, Beranek and Newman, Inc.) This was the longest programming project these children had encountered, and our intention was to give them a model of how to go about working under these conditions.

¢255 Why Conniving Is Better Than Planning, Gerald Jay Sussman, Feb. 3, 1972.

255A Why Conniving Is Better Than Planning, Gerald Sussman and Drew McDermott, April 1972 (see also A.I. Memo 259). ($1.30)

This paper is a critique of a computer programming language, Carl Hewitt's PLANNER, a formalism designed especially to cope with the problems that Artificial Intelligence encounters. It is our contention that the backtrack control structure that is the backbone of PLANNER is more of a hindrance in the solution of problems than a help. In particular, automatic backtracking encourages inefficient algorithms, conceals what is happening from the user, and misleads him with primitives having powerful names whose power is only superficial. An alternative, a programming language called CONNIVER which avoids these problems, is presented from the point of view of this critique.

*256 Efficiency of Equivalence Algorithms, Michael J. Fischer. Presented at Symposium on Complexity of Computer Computations, T.J. Watson Research Center, March 22, 1972; issued as A.I. Memo April 1972.

*257 A Two Counter Machine Cannot Calculate 2**n , Richard Schroeppel, May 1973.
AD-773-918.

*259 Conniver Reference Manual, Drew McDermott and Gerald Sussman, May 1972.

259A THE CONNIVER REFERENCE MANUAL, Drew McDermott and Gerald Sussman, January 1974. ($2.10) AD-773-555.

This manual is an introduction and reference the latest version of the Conniver programming language, an A. I. language with general control and data-base structures.

*260 LOCK, Donald E. Eastlake, June 1972.

260A LOCK, Donald E. Eastlake, January 1974. ($.90) AD-773-920.

LOCK is a miscellaneous utility program operating under the ITS system. It allows the user to easily and conveniently perform a variety of infrequently required tasks. Most of these relate to console input-output or the operation of the ITS system.

*261 PEEK, Eastlake, May 1973.

261A PEEK, Donald E. Eastlake, February 1974. ($.90) AD-773-925.

PEEK is a utility program designed to operate under the ITS time sharing system. It enables a user to monitor a variety of aspects of the time sharing system by providing periodically updated display output or periodic printed output to teletype or line printer.

*262 A Concrete Approach To Abstract Recursive Definitions, Mitchel Wand, June 1972 (Paper presented at Symposium on the Theory of Automata Programming and Languages, I.R.I.A., Paris, July 1972).

*263 A Heterarchical Program for Recognition of Polyhedra, Yoshiaki Shirai, June 1972 (see Artificial Intelligence Vol 4, No 2) (also part of AI-TR-281 and The

Psychology of Computer Vision)

264 Developing a Musical Ear: A New Experiment, Jeanne Bamberger, July 1972. (LOGO Memo No. 6) ($.90)

This is a report on some ideas we have been developing at M.I.T. for self-paced, independent music study. The aim of our approach is to nurture in students that enigmatic quality called musical -- be it a musical ear or an individual's capacity to give a musical performance. While all of us cherish these qualities, rarely do we come to grips with them directly in teaching. More often we rely on our magical or mystical faith in the inspiration of music itself, and its great artists, to do the teaching. And for some (maybe ultimately all) this is the best course. But what about the others to whom we teach only the techniques of playing instruments or some facts about music -- its forms, its history and its apparent elements? How often do we have or take the time to examine the assumptions underlying these facts we teach, or to question the relation between what we teach and what we do as musicians?

265 Infants in Children Stories-Toward a Model of Natural Language Comprehension, Garry S. Meyer, August 1972. ($2.10)

How can we construct a program that will understand stories that children would understand? By understand we mean the ability to answer questions about that story. We are interested here with understanding natural language in a very broad area. In particular how does one understand stories about infants? We propose a system which answers such questions by relating the story to background real world knowledge. We make use of the general model proposed by Eugene Charniak in his Ph.D. thesis (Charniak 72). The model sets up expectations whih can be used to help answer questions about the story. There is a set of routines called BASE-routines that correspond to our real world knowledge and routines that are put-in which are called DEMONS that correspond to contextual information. Context can help to assign a particular meaning to an ambiguous word, or pronoun.

267 Manipulator Design Vignettes, Marvin Minsky, October 1972. ($1.30)

This memo is about mechanical arms. The literature on robotics seems to be deficient in such discussions, perhaps because not enough sharp theoretical problems have been formulated to attract interest. I'm sure many of these matters have been discussed in other literatures -- prosthetics, orthopedics, mechanical engineering, etc., and references to such discussions would be welcome. We raise these issues in the context of designing the mini-robot system in the A.I. Laboratory in 1972-1973. But we would like to attract the interest of the general heuristic programming community to such questions.

268 A Human Oriented Logic for Automatic Theorem Proving, Arthur J. Nevins, October 1972. ($1.70)

A deductive system is described which should replace the resolution principle as the primary instrument for proving theorems by computer. The system has been realized by a computer program whose performance compares favorably with the best resolution type theorem provers while at the same time it appears more consistent with the logical processes used by humans when proving theorems.

*269 Proposal to ARPA for Continued Research on A.I., Marvin Minsky, October 1972.

*270 Teaching of Procedures-Progress Report, Gerald Jay Sussman, October 1972. (Replaced by AI-TR-297)

*272 How the GAS Program Works With a Note on Simulating Turtles with Touch Sensors, Michael Speciner, December 1972.

273 The Little Robot System, David Silver, January 1973. ($.90) AD-773-929.

The Little Robot System provides for the I.T.S. user a medium size four degree of freedom six axis robot which is controlled by the PDP-6 computer through the programming language LISP. The robot includes eight force feedback channels which when interpreted by the PDP-6 are read by LISP as the signed force applied to the end of the fingers.

*274 Proposal to ARPA for Continuation of Micro-Automation Development, Marvin Minsky, January 1973.

275 Differential Peceptrons, Martin Brooks, Jerrold Ginsparg, January 1973. ($1.30) AD-773-919.

This paper shows that differential perceptrons are equivalent to perceptrons on the class of figures that fit exactly onto a sufficiently small square grid. By investigating predicates of various geometric transformations, we discover that translation and symmetry can be computed in finite order using finite coefficients in both continuous and discrete cases.

276 The Making of the Film, SOLAR CORONA, Michael Beeler, February 1973. (Film Memo No. 1.). ($.90) AD-773-928.

The film SOLAR CORONA was made from data taken from August 14, 1969 through May 7, 1970, by OSO-VI, one of the Orbiting Satellite Observatories.

277 A Linguistics Oriented Programming Language, Vaughan R. Pratt, February 1973. ($1.30) AD-773-566.

A programming language for natural language processing programs is described. Examples of the output of programs written using it are given. The reasons for various design decisions are discussed. An actual session with the system is presented, in which a small fragment of an English-to-French translator is developed. Some of the limitations of the system are discussed, along with plans for further development.

278 D-SCRIPT: A Computational Theory of Descriptions, Robert C. Moore, February 1973. ($1.30) AD-773-926.

This paper describes D-SCRIPT a language representing knowledge in artifical intelligence programs. D-SCRIPT contains a powerful formalism for descriptions, which permits the representation of statements that are problematic for other systems. Particular attention is paid to problems of opaque contexts, time contexts, and knowledge about knowledge. The design of a theorem prover for this language is also considered.

279 Pretty-Printing, Converting List to Linear Structure, Ira Goldstein, February 1973. ($1.30) AD-773-927.

Pretty-printing is the conversion of list structure to a readable format. This paper outlines the computational problems encountered in such a task and documents the current algorithm in use.

280 Elementary Geometry Theorem Proving, Ira Goldstein, April 1973. ($1.70) AD-735-568.

An elementary theorem prover for a small part of plane Euclidean geometry is presented. The purpose is to illustrate important problem solving concepts that naturally arise in building procedural models for mathematics.

282 Grammar for the People: Flowcharts of SHRDLU's Grammar, Andee Rubin, March 1973. ($1.30)

The grammar which SHRDLU uses to parse sentences is outlined in a series of flowcharts which attempt to modularize and illuminate its structure. In addition, a short discussion of systematic grammar is included.

¤284 Proposal To ARPA For Continued Research On A.I. For 1973, Marvin Minsky and Seymour Papert, June 1973. (see A.I.Memo 299)

285 The Binford-Horn LINEFINDER, B.K.P. Horn, Revised December 1973. (Originally published as A.I.Vision Flash 16, July 1971). ($.90)

This paper briefly describes the processing performed in the course of producing a line drawing from an image obtained through an image dissector camera. The edge-marking phase uses a non-linear parallel line-follower. Complicated statistical measures are not used. The line and vertex generating phases use a number of heuristics to guide the transition from edge-fragments to cleaned-up line-drawing. Higher-level understanding of the blocks-world is not used. Sample line-drawings produced by the program are included.

¤286 The FINDSPACE Problem, Gerald J. Sussman, March 1973. (Originally issued as A.I. Vision Flash 18, August 1971. See WP #113.)

¤287 Finding The Skeleton of a Brick, Tim Finin, March 1973. (Originally issued as A.I. Vision Flash 19, August 1971.)

¤288 Richard Schroeppel, Never written.

¤289 Visual Position Extraction Using Stereo Eye Systems With A Relative Rotational Motion Capability, Daniel W. Corwin, March 1973. (Originally issued as Vision Flash 22, January 1972.)

290 Paterson's Worm, Michael Beeler, June 1973. ($.90) AD-775-351.

A description of a mathematical idealization of the feeding pattern of a kind of worm is given.

292 UT: Telnet Reference Manual, Donald Eastlake, April 1974. ($.90)

UT is a user telnet program designed to run under the ITS time sharing system. It implements the relatively recent ARPA negotiating protocol for telnet.

*293 Minsky and Papert, Never written

295 On Lightness, Berthold K.P. Horn, October 1973 (see Computer Graphics and Information Processing, December, 1974). ($2.10) (AD-773-569).

The intensity at a point in an image is the product of the reflectance at the corresponding object point and the intensity of illumination at that point. We are able to perceive lightness, a quantity closely correlated with reflectance. How then do we eliminate the component due to illumination from the image on our retina? The two components of image intensity differ in their spatial distribution. A method is presented here which takes advantage of this to compute lightness from image intensity in a layered, parallel fashion.

296 An Essay on the Primate Retina, David Marr, January 1974 (see Vision Research, 1974, 14, 1377-1388). ($2.10) (AD-034-482).

This essay is considerably longer than the published version of the same theory, and is designed for readers who have only elementary knowledge of the retina. It is organized into four parts. The first is a review that consists of four sections: retinal anatomy, physiology, psychophysics, and the retinex theory. The main exposition starts with Part II, which deals with the operation of the retina in conditions of moderate ambient illumination. The account is limited to an analysis of a single cone channel -- like the red or the green one -- the rod channel being referred to frequently during the account. Part III considers various interesting properties of retinal signals, including those from the fully dark-adapted retina; and finally the thorny problem of bleaching adaptation is dealt with in Pari IV. The general flow of the account will be from the receptors to the ganglion cells, and an analysis of each of the retinal cells and synapses ius given in the appropriate place.

298 Uses of Technology to Enhance Education, Seymour Papert, June 1973 (LOGO Memo 8). ($2.60)

This paper is the substance of a proposal to the N.S.F. for support of research on children's thinking and elementary education. This work was supported by the National Science Foundation under grant GJ-1049 and conducted at the Artificial Intelligence Laboratory.

299 Proposal to ARPA for Research on Intelligent Automata and Micro-Automation, 1974-1976, September 1973. ($2.60)

This document consists of thorough descriptions of the various sections of the laboratory and their intent for the next two years.

300 Design Outline for Mini-Arms Based on Manipulator Technology, Carl R. Flatau, May 1973 (issued as A.I. Memo January 1974). ($1.70) AD-773-570.

The design of small manipulators is an art requiring proficiency in diverse disciplines. This paper documents some of the general ideas illustrated by a particular design for an arm roughly one quarter human size. The material is divided into the following sections:

A. General design constraints.
B. Features of existing manipulator technology.
C. Scaling relationships for major arm components.
D. Design of a particular small manipulator.
E. Comments on future possibilities.

301 A Mechanical Arm Control System, Richard C. Waters, January 1974. ($1.30) (AD-A004-672).

This paper describes a proposed mechanical arm control system, and some of the lines of thought which lead to the current design. It is divided into five main sections:

1. Some Ideas on Control
2. The Basic Capabilities of the Arm Controller
3. The Internal Structure of the Arm Controller
4. The Dynamic Level of the Arm Control System
5. The Procedural Level of the Arm Control System

302 A Relaxation Approach to Splitting in an Automatic Theorem Prover, Arthur Nevins, January 1974. ($1.30) (AD-A004-269).

The splitting of a problem into subproblems often involves the same variable appearing in more than one of the subproblems. This makes these subproblems dependent upon one another since a solution to one may not qualify as a solution to another. A two stage method of splitting is described which first obtains solutions by relaxing the dependency requirement and then attempts to reconcile solutions to different subproblems. The method has been realized as part of an automatic theorem prover programmed in LISP which takes advantage of the procedural power that LISP provides.

303 Plane Geometry Theorem Proving Using Forward Chaining, Arthur Nevins, January 1974. ($1.30) AD-A004-223.

A computer program is described which operates on a subset of plane geometry. Its performance not only compares favorably with previous computer programs, but within its limited problem domain (e.g. no curved lines or introduction of new points), it also compares favorably with the best human theorem provers. The program employs a combination of forward and backward chaining with the forward component playing the more important role.

304 Acceleration of Series, William Gosper, March 1974. (AD-A011837). ($2.10)

The convergence of infinite series can be accelerated by a suitable splitting of each term into two parts and then combining the second part of the n-th term with the first part of the (n+1)-th term to get a new series and leaving the first part of the first term as an "orphan." Repeating this process an infinite number of times, the series will often approach zero, and we obtain the series of orphans, which may converge faster than the original series. heuristics for determining the splits are given. Various mathematical constants, orginally defined as series having a term ratio which approaches 1, are accelerated into series having a term ratio less than 1.

*305 Summary of MYCROFT: A System for Understanding Simple Picture

Programs, Ira Goldstein, May 1974. (Replaced by AI-TR-294) AD-A004-671.

306 A Framework for Representing Knowledge, Marvin Minsky, June 1974. (AD-A011-168). ($2.10). (In Psychology of Computer Vision)

This is a partial theory of thinking, combining a number of classical and modern concepts from psychology, linguistics, and A. I. Whenever one encounters a new situation (or makes a substantial change in one's viewpoint) he selects from memory a structure called a frame: a remembered framework to be adapted to fit reality by changing details as necessary.

307 LLOGO: An Implementation of LOGO in LISP, Ira Goldstein, June 1974. (LOGO Memo No. 11) ($1.70) AD-A004-669.

This paper describes LLOGO, an implementation of the LOGO language written in MACLISP for the ITS, TEN50 and TENEX PDP-10 systems, and MULTICS. The relative merits of LOGO and LISP as educational languages are discussed. Design decisions in the LISP implementation of LOGO are contrasted with those of two other implementations: CLOGO for the PDP-10 and 11LOGO for the PDP-11, both written in assembler language. LLOGO's special facilities for character-oriented display terminals, graphic display "turtles", and music generation are also described.

308 Force Feedback in Precise Assembly Tasks, Hirochika Inoue, August 1974. (AD-A011-369). ($1.30)

This paper describes the execution of precise assembly tasks by a robot. The level of performance of the experimental system allows such basic actions as putting a peg into a hole, screwing a nut on a bolt and picking up a thin piece from a flat table. The tolerance achieved in the experiments was 0.001 inch. The experiments proved that force feedback enabled the reliable assembly of a bearing complex consisting of eight parts with close tolerances. A movie of the demonstration is available.

309 Commenting Proofs, James R. Geiser, August 1974. ($.90). (AD-A011-838).

This paper constitutes a summary of a seminar entitled "Commenting Proofs" given at the A.I. Lab at M.I.T. during the Spring of 1974. The work is concerned with new syntactic structures in formal proofs which derive from their pragmatic and semantic aspects. It is a synthesis of elements from Yessenin-Volpin's foundational studies and developments in Artificial Intelligence concerned with commenting programs and the use of this idea in automatic debugging procedures.

311 TORTIS: Toddler's Own Recursive Turtle Interpreter System, Radia Perlman, December 1974. (LOGO Memo No. 9) ($1.30)

TORTIS is a device for preschool children to communicate with and program the turtle. It consists of several boxes (currently 3 button boxes and two blox boxes) designed so that only a few new concepts are introduced at a time but more can be added when the child becomes familiar with what he has. Hopefully transitions are gradual enough so that the child never thinks talking to the turtle is too hard or that he is "too dumb." And hopefully playing with the system should teach such concepts as numbers, breaking large problems into small solvable steps, writing and debugging procedures, recursion, variables, and conditionals. Most important of all, it should teach that learning is fun.

312 The Luxury of Necessity, Jeanne Bamberger, December 1974. (LOGO Memo No.12) ($1.30)

This paper was originally written as an address to a conference of the National Association of Schools of Music on "The Music Consumer". Posing a series of questions which point to fundamental issues underlying the LOGO music project, the paper goes on to describe some of the specific projects with which students have been working in an effort to probe these issues. Emphasis is placed on "modes of representation" as a significant realm of enquiry: just how does an individual represent a tune to himself, what are the differences between formal and informal modes of representation--what features and relations of a melody does a representation capture, what does it leave out? What is the influence of such modes of "perception", how do they effect strategies of problem solving, notions of "same" and "different" or even influence musical "taste"? Finally, there are some hints at what might constitute "sufficiently powerful representations" of musical design with examples from both simple and complex pieces of music as well as a probe into what might distinguish "simple" from "complex" musical designs.

313 LOGO Manual, Hal Abelson, Nat Goodman, Lee Rudolph, December 1974. (LOGO Memo 7) ($2.10)

This document describes the LOGO system implemented for the PDP 11/45 at the M.I.T. Artificial Intelligence Laboratory. The "system" includes not only the LOGO evaluator, but also a dedicated time-sharing system which services about a dozen users. There are also various special devices such as robot turtles, tone generators, and CRT displays.

314 What's In A Tune, Jeanne Bamberger, December 1974. (LOGO Memo 13) ($1.30)

The work reported here began with two fundamental assumptions: 1) The perception of music is an active process; it volves the individual in selecting, sorting, and grouping the features of the phenomena before her. 2) Individual differences in response to a potentially sensible melody rest heavily on just which features the individual has access to or is able to focus on.

¤315 A Glossary of LOGO Primitives, Hal Abelson, Jim Adams, September 1974. (LOGO Memo 14).

315A A Glossary of PDP11 LOGO Primitives, E. Paul Goldenberg, March, 1975. (LOGO Memo 16). ($1.30).

This glossary was written for the purpose of providing a quick and concise yet accurate descripion of the primitives and special words and characters of the March 18, 1975 PDP 11 implementation of the LOGO language. Many entries include references to other related words and/or examples of the use of the primitive being described, but this is not intended to replace the function of a good manual. For a more detailed and comprehensive description of the language, see AIM 313 LOGO Manual (LOGO Memo 7)

319 Localization of Failures in Radio Circuits - A Study in Causal and Teleological Reasoning, Allen Brown, Gerald Sussman, December 1974. (AD-A011-839).($1.30)

This paper examines some methodologies for diagnosing correctly designed radio circuits which are failing to perform in the intended way because of some faulty component. Particular emphasis is placed on the utility and necessity of good teleological descriptions in successfully executing the task of isolating failing components.

320 Velocity Space and the Geometry of Planetary Orbits, Harold Abelson, Andrea diSessa, Lee Rudolph, December 1974. (LOGO Memo 15) ($1.30)

We develop a theory of orbits for the inverse-square central force law which differs considerably from the usual deductive approach. In particular, we make no explicit use of calculus. By beginning with qualitative aspects of solutions, we are led to a number of geometrically realizable physical invariants of the orbits. Consequently most of our theorems rely only on simple geometrical relationships. Despite its simplicity, our planetary geometry is powerful enough to treat a wide range of perturbations with relative ease. Furthermore, without introducing any more machinery, we obtain full quantitative results. The paper concludes with suggestions for further research into the geometry of planetary orbits.

321 Model-Driven Geometry Theorem Prover, Shimon Ullman, May 1975. ($.70). ADA-021446.

This paper describes a new Geometry Theorem Prover, which was implemented to illuminate some issues related to the use of models in theorem proving. The paper is divided into three parts: Part 1 describes the G.T.P. and presents the ideas embedded in it. It concentrates on the forward search method, and gives two examples of proofs produced that way. Part 2 describes the backward search mechanism, and presents proofs to a sequence of successively harder problems. The last section of the work addresses the notion of similarity in a problem, defines a notion of semantic symmetry, and compares it to Gelernter's concept of syntactic symmetry.

322 A Frame for Frames: Representing Knowledge for Recognition, Benjamin J. Kuipers, March 1975. (AD-A012-835). ($1.30).

This paper presents a version of frames suitable for representing knowledge for a class of recognition problems. An initial section gives an intuitive model of frames, and illustrates a number of desirable features of such a representation. A more technical example describes a small recognition program for the Blocks World which implements some of these features. The final section discusses the more general significance of the representation and the recognition process used in the example.

323 Orienting Silicon Integrated Circuit Chips for Lead Bonding, Berthold K. P. Horn, January 1975. (In Computer Graphics and Information Processing, September, 1975). ($0.90). (ADA-021137).

Will computers that see and understand what they see revolutionize industry by automating the part orientation and part inspection processes? There are two obstacles: the expense of computing and our feeble understanding of images. We believe these obstacles are fast ending. To illustrate what can be done we describe a working program that visually determines the position and orientation of silicon chips used in integrated circuits.

¤324 On the Purpose of Low-level Vision, David Marr, December 1974. (AD-A012-392).

¤325 The Low-level Symbolic Representation of Intensity Changes in an Image, David Marr, December 1974. (AD-A013-669).

326 The Recognition of Sharp Closely Spaced Edges, David Marr, December 1974. (AD-A013-090). ($0.90).

The recognition of sharp edges from edge- and bar-mask convolutions with an image is studied for the special case where the separation of the edges is of the order of the masks' panel-width. Desmearing techniques are employed to separate the items in the image. Attention is also given to parsing de-smeared mask convolutions into edges and bars; to detecting edge and bar terminations; and to the detection of small blobs.

327 A Note on the Computation of Binocular Disparity in a Symbolic, Low-level Visual Processor, David Marr, December 1974. (AD-A012-393). ($0.90).

The goals of the computation that extracts disparity from pairs of pictures of a scene are defined, and the constraints imposed upon that computation by the three-dimensional structure of the world are shown to be inadequate. A precise expression of the goals of the computation is possible in a low-level symbolic visual processor; the constraints translate in this environment to pre-requisites on the binding of disparity values to low-level symbols. The outline of a method based on this is given.

328 Heuristic Techniques in Computer Aided Circuit Analysis, Gerald Jay Sussman and Richard Matthew Stallman, March 1975. ($1.30). (ADA-021171).

We present EL, a new kind of circuit analysis program. Whereas other circuit analysis systems rely on classical, formal analysis techniques, EL employs heuristic "inspection" methods to solve rather complex DC bias circuits. These techniques also give EL the ability to explain any result in terms of its own qualitative reasoning processes. EL's reasoning is based on the concept of a "local one-step deduction" augmented by various "teleological" principles and by the concept of a "macro-element". We present several annotated examples of EL in operation and an explanation of how it works. We also show how EL can be extended in several directions, including sinusoidal steady state analysis. Finally, we touch on possible implications for engineering education. We feel that EL is significant not only as a novel approach to circuit analysis ut also as an application of Artificial Intelligence techniques to a new and intersting domain.

329 Parsing Intensity Profiles, Tomas Lozano-Perez, May 1975. ($1.30). (ADA-021172).

Much low-level vision work in AI deals with one-dimensional intensity profiles. This paper describes PROPAR, a system that allows a convenient and uniform mechanism for recognizing such profiles. PROPAR is a modified Augmented Transition Networks parser. The grammar used by the parser serves to describe and label the set of acceptable profiles. The input to the parser are descriptions of segments of a piecewise linear approximation to an intensity profile. A sample grammar is presented and the results discussed.

330 A Computational View Of The Skill Of Juggling, Howard Austin, December 1974. ($1.70).

This research has as its basic premise the belief that physical and mental skills are highly similar, enough so in fact that computation paradigms such as the ones used in Artificial Intelligence research about predominantly mental skills can be usefuly extended to include physical skills. This thesis is pursued experimentally by the categorization of "juggling bugs" via detailed video observations. A descriptive language for juggling movements is developed and a taxonomy of bugs is presented. The remainder of the paper is concerned with an empirical determination of the characteristics of an ultimate theory of juggling movements. The data presented is relevant to the computational issues of control structure, naming, addressing and subprocedurization.

331 Thesis Progress Report: A System For Representing and Using Real-World Knowledge, Scott E. Fahlman, May 1975. ($2.10). (ADA-021178).

This paper describes progress to date in the development of a system for representing various forms of real-world knowledge. The knowledge is stored in the form of a net of simple parallel processing elements, which allow certain types of deduction and set-intersection to be performed very quickly and easily. It is claimed that this approach offers definite advantages for recognition and many other data-accessing tasks. Suggestions are included for the application of this system as a tool in vision, natural-language processing, speech recognition, and other problem domains.

332 Ideas about Management of Data Bases, Erik Sandewll, May 1975. (AD-A013-312). ($1.30).

The paper advocates the need for systems which support maintenance of LISP-type data bases, and describes an experimental system of this kind, called DABA. In this system, a description of the data base's structure is kept in the data base itself. A number of utility programs use the description for operations on the data base. The description must minimally include syntactic information reminiscent of data structure declarations in more conventional programming languages, and can be extended by the user.

333 On the Detection of Light Sources, Shimon Ullman, May 1975. ($0.90).

The paper addresses the following problem: Given an array of light intensities obtained from some scene, find the light sources in the original scene. The following factors are discussed from the point of view of their relevance to light sources detection: The highest intensity in the scene, absolute intensity value, local and global contrast, comparison with the average intensity, and lightness computation. They are shown to be insufficient for explaining humans' ability to identify light sources in their visual field. Finally, a method for accomplishing the source detection task in the mondrian world is presented.

*334 Analyzing Natural Images: a Computational Theory of Texture Vision, David

Marr, June 1975. (AD-A013-257).

335 Image Intensity Understanding, Berthold K. P. Horn, August 1975. ($2.10). (ADA-021135).

Image intensities have been processed traditionally without much regard to how they arise. Typically they are used only to segment an image into regions or to find edge-fragments. Image intensities do carry a great deal of useful information about three-dimensional aspects of objects and some initial attempts are made here to exploit this. An understanding of how images are formed and what determines the amount of light reflected from a point on an object to the viewer is vital to such a development. The gradient-space, popularized by Huffman and Mackworth is a helpful tool in this regard.

336 Teaching Teachers Logo, The Lesley Experiments, Howard Austin, April 1976. ($1.30).

This research is concerned with the question of whether or not teachers who lack specialized backgrounds can adapt to and become proficient in the technically complex, philosophically sophisticated LOGO learning environment. Excellent results were obtained and are illustrated through a series of examples of student work. The report then gives some brief observations about the thought styles observed and concludes with suggestions for further work.

337 Artificial Intelligence, Language and the Study of Knowledge, Ira Goldstein and Seymour Papert, July 1975, Revised March 1976. ($2.10).

This paper studies the relationship of Artificial Intelligence to the study of language and the representation of the underlying knowledge which supports the comprehension process. It develops the view that intelligence is based on the ability to use large amounts of diverse kinds of knowledge in procedural ways, rather than on the possession of a few general and uniform principles. The paper also provides a unifying thread to a variety of recent approaches to natural language comprehension. We conclude with a brief discussion of how Artificial Intelligence may have a radical impact on education if the principles which it utilizes to explore the representation and use of knowledge are made available to the student to use in his own learning experiences. This paper is a revised version of an earlier document written with Marvin Minsky. Many of the ideas in this paper owe much to Minsky's thoughtful critique; the authors, however take responsibility for the organization and wording of this document.

338 The Art of Snaring Dragons, Harvey A. Cohen, November 1974, revised May 1975. (LOGO Memo 18). ($2.10).

DRAGONS are formidable problems in elementary mechanics not amenable to solution by naive formula cranking. What is the intellectual weaponry one needs to snare a Dragon? To snare a Dragon one brings to mind an heuristic frame -- a specifically structured association of problem solving ideas. Data on the anatomy of heuristic frames - just how and what ideas are linked together - has been obtained from the protocols of many attacks on Dragons by students and physicists. In this paper various heuristic frames are delineated by detailing how they motviate attacks on two particular Dragons, Milko and Jugglo, from the writer's compilation. This model of the evolution of problem solving skills has also been applied to the interpretation of the intellectual growth of children, and in an Appendix we use it to give a cogent interpretation for the protocols of Piagetian "Conservation" experiments. The model provides a sorely needed theoretical framework to discuss teaching strategems calculated to promote problem solving skills.

339 Very Large Planner-type Data Bases, Drew V. McDermott, September 1975. ($1.70).

This paper describes the implementation of a typical data-base manager for an AI language like Planner, Conniver, or QA4, and some proposed extensions for applications involving greater quantities of data than usual. The extensions are concerned with data bases involving several active and potentially active sub-data-bases, or"contexts". The major mechanisms discussed are the use of contexts as packets of data with free variables; and indexing data according to the contexts they appear in. The paper also defends the Planner approach to data representation against some more recent proposals.

340 Early Processing of Visual Information, D. Marr, December 1975. ($2.10).

The article describes a symbolic approach to visual information processing, and sets out four principles that appear to govern the design of complex symbolic information processing systems. A computational theory of early visual information processing is presented, which extends to about the level of figure-ground separation. It includes a process-oriented theory of texture vision. Most of the theory has been implemented, and examples are shown of the analysis of several natural images. This replaces Memos 324 and 334.

341 Spatial Disposition of Axes in a Generalized Cylinder Representation of Objects that Do Not Encompass the Viewer, D. Marr and K. Nishihara, December 1975. ($1.70). AD-A023456.

It is proposed that the 3-D representation of an object is based primarily on a stick-figure configuration, where each stick represents one or more axes in the object's generalized cylinder representation. The loosely hierarchical description of a stick-figure is interpreted by a special purpose processor, able to maintain two vectors and the gravitational vertical relative to a Cartesian space-frame. It delivers information about the appearance of these vectors, which helps the system to rotate its model into the correct 3-D orientation relative to the viewer during recognition.

342 The Development of Musical Intelligence I: Strategies for Representing Simple Rhythms, Jeanne Bamberger, November 1975. ($2.10). (LOGO Memo 19).

This paper describes two distinct and contrasting strategies which individuals use for making sense of simple rhythmic figures. The strategies are characterized by the particular features of the figures which each captures. The distinctions between the two strategies are significant to the general development of musical intelligence and may also suggest implications for learning and teaching in other domains as well. It is the thesis of the paper that while individuals tend to favor one strategy or the other, both are necessary for intelligent musical behavior. I will suggest further that it is the dynamic interaction of the two strategies which leads to the development of significant new knowledge and to creative learning and performance in the domain of music and perhaps in other disciplines, too. Finally, I will propose the theory that traditional schooling tends to emphasize one strategy over the other and in any case encourages the tendency to keep them separate rather than stimulating their interaction.

343 Leading a Child to a Computer Culture, Cynthia Solomon, December 1975. (LOGO Memo #20). ($0.90).

344 How Near Is Near? A Near Specialist, Murray Elias Denofsky, February 1976. ($2.10).

This paper presents a system for understanding the concept of near and far, weighing such factors as purpose of judgement, dimensions of the objects, absolute size of the distance, and size of the distance relative to other objects, ranges, and standards. A further section discusses the meaning of phrases such as very near, much nearer than, and as near as. Although we will speak of near as a judgement about physical distance, most of the ideas developed will be applicable to any continuous measurable parameter, such as size or time. An adaptation for rows (discrete spaces) is made as well.

348 Turtle Escapes the Plane: Some Advanced Turtle Geometry, Andy diSessa, December 1975. ($1.30). (LOGO Memo 21).

Since the LOGO Turtle took his first step he has been mathematically confined to running around on flat surfaces. Fortunately the physically intuitive, procedurally oriented nature of the Turtle which makes him a powerful explorer in the plane is equally, if not more apparent when he is liberated to tread on curved surfaces. This paper is aimed roughly at the High School level. Yet because it is built on intuition and physical action rather than formalism, it can reach such "graduate school" mathematical ideas as geodesics, Gaussian Curvature, and topological invariants as expressed in the Gauss-Bonnet Theorem.

349 An Interpreter for Extended Lambda Calculus, Gerald J. Sussman and Guy L. Steele, Jr., December 1975. ($1.70).

Inspired by ACTORS (Greif and Hewitt) (Smith and Hewitt), we have implemented an interpreter for a LISP-like language, SCHEME, based on the lambda calculus (Church), but extended for side effects, multiprocessing, and process synchronization. The purpose of this implementation is tutorial. We wish to: (1) alleviate the confusion caused by Micro-PLANNER, CONNIVER, etc. by clarifying the embedding of non-recursive control structures in a recursive host language like LISP;(2) explain how to use these control structures, independent of such issues as pattern matching and data base manipulation; (3) have a simple concrete experimental domain for certain issues of programming semantics and style.

351 A State Space Model for Sensorimotor Control and Learning, Marc Raibert, January 1976. ($1.30).

This is the first of a two part presentation of a model which deals with problems of motor control, motor learning, adaptation, and sensorimotor integration. In this section the problems are outlined and a solution is given which makes use of a state space memory and a piece-wise linearization of the equations of motion. A forthcoming companion article will present the results of tests performed on an implementation of the model.

353 Lambda: The Ultimate Imperative, Guy Lewis Steele, Jr. and Gerald Jay Sussman, March, 1976. ($1.70).

We demonstrate how to model the following common programming constructs in terms of an applicative order language similar to LISP: Simple recursion; iteration; compound statements and expressions; GO TO and assignment; continuation-passing; escape expressions; fluid variables; call by name, call by need, and call by reference. The models require only (possibly self-referent) lambda application, conditionals, and (rarely) assignment. No complex data structures such as stacks are used. The models are transparent, involving only local syntactic transformations. Some of these models, such as those for GO TO and assignment, are already well known, and appear in the work of Landin, Reynolds, and others. The models for escape expressions, fluid variables, and call by need with side effects are new. This paper is partly tutorial in intent, gathering all the models together for purposes of context.

355 Artificial Intelligence -- a personal view, David Marr, March 1976. ($0.90).

The goal of A.I. is to identify and solve useful information processing problems. In so doing, two types of theory arise. Here, they are labelled Types 1 and 2, and their characteristics are outlined. This discussion creates a more than usually rigorous perspective of the subject, from which past work and future prospects are briefly reviewed.

356 Logo Progress Report 1973-1975, H. Abelson, J. Bamberger, I. Goldstein, and S. Papert, September 1975, Revised March 1976. (LOGO Memo #22). ($1.30).

Over the past two years, the Logo Project has grown along many dimensions. This document provides an overview in outline form of the main activities and accomplishments of the past as well as the major goals guiding our current research. Research on the design of learning environments, the corresponding development of a theory of learning and the exploration of teaching activities in these environments is presented.

357 From Understanding Computation to Understanding Neural Circuitry, D. Marr and T. Poggio, May 1976. ($1.30).

The CNS needs to be understood at four nearly independent levels of description: (1) that at which the nature of a computation is expressed; (2) that at which the algorithms that implement a computation are characterized; (3) that at which an algorithm is committed to particular mechanisms; and (4) that at which the mechanisms are realized in hardware. In general, the nature of a computation is determined by the problem to be solved, the mechanisms that are used depend upon the available hardware, and the particular algorithms chosen depend on the problem and on the available mechanisms. Examples are given of theories at each level.

358 The Text-Justifier TJ6, Joseph D. Cohen, May 1976. ($1.70).

This memo, intended as both a reference and user's manual describes the text-justifying program TJ6, which compiles a neat output document from a sloppy input manuscript. TJ6 can justify and fill text; automatically number pages and figures; control page format and indentation; underline, superscript, and subscript; print a table of contents; etc.

359 Spatial Knowledge, Benjamin Kuipers, June 1976. ($1.70).

This paper introduces a model of spatial cognition to describe the states of partial knowledge that people have about the spatial structure of a large-scale environment. Spatial knowledge has several different representations, each of which captures one aspect of the geography. With knowledge stored in multiple representations, we must examine the procedures for assimilating new information, for solving problems, and for communicating information between representations. The model centers on an abstract machine called the TOUR machine, which executes a description of the route to drive the "You Are Here" pointer (a small working memory) through a map that describes the geography. Representations for local and global spatial knowledge are discussed in detail. The model is compared with a survey of the psychological literature. Finally, the directions of necessary and desirable future research are outlined.

360 Using Computer Technology to Provide a Creative Learning Environment for Preschool Children, Radia Perlman, May 1976. ($1.30).

TORTIS is a system of special terminals together with software which is designed to provide programming capability and be accesible for use by very young children. The system is designed to add capabilities in small increments so that the child is never overwhelmed by too much to learn at one time, and maintains a feeling of control over the environment. This system facilitates learning of various concepts such as relative size of numbers, frames of reference, procedures, conditionals, and recursion, but more importantly it teaches good problem solving techniques and a healthy approach to learning.

BOOKS IN PRINT

Computation, Marvin Minsky, Prentice Hall, 1967.

Perceptrons, Marvin Minsky and Seymour Papert, MIT Press, 1968.

Semantic Information Processing, Marvin Minsky (Ed.) MIT Press, 1968.

Counter-Free Automata, Seymour Papert and Robert McNaughton, MIT Press, 1971.

Understanding Natural Language, Terry Winograd, Academic Press, 1972.

A Computer Model of Skill Acquisition, Gerald Sussman, American Elsevier, February 1975.

The Psychology of Computer Vision, Patrick H. Winston (Ed.), McGraw-Hill, 1975.

TECHNICAL REPORTS

Please note: All technical reports accompanied by an 'AD' number may be obtained from the following sources:

Government contractors may obtain AI-TR reports from: Defense Documentation Center, Cameron Station, Alexandria, Va. 22314. Specify AD number.

Others may obtain the reports from: National Technical Information Service, Operations Division, Springfield, Va. 22151. Specify AD number. Prices differ for microfiche and paper documents.

¢AI-TR-219 Bobrow, Daniel G., Natural Language Input for a Computer Problem Solving Language, June 1964. (MAC-TR-1) AD-604-730. (In Semantic Information Processing)

¢AI-TR-220 Raphael, Bertram, SIR: A Computer Program for Semantic Information Retrieval, June 1964. (MAC-TR-2) AD-608-499. (In Semantic Information Processing)

☼AI-TR-221 (Thesis) Teitelman, Warren, PILOT: A Step Toward Man-Computer Symbiosis, Sept. 1966. (MAC-TR-32) AD-638-446.

☼AI-TR-222 (Thesis) Norton, Lewis M., ADEPT: A Heuristic Program for Proving Theorems of Group Theory, Oct. 1966. (MAC-TR-33) AD-645-660.

☼AI-TR-223 (Thesis) Martin, William A., Symbolic Mathematical Laboratory, Jan. 1967. (MAC-TR-36) AD-657-283.

☼AI-TR-224 (Thesis) Guzman-Arenas, Adolfo, Some Aspects of Pattern Recognition by Computer, Feb. 1967. (MAC-TR-37) AD-656-041.

☼AI-TR-225 Forte, Allen, Syntax-Based Analytic Reading of Musical Scores, April 1967. (MAC-TR-39) AD-661-806.

☼AI-TR-226 (Thesis) Moses, Joel, Symbolic Integration, Dec. 1967. (MAC-TR-47) AD-662-666.

☼AI-TR-227 (Thesis) Charniak, Eugene, CARPS: A Program Which Solves Calculus Word Problems, July 1968. (MAC-TR-51) AD-673-670.

☼AI-TR-228 (Thesis) Guzman-Arenas, Adolfo, Computer Recognition of Three-Dimensional Objects in a Visual Scene, Dec. 1968. (MAC-TR-59) AD-692-200.

☼AI-TR-229 (Thesis) Beyer, Wendell Terry, Recognition of Topological Invariants by Iterative Arrays, Oct. 1969. (MAC-TR-66) AD-699-502.

AI-TR-230 (Thesis) Griffith, Arnold K., Computer Recognition of Prismatic Solids, Aug. 1970. (MAC-TR-73) AD-711-763. ($3.25)

An investigation is made into the problem of constructing a model of the appearance to an optical input device of scenes consisting of plane-faced geometric solids. The goal is to study algorithms which find the real straight edges in the scene, taking into account smooth variations in intensity over faces of the solids, blurring of edges, and noise.

☼AI-TR-231 (Thesis) Winston, Patrick H., Learning Structural Descriptions From Examples, Sept. 1970. (MAC-TR-76) AD-713-988.

☼AI-TR-232 (Thesis) Horn, Berthold K.P., Shape From Shading: A Method for Obtaining the Shape of a Smooth Opaque Object From One View, Nov. 1970. (MAC-TR-79) AD-717-336. (In The Psychology of Computer Vision).

☼AI-TR-233 (Thesis) Banks, Edwin Roger, Information Processing and Transmission in Cellular Automata, Jan. 1971. (MAC-TR-81)

AI-TR-234 (Thesis) Krakauer, Lawrence J., Computer Analysis of Visual Properties of Curved Objects, May 1971. (MAC-TR-82) AD-723-647. ($2.25)

A method is presented for the visual analysis of objects by computer.It is particularly well suited for opaque objects with smoothly curved surfaces. The method extracts information about the object's surface properties, including measures of its specularity, texture, and regularity. It also aids in determining the object's shape.

☼AI-TR-235 (Thesis) Winograd, Terry, Procedures as a Representation for Data in a Computer Program for Understanding Natural Language, February 1971. (MAC-TR-84) AD-721-399.

Available in book form under the title: Understanding Natural Language, Terry Winograd, Academic Press (New York) 1972. In Britain and Europe: Edinburgh University Press, 1972.

☼AI-TM-236 (Thesis) Jones, Thomas L., A Computer Model of Simple Forms of Learning, Jan. 1971. (MAC-TR-20). AD-720-337.

AI-TR-242 (Thesis) Smoliar, Stephen W., A Parallel Processing Model of Musical Structures, September 1971. (MAC-TR-91) AD-731-690. ($3.25)

EUTERPE is a real-time computer system for the modeling of musical structures. It provides a formalism wherein familiar concepts of musical analysis may be readily expressed. This is verified by its application to the analysis of a wide variety of conventional forms of music; Gregorian chant, Mediaeval polyphony, Bach counterpoint, and sonata form. It may be of further assistance in the real-time experiments in various techniques of thematic development. Finally, the system is endowed with sound synthesis apparatus with which the user may prepare tapes for musical performances.

☼AI-TR-258 (Thesis) Hewitt, Carl, Description and Theoretical Analysis (Using Schemata) of PLANNER: A Language for Proving Theorems and Manipulating Models In A Robot, April 1972. AD-744-620.

☼AI-TR-266 (Thesis) Charniak, Eugene, Toward A Model Of Children's Story Comprehension, December 1972. AD-755-232. ($4.25)

☼AI-TR-271 (Thesis) Waltz, David L., Generating Semantic Descriptions From Drawings of Scenes With Shadows, November 1972. AD-754-080. (In The Psychology of Computer Vision)

☼AI-TR-281 Winston, Patrick H., Editor, Progress In Vision And Robotics, May 1973. AD-775-439.

AI-TR-283 (Thesis) Fahlman, Scott E., A Planning System For Robot Construction Tasks, May 1973. AD-773-471. ($3.25)

This paper describes BUILD, a computer program which generates plans for buildings specified structures out of simple objects such as toy blocks. A powerful heuristic control structure enables BUILD to use a number of sophisticated construction techniques in its plans. Among the final design, pre-assembly of movable sub-structures on the table, and the use of extra blocks as temporary supports and counterweights in the course of the construction.

AI-TR-291 (Thesis) McDermott, Drew V., Assimilation of New Information by a Natural Language-Understanding System, February 1974. AD-780-194. ($3.25).

This work describes a program, called TOPLE, which uses a procedural model of the world to understand simple declarative sentences. It accepts sentences in a modified predicate calculus symbolism, and uses plausible reasoning to visualize scenes, resolve ambiguous pronoun and noun phrase references, explain in events, and make conditional predications. Because it does plausible deduction, with tentative conclusions, it must contain a formalism for describing its reasons for its conclusions and what the alternaives are.

AI-TR-294 (Thesis) Goldstein, Ira P., Understanding Simple Picture Programs, April 1974. (AD-A005-907). ($3.50).

What are the characteristics of the process by which an intent is transformed into a plan and then a program? How is a program debugged? A system has been designed to investigate these questions in the context of understanding simple turtle programs for drawing pictures. The system called MYCROFT generates commentary to describe the effects, plan and anomalies of a turtle program when compared to a model of the intended picture. Repair of the program is then based upon a combination of general debugging technique and specific repair knowledge associated with geometric model primitives.

☼AI-TR-297 (Thesis) Sussman, Gerald J., A Computational Model of Skill Acquisition, August 1973. (In A Computer Model of Skill Acquisition)

☼AI-TR-310 Winston, Patrick H., New Progress in Artificial Intelligence, September 1974. AD-A002-272.

AI-TR-316 (Thesis) Rubin, Ann D., Hypothesis Formation and Evaluation in Medical Diagnosis, January 1975 ($4.25).

This thesis describes some aspects of a computer system for doing medical diagnosis in the specialized field of kidney disease. A four-step process which consists of disposing of findings, activating hypotheses, evaluating hypotheses locally and combining hypotheses globally is examined for its heuristic implications. The work attempts to fit the problem of medical diagnosis into the framework of other Artificial Intelligence problems and paradigms and in particular explores the notions of pure search vs. heuristics methods, linearity and interaction, local vs. global knowledge, and the structure of hypotheses within the world of kidney disease.

AI-TR-346 (Thesis) John M. Hollerbach, Hierarchical Shape Description of Objects by Selection and Modification of Prototypes, November 1975. ($3.35).

An approach towards shape description, based on prototype modification and generalized cylinders, has been developed and applied to the object domains pottery and polyhedra: 1. A program describes and identifies pottery from vase outlines entered as lists of points. The descriptions have been modeled after descriptions by archeologists, with the result that identifications made by the program as remarkably consistent with those of the archeologists. It has been possible to quantify their shape descriptors, which are everyday terms in our language applied to many sorts of objects besides pottery, so that the resulting descriptions seem very natural. 2. New parsing strategies for polyhedra overcome some limitations of previous work. A special feature is that the processes of parsing and identification are carried out simultaneously. With this descriptive approach, the evidently unrelated domains of pottery and polyhedra are treated similarly. Objects are segmented into multiple generalized cylinders. The cylinders are then described by assigning a prototype, a standard shape from a small repertoire, which is modified to conform more exactly with the cylinder. The modifications are structured hierarchically and specify the degree of modification as coarsely or precisely as desired. Some modifications are specific to a given prototype, others are applicable to several of them. The emphasis throughout this work has been to develop useful, qualitative descriptions which bring out the significant features and subordinate lesser ones. To this purpose curved lines representing the boundary of vases have been quantized into a few curvature levels. Line, region, and volume shapes are all described by assigning and modifying prototypes. In each instance the prototypes are specialized to the domain, and pose diferent problems in selection and modification.

AI-TR-347 (Thesis) Robert Carter Moore, Reasoning from Incomplete Knowledge in a Procedural Deduction System, December 1975. ($2.35).

One very useful idea in AI research has been the notion of an explicit model of a problem situation. Procedural deduction languages, such as PLANNER, have been valuable tools for building these models. But PLANNER and its relatives are very limited in their ability to describe situations which are only partially specified. This thesis explores methods of increasing the ability of procedural deduction systems to deal with incomplete knowledge. The thesis examines in detail, problems involving negation, implication, disjunction, quantification, and equality. Control structure issues and the problem of modelling change under incomplete knowledge are also consider. Extensive comparisons are also made with systems for mechanical theorem proving.

AI-TR-352 (Thesis) Johan de Kleer, Qualitative and Quantitative Knowledge in Classical Mechanics, December 1975. ($2.35). (ADA-021515).

This thesis investigates what knowledge is necessary to solve mechanics problems. A program NEWTON is described which understands and solves problems in a mechanics mini-world of objects moving on surfaces. Facts and equations such as those given in a mechanics text need to be presented. However, this is far from sufficient to solve problems. Human problem solvers rely on "commonsense" and "qualitative" knowledge which the physics text tacitly assumes to be present. A mechanics problem solver must embody such knowledge. Quantitative knowledge given by equations and more qualitative common sense knowledge are the major research points exposited in this thesis. The major issue in solving prolems is planning. Planning involves tentatively outlining a possible path to the solution without actually solving the problem. Such a plan needs to be constructed and debugged in the process of solving the problem. Envisionment, or qualitative simulation of the event, plays a central role in this planning process.

order form

name

address

memo	price	memo	price

amount enclosed

RETURN TO: Ms. Mary Castro, MIT AI Lab, NE43-800E,
545 Technology Square, Cambridge, MA 02139