```
************************************************
************************************************
***                                          ***
***  Name:                                   ***
***                                          ***
***  Project:    1       Programmer:  MWK    ***
***                                          ***
***  File Name: AIMS[BIB,DOC]                ***
***                                          ***
***  File Last Written:  10:49 31 May 1973   ***
***                                          ***
***  Time: 19:54         Date: 15 Jul 1973   ***
***                                          ***
***            Stanford University           ***
***      Artificial Intelligence Project     ***
***        Computer Science Department       ***
***           Stanford,  California          ***
***                                          ***
************************************************
************************************************
```

```
COMMENT ⊗   VALID 00003 PAGES
C REC  PAGE   DESCRIPTION
C00001 00001
C00002 00002                         ARTIFICIAL INTELLIGENCE MEMOS
C00040 00003                                     1973
C00052 ENDMK
C⊗;
```

ARTIFICIAL INTELLIGENCE MEMOS

These memos report research results. Abstracts of memos published in 1972 and later are listed here. Interested researchers may obtain available copies upon request to:

Artificial Intelligence Laboratory Secretary
Computer Science Department
Stanford University
Stanford, California 94305

Alternatively, they are available from:

National Technical Information Service
Springfield, Virginia 22151

The National Technical Information Service charges $3.00 per full size copy and $.95 for a microfiche copy.

1972

AIM-159, J.A. Feldman and P. C. Shields, TOTAL COMPLEXITY AND INFERENCE OF BEST PROGRAMS, April 1972.

Axioms for a total complexity measure for abstract programs are presented. Essentially, they require that total complexity be an unbounded increasing function of the Blum time and size measures. Algorithms for finding the best program on a finite domain are presented, and their limiting behavior for infinite domains described. For total complexity, there are important senses in which a machine can find the best program for a large class of functions.

AIM-160, Jerome A. Feldman, AUTOMATIC PROGRAMMING, February 1972, 20 pages.

The revival of interest in Automatic Programming is considered. The research is divided into direct efforts and theoretical developments and the successes and prospects of each are described.

AIM-161, Yorick Wilks, ARTIFICIAL INTELLIGENCE APPROACH TO MACHINE TRANSLATION, February 1972, 44 pages.

The paper describes a system of semantic analysis and generation, programmed in LISP 1.5 and designed to pass from paragraph length input in English to French via an interlingual representation. A wide class of English input forms will be covered, but the vocabulary will initially be restricted to one of a few hundred words. With this subset working, and during the current year (1971-72), it is also hoped to map the interlingual representation onto some predicate calculus notation so as to make possible the answering of very simple questions about the translated matter. The specification of the translation system itself is complete, and its main points are:
i) It translates phrase by phrase--with facilities for reordering phrases and establishing essential semantic connectivities between them--by mapping complex semantic stuctures of "message" onto each phrase. These constitute the interlingual representation to be translated. This matching is done without the explicit use of a conventional syntax analysis, by taking as the appropriate matched structure the "most dense" of the alternative structures derived.

This method has been found highly successful in earlier versions of this analysis system.

ii) The French output strings are generated without the explicit use of a generative grammar. That is done by means of STEREOTYPES: strings of French words, and functions evaluating to French words, which are attached to English word senses in the dictionary and built into the interlingual representation by the analysis routines. The generation program thus receives an interlingual representation that already contains both French output and implicit procedures for assembling the output, since the stereotypes are in effect recursive procedures specifying the content and production of the output word strings. Thus the generation program at no time consults a word dictionary or inventory of grammar rules.

It is claimed that the system of notation and translation described is a convenient one for expressing and handling the items of semantic information that are ESSENTIAL to any effective MT system. I discuss in some detail the semantic information needed to ensure the correct choice of output prepositions in French; a vital matter inadequately treated by virtually all previous formalisms and projects.

AIM-162, Roger C.Schank, Neil Goldman, Charles J. Rieger, Christopher K. Riesbeck, PRIMITIVE CONCEPTS UNDERLYING VERBS OF THOUGHT, April 1972, 102 pages.

In order to create conceptual structures that will uniquely and unambiguously represent the meaning of an utterance, it is necessary to establish 'primitive' underlying actions and states into which verbs can be mapped. This paper presents analyses of the most common mental verbs in terms of such primitive actions and states. In order to represent the way people speak about their mental processes, it was necessary to add to the usual ideas of memory structure the notion of Immediate Memory. It is then argued that there are only three primitive mental ACTs.

AIM-163, J.M. Cadiou, RECURSIVE DEFINITIONS OF PARTIAL FUNCTIONS AND THEIR COMPUTATIONS, April 1972, 160 pages.

A formal syntactic and semantic model is presented for 'recursive definitions' which are generalizations of those found in LISP, for example. Such recursive definitions can have two classes of fixpoints, the strong fixpoints and the weak fixpoints, and also possess a class of computed partial functions.

Relations between these classes are presented: fixpoints are shown to be extensions of computed functions. More precisely, strong fixpoints are shown to be extensions of computed functions when the computations may involve "call by name" substitutions; weak fixpoints are shown to be extensions of computed functions when the computation only involve "call by value" substitutions. The Church-Rosser property for recursive definitions with fixpoints also follows from these results.

Then conditions are given on the recursive definitions to ensure that they possess least fixpoints (of both classes), and computation rules are given for computing these two fixpoints: the "full" computation rule, which leads to the least weak fixpoint. A general class of computation rules, called 'safe innermost', also lead to the latter fixpoint. The "leftmost innermost" rule is a special case of those, for the LISP recursive definitions.

AIM-164, Zohar Manna and Jean Vuillemin, FIXPOINT APPROACH TO THE THEORY OF COMPUTATION, April 1972, 29 pages.

Following the fixpoint theory of Scott, we propose to define the semantics of computer programs in terms of the least fixpoints of recursive programs. This allows one not only to justify all existing verification techniques, but also to extend them to handle various properties of computer programs, including correctness, termination and equivalence, in a uniform manner.

AIM-165, D.A. Bochvar, TWO PAPERS ON PARTIAL PREDICATE CALCULUS, April 1972, 50 pages.

These papers, published in 1938 and 1943, contain the first treatment of a logic of partial predicates. Bochvar's treatment is of current interest for two reasons. First, partial predicate and function logic are important for mathematical theory of computation because functions defined by programs or by recursion cannot be guaranteed to be total. Second, natural language may be better approximated by a logic in which some sentences may be undefined than by a conventional logic. Bochvar use of his system to avoid Russell's paradox is of interest here, and in partial predicate logic it may be possible to get more of an axiomatization of truth and knowledge than in a conventional logic.

The papers translated are "On a three-valued logical calculus and its application to the analysis of contradictions", Recueil Mathematique, N. S. 4 (1938), pp. 287-308, and "On the consistency of a three-valued logical calculus", ibid. 12 (1943), pp. 353-369.

We also print a review and a correction by Alonzo Church that appeared in the Journal of Symbolic Logic. The review was in vol. 4.2 (June 1939), p. 99, and the additional comment was in vol. 5.3 (September 1940), p. 119.

AIM 166, Lynn H. Quam, Sidney Liebes, Jr., Robert B. Tucker, Marsha Jo Hannah, Botond G. Eross, COMPUTER INTERACTIVE PICTURE PROCESSING, April 1972, 40 pages.

This report describes work done in image processing using an interactive computer system. Techniques for image differencing are described and examples using images returned from Mars by the Mariner Nine spacecraft are shown. Also described are techniques for stereo image processing. Stereo processing for both conventional camera systems and the Viking 1975 Lander camera system is reviewed.

AIM-167, Ashok K. Chandra, EFFICIENT COMPILATION OF LINEAR RECURSIVE PROGRAMS, June 1972, 43 pages.

We consider the class of linear recursive programs. A linear recursive program is a set of procedures where each procedure can make at most one recursive call. The conventional stack implementation of recursion requires time and space both proportional to n, the depth of recursion. It is shown that in order to implement linear recursion so as to execute in time n one doesn't need space proportional to n: n**ε for sufficiently small ε will do. It is also known that with constant space one can implement linear recursion in time n**2. We show that one can do much better: n**(1+ε) for arbitrarily small ε. We also describe an algorithm that lies between these two: it takes time n.log (n) and space log (n).

It is shown that several problems are closely related to the linear recursion problem, for example, the problem of reversing an input tape given a finite automaton with several one-way heads. By casting all these problems into canonical form, efficient solutions are obtained simultaneously for all.

AIM-168, Shigeru Igarashi, ADMISSIBILITY OF FIXED-POINT INDUCTION IN FIRST-ORDER LOGIC OF TYPED THEORIES, May 1972, 40 pages.

First-order logic is extended so as to deal with typed theories, especially that of continuous functions with fixed-point induction formalized by D. Scott. The translation of his formal system, or the λ calculus-oriented system derived and implemented by R. Milner, into this logic amounts to adding predicate calculus features to them.

In such a logic the fixed-point induction axioms are no longer valid, in general, so that we characterize formulas for which Scott-type induction is applicable, in terms of syntax which can be checked by machines automatically.

Diskfile: Aim168.igr[aim,doc]

AIM-169, Robin Milner, LOGIC FOR COMPUTABLE FUNCTIONS. DESCRIPTION OF A MACHINE IMPLEMENTATION, May 1972, 36 pages.

This paper is primarily a user's manual for LCF, a proof-checking program for a logic of computable functions proposed by Dana Scott in 1969, but unpublished by him. We use the name LCF also for the logic itself, which is presented at the start of the paper. The proof-checking program is designed to allow the user interactively to generate formal proofs about computable functions and functionals over a variety of domains, including those of interest to the computer scientist--for example, integers, lists and computer programs and their semantics. The user's task is alleviated by two features: a subgoaling facility and a powerful simplification mechanism. Applications include proofs of program correctness and in particular of compiler correctness; these applications are not discussed herein, but are illustrated in the papers referenced in the introduction.

Diskfile: Lcfman.rgm[aim,doc]

AIM-170, Yorick Wilks, LAKOFF ON LINGUISTICS AND NATURAL LOGIC, June 1972, 19 pages.

The paper examines and criticizes Lakoff's notions of a natural logic and of a generative semantics described in terms of logic, I argue that the relationship of these notions to logic as normally understood is unclear, but I suggest, in the course of the paper, a number of possible interpretations of his thesis of generative semantics. I argue further that on these interpretations a mere notational variant of Chomskyan theory. I argue, too, that Lakoff's work may provide a service in that it constitutes a reductio ad absurdum of the derivational paradigm of modern linguistics; and shows, inadvertently, that only a system with the ability to reconsider its own inferences can do the job that Lakoff sets up for linguistic enquirey -- that is to say, only an "artificial intelligence" system.

Diskfile: Lakoff.Yaw[aim,doc]

AIM-171, Roger Schank, ADVERBS AND BELIEF, June 1972, 30 pages.

The treatment of a certain class of adverbs in conceptual representation is given. Certain adverbs are shown to be representative of complex belief structures. These adverbs serve as pointers that explain where the sentence that they modify belongs in a belief structure.

AIM-172, Sylvia Weber Russell, SEMANTIC CATEGORIES OF NOMINALS FOR
CONCEPTUAL DEPENDENCY ANALYSIS OF NATURAL LANGUAGE, July 1972,
64 pages.

A system for the semantic categorization of conceptual objects (nominals) is provided. The system is intended to aid computer understanding of natural language. Specific implementations for "noun-pairs" and prepositional phrases are offered.

AIM-173, Gerald Jacob Agin, REPRESENTATION AND DESCRIPTION OF CURVED
OBJECTS, October 1972, 134 pages.

Three dimensional images, similar to depth maps, are obtained with a triangulation system using a television camera, and a deflectable laser beam diverged into a plane by a cylindrical lens.

Complex objects are represented as structures joining parts called generalized cylinders. These primitives are formalized in a volume representation by an arbitrary cross section varying along a space curve axis. Several types of joint structures are discussed.

Experimental results are shown for the description (building of internal computer models) of a handful of complex objects, beginning with laser range data from actual objects. Our programs have generated complete descriptions of rings, cones, and snake-like objects, all of which may be described by a single primitive. Complex objects, such as dolls, have been segmented into parts, most of which are well described by programs which implement generalized cylinder descriptions.

AIM-174, Francis Lockwood Morris, CORRECTNESS OF TRANSLATIONS OF
PROGRAMMING LANGUAGES -- AN ALGEBRAIC APPROACH, August 1972,
124 pages.

Programming languages and their sets of meanings can be modelled by general operator algebras; semantic functions and compiling functions by homomorphisms of operator algebras. A restricted class of individual programs, machines, and computations can be modelled in a uniform manner by binary relational algebras. A restricted class of individual manner by binary relational algebras. These two applications of algebra to computing are compatible: the semantic function provided by interpreting ("running") one binary relational algebra on another is a homomorphism on an operator algebra whose elements are binary relational algebras.

Using these mathematical tools, proofs can be provided systematically of the correctness of compilers for fragmentary programming languages each embodying a single language "feature". Exemplary proofs are given for statement sequences, arithmetic expressions, Boolean expressions, assignment statements, and while statement. Moreover, proofs of this sort can be combined to provide (synthetic) proofs for in principle, many different complete programming languages. One example of such a synthesis is given.

AIM-175, Hozumi Tanaka, HADAMARD TRANSFORM FOR SPEECH WAVE ANALYSIS, August 1972, 34 pages.

Two methods of speech wave analysis using the Hadamard transform are discussed. The first method is a direct application of the Hadamard transform for speech waves. The reason this method yields poor results is discussed. The second method is the application of the Hadamard transform to a log-magnitude frequency spectrum. After the application of the Fourier transform the Hadamard transform is applied to detect a pitch period or to get a smoothed spectrum. This method shows some positive aspects of the Hadamard transform for the analysis of a speech wave with regard to the reduction of processing time required for smoothing, but at the cost of precision. A formant tracking program for voiced speech is implemented by using this method and an edge following technique used in scene analysis.

AIM-176, J. A. Feldman, J. R. Low, D. C. Swinehart, R. H. Taylor RECENT DEVELOPMENTS IN SAIL. AN ALGOL BASED LANGUAGE FOR ARTIFICIAL INTELLIGENCE, November 1972, 22 pages.

New features added to SAIL, an ALGOL based languaged for the PDP-10, are discussed. The features include: procedure variables; multiple processes; coroutines; a limited form of backtracking; an event mechanism for inter-process communication; and matching procedures, a new way of searching the LEAP associative data base.

AIM-177, Richard Paul, MODELLING, TRAJECTORY CALCULATION AND SERVOING OF A COMPUTER CONTROLLED ARM, November 1972, 89 pages.

The problem of computer contral of an arm is divided into four parts: modelling, trajectory calculation, servoing and control.

In modelling we use a symbolic data structure to represent objects in the environment. The program considers how the hand may be positioned to grasp these objects and plans how to turn and position them in order to make various moves. An arm model is used to calculate the configuration-dependent dynamic properties of the arm before it is moved.

The arm is moved along time-coordinated space trajectories in which velocity and acceleration are controlled. Trajectories are calculated for motions along defined space curves, as in turning a crank; in such trajectories various joints must be free due to external motion constraints.

The arm is servoed by a small computer. No analog servo is used. The servo is compensated for gravity loading and for configuration-dependent dynamic properties of the arm.

In order to control the arm, a planning program interprets symbolic arm control instructions and generates a plan consisting of arm motions and hand actions.

The move planning program has worked successfully in the manipulation of plane faced objects. Complex motions, such as locating a bold and screwing a nut onto it, have also been performed.

AIM-178, Aharon Gill, VISUAL FEEDBACK AND RELATED PROBLEMS IN COMPUTER CONTROLLED HAND EYE COORDINATION, October 1972, 130 pages.

A set of programs for precise manipulation of simple planar bounded objects, by means of visual feedback, was developed for use in the Stanford hand-eye system. The system includes a six degrees of freedom computer controlled manipulator (arm and hand) and a fully instrumented computer television camera.

The image of the hand and manipulated objects is acquired by the computer through the camera. The stored image is analyzed using a corner and line finding program developed for this purpose. The analysis is simplified by using all the information available about the objects and the hand, and previously measured coordination errors. Simple touch and force sensing by the arm help the determination of three dimensional positions from one view.

The utility of the information used to simplify the scene analysis depends on the accuracy of the geometrical models of the camera and arm. A set of calibration updating techniques and programs was developed to maintain the accuracy of the camera model relative to the arm model.

The precision obtained is better than .1 inch. It is limited by the resolution of the imaging system and of the arm position measuring system.

AIM-179, Bruce G. Baumgart, WINGED EDGE POLYHEDRON REPRESENTATION, October 1972, 46 pages.

A winged edge polyhedron representation is stated and a set of primitives that preserve Euler's F-E+V=2 equation are explained. Present use of this representation in Artificial Intelligence for computer graphics and world modeling is illustrated and its intended future application to computer vision is described.

AIM-180, Ruzena Bajcsy, COMPUTER IDENTIFICATION OF TEXTURED VISUAL SCENES, October 1972, 156 pages.

This work deals with computer analysis of textured outdoor scenes involving grass, trees, water and clouds. Descriptions of texture are formalized from natural language descriptions; local descriptors are obtained from the directional and non-directional components of the Fourier transform power spectrum. Analytic expressions are obtained for orientation, contrast, size, spacing, and in periodic cases, the locations of texture elements. These local descriptors are defined over windows of various sizes; the choice of sizes is made by a simple higher-level program.

The process of region growing is represented by a sheaf-theoretical model which formalizes the operation of pasting local structure (over a window) into global structure (over a region). Programs were implemented which form regions of similar color and similar texture with respect to the local descriptors.

An interpretation is made of texture gradient as distance gradient in space. A simple world model is described. An interpretation of texture regions and texture gradient is made with a simulated correspondence with the world model. We find that a problem-solving approach, involving hypothesis-verification, more satisfactory than an earlier pattern recognition effort (Bajcsy 1970) and more crucial to work with complex scenes than in scenes of polyhedra. Geometric clues from relative sizes, texture gradients, and interposition are important in interpretation.

AIM-181, Bruce G. Buchanan, REVIEW OF HUBERT DREYFUS' WHAT COMPUTERS CAN'T DO: A CRITIQUE OF ARTIFICIAL REASON (Harper & Row, New York, 1972), November 1972, 14 pages.

The recent book "What Computers Can't Do" by Hubert Dreyfus is an attack on artificial intelligence research. This review takes the position that the philosophical content of the book is interesting, but that the attack on artificial intelligence is not well reasoned.

AIM-182, Kenneth Mark Colby and Franklin Dennis Hilf, CAN EXPERT JUDGES, USING TRANSCRIPTS OF TELETYPED PSYCHIATRIC INTERVIEWS, DISTINGUISH HUMAN PARANOID PATIENTS FROM A COMPUTER SIMULATION OF PARANOID PROCESSES? December,1972, 10 pages.

Expert judges (psychiatrists and computer scientists) could not correctly distinguish a simulation model of paranoid processes from actual paranoid patients.

1973

AIM-183, Roger C. Schank, THE FOURTEEN PRIMITIVE ACTIONS AND THEIR INFERENCES, March 1973, 70 pages.

In order to represent the conceptual information underlying a natural language sentence, a conceptual structure has been established that uses the basic actor-action-object framework. It was the intent that these structures have only one representation for one meaning, regardless of the semantic form of the sentence being represented. Actions were reduced to their basic parts so as to effect this. It was found that only fourteen basic actions were needed as building blocks by which all verbs can be represented. Each of these actions has a set of actions or states which can be inferred when they are present.

AIM-184, Malcolm Newey, AXIOMS AND THEOREMS FOR INTEGERS, LISTS AND FINITE SETS IN LCF, January 1973, 53 pages.

LCF (Logic for Computable Functions) is being promoted as a formal language suitable for the discussion of various problems in the Mathematical Theory of Computation (MTC). To this end, several examples of MTC problems have been formalised and proofs have been exhibited using the LCF proof-checker. However, in these examples, there has been a certain amount of ad-hoc-ery in the proofs; namely many mathematical theorems have been assumed without proof and no axiomatisation of the mathematical domains involved was given. This paper describes a suitable mathematical environment for future LCF experiments and its axiomatic basis. The environment developed deemed appropriate for such experiments, consists of a large body of theorems from the areas of integer arithmetic, list manipulation and finite set theory.

AIM-185, Ashok K.Chandra and Zohar Manna, ON THE POWER OF PROGRAMMING FEATURES, January 1973, 29 pages.

We consider the power of several programming features such as counters, pushdown stacks, queues, arrays, recursion and equality. In this study program schemas are used as the model for computation. The relations between the powers of these features is completely described by a comparison diagram.

AIM-186, Robin Milner, MODELS OF LCF, January 1973, 17 pages.

LCF is a deductive system for computable functions proposed by D. Scott in 1969 in an unpublished memorandum. The purpose of the present paper is to demonstrate the soundness of the system with respect to certain models, which are partially ordered domains of continuous functions. This demonstration was supplied by Scott in his memorandum; the present paper is merely intended to make this work more accessible.

AIM-187, George E. Collins, THE COMPUTING TIME OF THE EUCLIDEAN

ALGORITHM, January 1973, 17 pages.

The maximum, minimum and average computing times of the classical Euclidean algorithm for the greatest common divisor of two integers are derived, to within codominance, as functions of the lengths of the two inputs and the output.

AIM-188, Ashok K. Chandra, ON THE PROPERTIES AND APPLICATIONS OF PROGRAM SCHEMAS, March 1973, 231 pages.

The interesting questions one can ask about program schemas include questions about the "power" of classes of schemas and their decision problems viz. halting divergence, equivalence, etc. We first consider the powers of schemas with various features: recursion, equality tests, and several data structures such as pushdown stacks, lists, queues and arrays. We then consider the decision problems for schemas with equality and with commutative and invertible functions. Finally a generalized class of schemas is described in an attempt to unify the various classes of uninterpreted and semi-interpreted schemas and schemas with special data structures.

AIM-189, James Gips and George Stiny, AESTHETICS SYSTEMS,January 1973, 22 pages.

The formal structure of aesthetics systems is defined. Aesthetics systems provide for the essential tasks of interpretation and evaluation in aesthetic analysis. Kolmogorov's formulation of information theory is applicable. An aesthetics system for a class of non-representational, geometric paintings and its application to three actual paintings is described in the Appendix.

AIM-190, Malcolm Newey, NOTES ON A PROBLEM INVOLVING PERMUTATIONS AS SEQUENCES, March 1973, 20 pages.

The problem (attributed to R. M. Karp by Knuth (see #36 of [1])) is to describe the sequences of minimum length which contain, as subsequences, all the permutations of an alphabet of n symbols. This paper catalogs some of the easy observations on the problem and proves that the minimum lengths for n=5, n=6 and n=7 are 19, 28, and 39 respectively. Also presented is a construction which yields (for n>2) many appropriate sequences of length n*-2n+4 so giving an upper bound on length of minimum strings which matches exactly all known values.

AIM-192, George E. Collins and Ellis Horowitz, THE MINIMUM ROOT SEPARATION OF A POLYNOMIAL, April 1973, 13 pages.

The minimum root separation of a complex polynomial A is defined as the minimum of the distances between distinct roots of A. For polynomials with Gaussian integer coefficients and no multiple roots, three lower bounds are derived for the root separation. In each case the bound is a function of the degree, n, of A and the sum, d, of the absolute values of the coefficients of A. The notion of a semi-norm for a commutative ring is defined, and it is shown how any semi-norm can be extended to polynomial rings and matrix rings, obtaining a very general analogue of Hadamard's determinant theorem.

AIM-193, Kenneth Mark Colby, THE RATIONALE FOR COMPUTER BASED TREATMENT OF LANGUAGE DIFFICULTIES IN NONSPEAKING AUTISTIC CHILDREN, March 1973, 13 pages.

The principles underlying a computer-based treatment method for language acquisition in nonspeaking autistic children are described. The main principle involves encouragement of exploratory learning with minimum adult interference.

AIM-194, Kenneth Mark Colby and Franklin Dennis Hilf,MULTIDIMENSIONAL ANALYSIS IN EVALUATING A SIMULATION OF PARANOID THOUGHT, May 1973, 10 pages.

The limitations of Turing's Test as an evaluation procedure are reviewed. More valuable are tests which ask expert judges to make ratings along multiple dimensions essential to the model. In this way the model's weaknesses become clarified and the model builder learns where the model must be improved.

AIM-196, Neil M. Goldman and Christopher K. Riesbeck, A CONCEPTUALLY BASED SENTENCE PARAPHRASER, May 1973, 88 pages.

This report describes a system of programs which perform natural language processing based on an underlying language free (conceptual) representation of meaning. This system is used to produce sentence paraphrases which demonstrate a form of understanding with respect to a given context. Particular emphasis has been placed on the major subtasks of language analysis (mapping natural language into conceptual structures) and language generation (mapping conceptual structures into natural language), and on the interaction between these processes and a conceptual memory model.