

GRADUATE STUDIES COMMITTEE  
Dr. David A. Gustafson - Chairperson  
Dr. William J. Hankley  
Dr. David A. Schmidt  
Joseph K. Campbell - Administrative Assistant

January 1987

GUIDELINES  
FOR THE  
DOCTOR OF PHILOSOPHY DEGREE  
IN THE  
DEPARTMENT OF COMPUTING AND  
INFORMATION SCIENCES  
KANSAS STATE UNIVERSITY

## I. INTRODUCTION

These guidelines set forth departmental and university standards for attainment of a Doctor of Philosophy (PhD) Degree in Computer Science. You are expected to adhere to these standards. If exceptions are warranted, your advisor must be consulted to determine legitimate, alternate means of meeting the standards. Additional information is published every semester in the "Student Guide for Masters and Doctoral Degrees," which is available from the Graduate School Office and in the "Graduate Student Handbook," published by the Graduate Student Council.

The guidelines stated herein are those of the faculty of the Computer Science Department at KSU. Certain other regulations are imposed by the KSU Graduate School and are described in the "Graduate Faculty Handbook." It is the your responsibility to know and satisfy all relevant requirements. It is the responsibility of the Graduate Studies Committee of the Department of Computer Science at KSU to oversee the application of these guidelines.

The Graduate Studies Committee promises to keep you informed of the committee's view of your progress towards the graduate degree. In keeping with this commitment, an annual review of all graduate students is performed each January, and an evaluation of each student's progress is written. This evaluation is transmitted to the student and a copy is placed in his permanent and open record.

The primary overseer and guide for your PhD program is your adviser — first a temporary *academic adviser* and then your *research adviser*, who is identified as the Major Professor on your supervisory committee. Upon beginning your PhD studies, you are assigned an academic advisor. Approximately one year into your studies, you should locate a professor who agrees to be your research advisor. (These stages are discussed in detail in Section III.)

The PhD program in Computer Science is offered jointly by Kansas State University in Manhattan and the University of Kansas in Lawrence. Acceptance into the program implies acceptance by both Computer Science Departments. You may elect to fulfill residency and other requirements at either school. You may select courses from offerings at both schools. Your degree is awarded by the school that employs your research adviser.

## II. GENERAL REQUIREMENTS

The PhD normally requires at least three years of full-time graduate study beyond the bachelor's degree, credited as at least 90 semester hours (typically distributed as 30 hours of Master's work, 30 hours of PhD level coursework and 30 hours of PhD research, culminating in a PhD dissertation). All work must be completed within seven years. At least one full year must be spent in residence. A student usually starts in the Master's program and receives a Master's degree. A student is then admitted into the PhD program as a *PhD student*. Upon successful completion of the comprehensive examination, the student becomes a *PhD aspirant*. Each aspirant must complete preliminary examinations and submit a research proposal to become a *PhD candidate*. A candidate must submit and defend a dissertation to complete the degree.

Formal writing is considered an integral part of your PhD program. If it becomes evident

that you need to improve your writing skills, you will be required to complete English 516, Written Communication for Scientists, with associated English 030, Writing Laboratory, in addition to other requirements.

You are expected to participate in the professional activities of the Department. Thus, you should attend seminars and colloquia offered by the Department and by the professional societies within the Departments. Sometime in your graduate career you must participate in teaching within the Department. We will welcome your suggestions for improvements in the curriculum, research, and functioning of the Department.

You must maintain a 3.00 overall grade point average in all coursework and continue to evidence regular progress toward completion of the degree. Progress of graduate students is reviewed regularly by the Graduate Studies Committee. Any student who does not sustain adequate progress will receive written warning from the Computer Science Department. Any student who does not maintain an adequate grade point average or who does not respond to a warning of inadequate progress will be placed on probation, with written notice from the Graduate School. A student on probation must correct deficiencies within the time limit indicated in the written notice or be dismissed from the graduate program.

### III. SPECIFIC REQUIREMENTS FOR THE PHD DEGREE

In the usual case, you must first complete a Master's degree in Computer Science or a related field. Then, you must apply for admission as a PhD student in the computer science department, and you must be accepted. (See the booklets "Directions for Applying for Graduate Study in Computer Science" and "Guidelines for the Master of Science Degree in Computer Science at Kansas State University.") Your Master's degree must include work equivalent to the following core courses:

- CMPSC671 Programming Science (includes symbolic logic, program specification, and verification)
  - CMPSC700 Compiler Design (includes grammars and automata)
  - CMPSC720 Operating Systems (includes concurrent programming and queuing models)
  - CMPSC740 Software Engineering (includes lifecycle models and tools)
  - CMPSC761 Data Base Systems (includes formal relations and graph models)
- The core courses prepare you for taking the *comprehensive examination*, which covers the breadth of the core courses. (In special cases, a copy of a Master's examination and results taken at another university may be submitted as a substitute. The Graduate Studies Committee must approve the substitution.)

*You are required to take the comprehensive exam by the end of your first semester in the PhD program.* (You are encouraged to take the exam even earlier, at the end of your MS studies, if at all possible). Extensions of the time limit may be made for students entering the program from other universities if they have not had similar core courses. The request for an extension must be made before the end of the first semester.

If you fail the comprehensive exam, you must retake it and pass it the following semester. A third try is not allowed.

When you pass the comprehensive exam, you become a PhD aspirant. Obtain a *Declaration of Intent* form from the Graduate Studies secretary in the Computer Science Department and complete it. Once you return the form, you are officially admitted to the joint PhD program.

At the beginning of your second year of PhD studies, you must find a research adviser. Your research adviser must be a member of the Graduate Faculty. (See the KSU General Catalog for further information.) Since the research adviser organizes and directs your research, you should choose an advisor carefully. How do you find an advisor? Shop around. Talk to faculty members. Take some of the 800-level or 900-level research-oriented courses. Read current survey and research papers in computer science journals and magazines, and look at previous dissertations. *It is your responsibility to obtain a research adviser.* No faculty member is obligated to accept you as an advisee; the selection process requires mutual consent. If you have not obtained a research adviser by the end of the second year of your PhD studies, you will receive written warning from the Computer Science Department and will be placed on probation.

Once you have obtained a research adviser, work with your adviser to plan a *Program of Study*. (Obtain the Program of Study forms from the Graduate School.) The Program of Study contains the following information:

-major professor (that is, the research advisor)

-members of the supervisory committee (see the paragraph below)

-general area of research

-the three preliminary examination areas (see the paragraph below)

-all graduate course credits (at least 90 hours)

The graduate course credits must include the following:

-the KSU MS core courses (unless equivalent courses were taken at another institution. The Graduate Studies Committee reserves the right to determine equivalence.)

-at least three 900-level courses

-at least 30 hours of PhD research

-at least 24 hours of course credit at KSU beyond the Master's degree.

At the same time that you and your research advisor formulate your Program of Study, you also select a *Supervisory Committee*. Your supervisory committee must include at least four members of the Graduate Faculty. One member must be from the graduate faculty of the Computer Science Department at the University of Kansas. One member must be a Graduate Faculty member from a department other than Computer Science. This member should be chosen for his appropriateness to your research topic.

You and your advisor must also plan your preliminary exams. The exams must be taken at the end of your second year of PhD studies. You must submit a written request to take the exams one full semester in advance. You must also apply to the Graduate School for permission to schedule the exams. You may not take the exams until the Graduate School issues the ballot form.

The preliminary examinations are offered as a set of three written examinations and may include an additional oral examination. The examinations are taken in areas supportive of your research area. Preparation for each exam must consist of at least a 700-level—800-level course sequence. At most one of these areas may be supported by readings and topics courses.

Each of the three exams is prepared by an examination committee. You will be informed of the members of the examination committees for your exams. Because of the dynamic nature of the computer science field and the variation in the content of advanced level courses, the scope of your examinations may not match exactly the scope of your preparatory courses. For this reason, after you schedule your examinations you must meet with your examination committees and obtain reading lists that identify the scope of each of the examinations.

The three preliminary examinations must be taken within a two week (that is, fifteen day) period. The format of the examinations vary. They may be closed-book (typically two to four hours in length) or open-book (one or two days' work).

The exams are graded by the respective examination committees. An exam may be graded as pass, fail, or conditional pass subject to remedial work. *You must not fail any exam.* If only one exam is marked fail, you must retake that exam the subsequent semester and pass it; if you fail it, you fail the preliminary exams and must leave the program. If you fail two or more exams, you must retake and pass all three exams the subsequent semester. Otherwise, you must leave the program.

The results of the preliminary exams are reviewed and approved by the Department's Faculty. You are notified in writing of the results.

After you have passed the preliminary exams, work with your adviser to prepare a written *research proposal* of your PhD thesis research. You must present it successfully to your supervisory committee in an open seminar. Your proposal defines your research topic and explains the significance and contribution of your work. It also serves as a contract for the direction and scope of your work. Upon the approval of your proposal, you are admitted as a PhD candidate. At this point the Graduate School designates an additional person outside of the Computer Science Department, who will serve as the Chairman for your final oral examination.

You must work closely with your advisor on your research. Each year, you must present a research colloquium that documents your progress. You are required to be a candidate for a period of at least seven months. Upon completion of your research work, you must write a *dissertation*. Each member of the supervisory committee must receive a copy of the dissertation and sign the Dissertation Approval Form, which you obtain from the Graduate School. (A member's signature does not indicate approval of the content, but merely receipt of a draft.) The committee members must be allowed at least two weeks to read your dissertation. (Don't forget to give a copy to the Chairman of your Final Examination Committee.)

You must schedule your oral presentation with the Graduate School. You cannot proceed with the presentation until the Graduate School issues the ballot form. Your presentation must be in an open seminar, but during the actual defense only the Final Examination Committee will be present. The committee will vote to pass or fail. If you pass the exam,

you submit the required dissertation copies, fees, and address information to the Graduate School. See the Graduate School for details.

A maximum of seven years is allowed from when you begin PhD course work until when you pass the oral examination.

Any issues not covered in this document will be resolved by the Graduate Studies Committee and the Computer Science Faculty.

FACULTY RESUMES

PAUL S. FISHER

Professor

Department of Computer Science

Kansas State University

Manhattan, Kansas 66506

Office Tel. (913) 532-6350

Education:

1963

BA

Mathematics

University of Utah

1964

MA

Mathematics

University of Utah

1969

PhD

Computer Science

Arizona State University

Teaching and Interests:

Languages and Language Implementation

Data Base Management System

Computer Architecture

Distributed Processing

Networking

Distributed Data Base Management Systems

Special Purpose-Functional Architecture

Automatic Generation of Data Base Systems and Application Software

Grants:

1. NSF Regional Conference 1970
2. Applicability of the Extensible Programming System (EPS) to General Software Problems 1975
3. Research into and Development of a Low Cost Hardware Monitor 1974-1975
4. Hardware Error Detection and Fault Isolation by Software 1974-1975
5. Reliable Software Workshop 1974-1975
6. Back-end Mini-Computers: Usability and Feasibility 1974-1975
7. Portability Issues for Mini-Computers using IDMS, A Data Base Management System 1975
8. Heterogeneous Mini-Computer Network with Distributed Data Base Resources 1976-1977
9. Software Portability Issues 1976-1977
10. Investigation of Mini Networks 1978
11. Communication Techniques 1976
12. A Back-end DBMS Communication System 1976-1977
13. Data, Its Internal and External Form 1976-1977
14. A Network Processing System 1978
15. Automatic Generation of Data Base System 1980
16. Full Screen Editor: An Experiment in Machine Independent Software 1981



Publications (papers and reports):

- Fisher, P. S., "A Mini Operating System," ACM-SIGPLAN Symposium, Pedagogic Languages with Small Computers, January 6-7, 1972.
- McDonough, V., P. S. Fisher, R. Weinberg, "Use of Computer Simulation for Designing a Dual-Processing, Time-Sharing, Mini-Computer System," ACM-SIGPLAN Symposium, Pedagogic Languages with Small Computers, January 6-7, 1972.
- Ahmed, N., P. S. Fisher, "Study of Algorithmic Properties of Chebyshev Coefficients," International Journal of Computer Mathematics, Vol. 2, 1970, pp. 307-317.
- Ahmed, N., P. S. Fisher, R. R. Rao, "On a Criteria for Data Compression and Binary Fourier Representation," presented Midwest Symposium on Circuit Theory, May 7-8, 1970.
- Fisher, P. S., E. E. Kohlbecker, "A Generalized Fibonacci Sequence," Fibonacci Quarterly, Vol. 10, 1972, pp. 337-344.
- Skidmore, E. L., P. S. Fisher, N. P. Woodruff, "Wind Erosion Equation: Computer Solution and Application," Soil Science Society of America Proceedings, Vol. 34, No. 5, November-December 1975.
- Fisher, P. S., W. J. Hankley, V. E. Wallentine, "Separation of Introductory Programming and Language Instruction," 4th ACM-SIGCSE Annual Conference, February 1973.
- Cassing, D., P. S. Fisher, R. James, "A Software Virtual Memory System for a Mini-Computer," Second Annual Computer Conference, Austin, Texas, November 1973.
- Fisher, P. S., "Virtual Memory: A New Horizon for Mini Computers," 9th IEEE Computer Society International Conference, Washington, D. C., September 1974.
- Hankley, W., P. S. Fisher, "A Vertical Assertion Facility," Third Annual Computer Conference, Austin, Texas, November 1974.
- Maryanski, F. J., P. S. Fisher, "A Mini Computer Distributed Data Base Management System," NBS-IEEE Trends and Applications Symposium: Micro and Mini Systems, May 1976, pp. 113-117.
- Maryanski, F. J., P. S. Fisher, and V. E. Wallentine, "Evaluation of Conversion to a Back-End DBMS, ACM National Conference, October 1976, pp. 293-297.
- Maryanski, F. J., P. S. Fisher, "Rollback and Recovery in Distributed Data Base Management Systems," Proc. ACM Annual Conference, October 1977.
- Maryanski, F. J., P. S. Fisher, V. E. Wallentine, M. A. Calhoun, "Distributed Data Base Management Using Mini Computers," INFOTECH State of the Art Report Minis Versus Main Frames, 1978.

- Stonim, J., P. S. Fisher, "Survey of Mini Data Base Management System in 1977," Proceedings First Symposium on Small Systems, ACM, New York, August 1978.
- Stonim, J., E. A. Unger, P. S. Fisher, "Data Base Management System Environments Present and Future," ACM SIGSMALL 79 Second Annual Symposium on Small Systems, October 1979, Dallas, TX.
- Maryanski, F. J., P. S. Fisher, R. Housh, D. Schmidt, "A Prototype Distributed DBMS," Proceedings of 12th Hawaii International Conference in Systems Sciences, January 1979.
- Fisher, P. S., J. Stonim, D. A. Schmidt, "Consideration for Determining the Degree of Centralization or De-centralization in the Computing Environment," Information and Management, Vol. 2, No. 1, February 1979.
- Maryanski, F. J., P. S. Fisher, V. E. Wallentine, "Data Access in Distributed Data Base Management Systems," Information and Management, Vol. 2, No. 6, December 1979.
- Su, S. Y. W., H. Chang, G. Copeland, P. S. Fisher, E. Lowenthal, S. Shuster, "Data Base Machines and Some Issues on DBMS Standards," Proceedings NCC, Vol. 49, May 1980, also published in Tutorial: Data Base Management in the 80's, J. A. Larson, H. A. Freeman, IEEE Publishers, 1981.
- Fisher, P. S., J. Stonim, "Software Engineering: An Example of Misuse," Software Practice and Experience, Vol. II, No. 6, June 1981.
- Fisher, P. S., P. Hollist, J. Stonim, "A Design Methodology for Distributed Data Bases," Proceedings IEEE COMPCON 80, Washington D.C., September 1980.
- Stonim, J., D. Copeland, L. Macrae, P. S. Fisher, "Information in Medicine: Past, Present and Future," Proceedings Fourteenth Annual Hawaii Conference on Systems Sciences, January 1981.
- Fisher, P. S., J. Stonim, "Enterprise Structure as Influenced by Hardware Topology," Proceedings MIMT'AD1, Mexico, May 1981.
- Jacob Stonim, F. J. Maryanski, P. S. Fisher, and L. J. Macrae, "Sequential vs. Concurrent Processing: A Throughput Model," Information Systems, Vol. 6, No. 4, December 1981.
- Case, J., P. S. Fisher, "Factoring F.I. Sequences," submitted for publication.
- Fisher, P. S., I. Marschik, "Biometrics: A Support Tool to Measure Human Factors in System Development and Operations," Proceedings Second Symposium on Microcomputer and Microprocessor Applications, Budapest, Hungary, Oct. 6, 1981.
- Butler, E., P. S. Fisher, "Computers and Data: The Hospital Information System," Proceedings Fifteenth Annual Hawaii Conference on Systems Sciences, January 1982.

Fisher, P. S., "A Data Base Design Technique For Centralized or Distributed Topologies", Computer Communications to appear in 1982.

Han, M. J., P. S. Fisher, "The Problem of Data Structure on a Heterogeneous Environment", Advances in Distributed Data Base Systems Management, Vol. II, Heyden Publishing Co., 1981.

Unger, E., P. Fisher, J. Slonim, "Evolving to Distributed Database Environments", Computer Communications, Vol. 5, No. 1, February 1982.

Slonim, J., R. A. McBride, P. S. Fisher, E. A. Unger, "A Quantitative Analysis of Information Processing in Centralized and Distributed Architectures", Advances in Distributed Processing, Vol. II, Heyden Publishing Co., 1984.

J. Case, and P. Fisher, "Long Term Memory Modules", Journal of Mathematical Biology, 1984.

Unger, E. A., J. Slonim, P. Fisher and R. A. McBride, "Quantitative Analysis of Information Processing in Centralized and Distributed Architectures", in Advances in Distributed Processing Management, Vol. 2, John Wiley and Sons, 1984.

Books:

Unger, E. A., J. Slonim and P. S. Fisher Editors, Advances in Database Management, Vol. 2, Heyden Publishing Company, 1983.

Slonim, J., P. S. Fisher, E. A. Unger Editors, Advances in Data Communication Management, Vol. 2, Heyden Publishing Company, 1983.

Fisher, P. S., E. A. Unger and J. Slonim Editors, Advances in Distributed Systems Management, Vol. 2, Heyden Publishing Company, 1983.

Research Interests:

There are two problems in which I am presently interested. The problems deal with movement and utilization of information and the understanding of that information. In the first case, I am working with information in the distributed environments. The objective is to develop a procedure which permits the movement of systems in a family of possible environments. For example, a data base consists of data of three essential forms: modifiable, non-modifiable, and derivable. Clearly, every data base must contain all modifiable and non-modifiable data. However, the inclusion of derivable data is optional. This type of data then comprises a range or family of equivalent data bases, and I am developing a framework for moving the data base and the corresponding applications over this range in an automatic fashion.

The second problem is, as indicated, dealing with understanding information. For example, a typewriter which types from verbal input is a case in point. This area borders upon AI, mathematics and information processing. We are well along into this area using a mathematical framework for describing such problems. It is called a pre-algebra for finitely inductive sequences. This class of sequences can be used to describe all regular phenomena. The typewriter is such a phenomenon.

Both of these areas are couched in larger areas of interest, especially the data base problem. Hence, I am interested in problems in several distributed processing and data bases, especially those with irregular internal structures, such as text data bases, networks, communication and associated languages for such systems.

RESUME

William J. Hankley

Address

1715 Laramie  
Manhattan, Kansas 66502  
(913-539-1028)

Position

Professor  
Department of Computer Science  
Kansas State University  
Manhattan, Kansas 66506  
(913-532-6350)

Areas of  
Interest/  
Expertise

Software engineering (specifications, design, very-high-  
level programming);  
Languages & compilers (ADA, Prolog, Modula2);  
Graphics and interactive environments.

Date of Birth

May 31, 1942, Chicago, Illinois

Soc. Sec. No.

392-34-9207

Married

Wally Lense

Children

Elizabeth (19), Cynthia (16), Kristin (12)

Education

1962 BS Elec. Engineering, Northwestern Univ.  
1964 MS (Information Science), Northwestern Univ.  
1967 PhD EE (Computer Science), Ohio State Univ.  
1967-present many years of computer science teaching

Teaching/  
Research

1981-Present Current Position  
1980-1981 Computer Scientist  
Programming Sciences Division  
National Bureau of Standards

1971-1980 Associate Professor

Kansas State University

1967-1972 Assistant Professor of Computer Science

and EE, University of Utah

Consulting

Physiology Dept., Univ. of Utah (1968; pattern recognition)  
Intermountain Regional Medical Program, Univ. of Utah  
(1969; systems analysis)  
Kennecott Copper Corp. (1970; process control simulation)  
Computer Sciences Corp. (1977; fingerprint recognition)  
General Electric Corporation (1978; programming languages)  
NCR Corporation (1977; network simulation)  
Computer & Information Sciences Corp. (1978-1985)  
US Army Topographic Command. (1979; recognition of river  
patterns)  
Computer Science Corp.  
(1981; seminars on concurrent systems design)  
Astronautics Corp. (1984; secure message system design)  
Programs for Learning. (Co-owner, 1983-1985)

Image Analysis and Recognition (ONR, 1966-67)  
 Digital Filtering of Minuteman Missile Tracking Data (Hill AFB, 1967-68)  
 Reliable Software (US Army, 1975)  
 Macroprocessors (US Army, 1975-76)  
 Computer Network Software (US Army, 1977-78)  
 Network Simulation (NCR, 1978)  
 Intermediate Representation of Programs (NBS, 1981)  
 Color graphics packages for KSU teaching labs  
 (FORTRAN 1972-74, Pascal 1978-80)  
 Graphics Kernel System subset (Pascal 1983)  
 Packet switching software for KSU network  
 (Pascal; Interdata/IBM 370 network, 1977)  
 Simulation package and network simulation tool  
 (Pascal 1978)  
 Worked with group supporting CPASCAL language and  
 operating system, lectured on compiler internals  
 (1977-79)  
 Sign management and billing system (1984, dBaseII)  
 Program generator tools (1984-1986, Pascal & Prolog)  
 Courses  
 Personal Computer Systems/Software  
 Programming science (specification & verification)  
 Programming languages  
 Compiler design I, II  
 Software engineering I, II  
 Computer graphics  
 Personal Computer Systems/Software  
 Operating systems  
 Semantics of languages  
 Real-time software  
 KSU students  
 32 MS students, 2 PhD students  
 Societies  
 Association for Computing Machinery (SIGPLAN, SIGSC, SIGGRAPH, student chapter adviser 1973-76)  
 IEEE Computer Society  
 Upsilon Pi Epsilon (Chapter Adviser 1977-80)  
 Reference  
 Dr. Paul Fisher, President, Computer & Information Sciences, Manhattan, KS (913-537-0613)  
 Dr. Virgil Wallentine, Head, Dept. of Computer Science, Nichols Hall, Kansas State University, Manhattan, KS 66506 (913-532-6350)  
 Publications  
 W. Hankley, J. Tou, "Note on Control of Multiple Input Discrete Systems", IEEE Transactions on Automatic Control, Vol. AC-12,4, August 1967, pp. 469-470.  
 W. Hankley, J. Tou, "Automatic Fingerprint Interpretation and Classification via Contextual Analysis and Topological Coding", in Pictorial Pattern Recognition, Thompson Book Co., Washington, D.C., 1968, pp. 411-456.

W. Hankley, "Fingerprint Classification for Automated Processing", Proceedings, Carnahan Conference on Electronic Crime Countermeasures, Univ. Kentucky, 1968, pp. 70-82. Also presented at 1968 WESCON.

W. Hankley, H. Merrill, "A Pattern Recognition Technique for System Error Analysis", IEEE Transactions on Reliability, Special Issue Vol. R20, 3, August 1971, pp. 148-153.

P. Aberglen, W. Farris, W. Hankley, "Design of a Real-Time Central Data Acquisition and Analysis System", Proceedings of the IEEE, Special Issue, Vol. 58, 1, January 1970, pp. 38-48. Reprinted in J. Schoeffler, R. Temple (Eds), Minicomputers: Hardware, Software, and Applications, IEEE Press, New York, 1972.

W. Hankley, N. Miller, "Characterization of Pulse Waveforms for Classification", (Abstract) IEEE Symposium on Feature Extraction and Selection, New York, 1971.

F. Templeton, W. Hankley, "Dynamic Control of a Process with Discrete and Continuous Decision Variables", in Techniques for Decision Making in the Mineral Industry, Canadian Institute of Mining and Metallurgy, Montreal, Quebec, 1971.

T. Carey, W. Hankley, "Empirical Modeling of Occurrences of Severe Weather Events", Proceedings, Conference on Environmental Engineering, Society of Engineering Science, Washington, D.C., 1971, pp. 83-108.

P. Fisher, W. Hankley, V. Wallentine, "Separation of Introductory Programming and Language Instruction", ACM SIGSCE Bulletin, Vol. 5, 1, February 1973.

W. Hankley, V. Wallentine, "Programming Language Design for a Spectrum of Users", Proceedings 7th Annual Conference on the Interface of Computer Science and Statistics, Iowa State University, October 1973.

W. Hankley, P. Fisher, "Top Down Refinement of Assertions", Proceedings Third Texas Conference on Computing Systems, November, 1974.

W. Hankley, V. Wallentine, "Color Graphics for Remote Teaching", Proceedings 1980 SIGGRAPH Conference SIGGRAPH, Vol. 14, 3, 1980.

W. Hankley, V. Wallentine, "Discrete Simulation with a Concurrent Language Base", Proceedings 1981 Summer Computer Simulation Conference, July 1981, pp. 12.

V. Wallentine, W. Hankley, "Modeling and Simulation of the Performance of Distributed Data Management Systems", in P. Fisher, E. Unger (Editors), Advances in Distributed Process Management, Haydon and Sons, 1983, pp. 39.

W. J. Hankley, W. L. Hankley, "Teach PDL, Not Flowcharts", submitted to Computing Teacher SIGCS Newsletter, Feb 1986.

W. Hankley, "Feature Analysis of Turbo Prolog", submitted to ACM SIG PLAN Notices, October 1986.

Technical Reports (\* denotes funded project) Repose available from KSU except as noted.

\*W. Hankley, "Source-Environment Models for SO2 Concentration", Kennecott Copper

Corp., Salt Lake City, Utah, 1971.

G. Anderson, W. Hankley, Users Guide, Computer Science Graphics Package, 100 pp., 1974. (Used as class reference.)

\*CS 75-01, P. Fisher, W. Hankley, J. McCall, Steps Toward Reliable Software: Proceedings of a Workshop, 90 pp., January 1975.

\*CS 75-02, J. Carrow, P. Fisher, W. Hankley, J. McCall, Toward Reliable Software: A Workshop on Structured Programming, Proceedings of a Workshop, 147 pp., February 1975.

\*CS 75-03, I. Sagie, A. Gonen, W. Hankley, EPS II - An Extension of ML/I: Users Primer and Guide to Applications, 115 pp., July 1975.

CS 76-18, W. Hankley, J. Rawlinson, Sequential PASCAL Supplement for FORTRAN Programmers: A Primer of Slides, 161 pp., December 1976.

\*W. Hankley, Design of the MIMICS Message System, 98 pp., 1977.

\*CS 77-01, V. Wallentine, W. Hankley, G. Anderson, M. Calhoun, F. Maryanski, Progress Report on Functionally Distributed Computer Systems Development: Software and Systems Structure, 142 pp., December 1976.

CS 77-12, P. Fisher, W. Hankley, F. Maryanski, Porting Software to Multiple Minis: A DBMS Case Study, 23 pp., December 1976.

D. Snyder, W. Hankley, Conversion of the Computer Science Graphics Package to PASCAL, 75 pp., 1977. (Used as class reference.)

\*CS 79-05, V. Wallentine, W. Hankley, R. McBride, SIMMON -- A Concurrent Pascal Based Simulation System, 52 pp., 1979.

\*CS 79-02, W. Hankley, V. Wallentine, A. Skidmore, NEISIM - Network Simulation System, 50 pp., 1979.

CS 83-02, W. Hankley, Language Structure for Reusable Software, 37 pp., 1983.

D. Bodie, "A Graphic Tool for Generating ADA Language Specifications", MS Thesis, 90 pp., 1985.

M. Peak, "A Prolog Prototype of a Module Development System", MS Thesis, 125 pp., 1985.