MARTIN, JOHN RANDOLPH. Development of Computer Supplements to Calculus. (1972) Directed by: Dr. William P. Love pp. 73

In this thesis the author presents a survey of computer calculus projects being conducted or recently conducted. A proposed computer calculus course for the University of North Carolina at Greensboro is also presented.

APPROVAL SHEET

This thesis has been approved by the following committee of the Faculty of the Graduate School at The University of North Carolina at Greensboro.

Thesis Advisor William P. Love

Oral Examination Committee Members Hughes B. Hoyle, III

Robert L. Dornhardt

E. E. Posey

June 23, 1972
Date of Examination

TABLE OF CONTENTS

CHAPTER I

SURVEY OF COMPUTER CALCULUS PROJECTS IN AMERICA

In 1965 the Committee of the Undergraduate Program in Mathematics (CUPM) presented to the Mathematical Association of America its report: A General Curriculum in Mathematics for Colleges. In an article concerning this report, W. L. Duren, Jr., the chairman of CUPM, discussed the question of including a computer course in the general mathematics curriculum. At that time he raised the following questions:

> "Should we say that a three-hour Computer Science, which might be labelled Math. 2C, is the business and even responsibility of mathematics departments to teach? Alternately can and should a shorter introduction to programming be adjoined to existing mathematics courses? Should mathematics courses be modified to include homework on the computer? These are hot issues. Our attitude was: Let us wait and see." [13,p.828]

This thesis intends to examine those questions relative to recommendations from prominent professionals and professional societies and a report of some of the experiments presently being conducted or recently conducted with respect to computer supplemented calculus, which will henceforth be referred to as computer calculus. This thesis also intends to present part of a computer calculus course based on what has been done previously and the particular needs of the Mathematics Department of the University of North Carolina at Greensboro. This course is being developed by

Dr. William P. Love, Ms Carolyn T. Jones and Mr. John R. Martin.

The motivation for including computing in the mathematics curriculum comes from many sources.

In February, 1965 the Committee on Applications of Mathematics of the Division of Mathematics, National Academy of Sciences - National Research Council recommended that any well-rounded curriculum should include courses that use the computer. [16]

At the Florida State University - ACM - SIAM Symposuim in 1966, Dr. Wallace Givens, then director of the applied mathematics division at Argonne National Laboratory, "defended the need for teaching algorithms which can be used to instruct computers to carry out long sequences of operations needed to obtain a reasonably accurate approximate answer, to complete an exhaustive search of possible alternatives, to compute and print needed tables or to perform lengthy sequences of symbol manipulation." [19,p.388]

In CUPM's Recommendations for an Undergraduate Program in Computational Mathematics, the basic recommendation was "that mathematics departments should experiment with innovative undergraduate mathematics programs which emphasize the constructive and algorithmic aspects of mathematics, and which acquaint students with computers and the uses of mathematics in computer applications." [8,preface]

In Guidelines and Standards for the Education of Secondary School Teachers of Science and Mathematics, published by the American Association for the Advancement of Science (AAAS) in the early 1970's stated the following:

"An undergraduate program for secondary school mathematics teachers should include a substantial experience with the field to computing as it relates to mathematics and to the teaching to mathematics." [1,p.26]

Since many mathematics teachers, whether teaching in high school or college, will eventually come in contact with a computer, the AAAS also stated that these mathematics teachers:

"1. Understand the relationship of mathematics to computing.
2. Have some appreciation of the effect computing has had, not only on the natural sciences but also on the social sciences, and on society in general.
3. Become familiar with the use of computers for individualized instruction and classroom demonstrations.
4. Recognize the limits of complexity of the general purpose computer." [1,p.26]

They also stated the following concerning competency of computing:

"Regardless of how he acquires his computer competency, the prospective teacher should be able to convert a physical model or problem to an algorithm; convert an algorithm to a flow chart; and convert a flow chart to a program in a language such as BASIC, FORTRAN, or ALGOL." [1,p.26-27]

One of the roles of the university is to prepare the student for his life outside the university community and to provide society with trained people for its many needs. In order to achieve this goal it is desirable to keep the student abreast of the current trend of computing in our society.

The necessity of computing in the mathematics curriculum has been established by the above recommendations and needs of society.

Concerning the appropriate place to introduce computing into the mathematics curriculum, the following recommendations have been made.

In 1965, the CUPM in its report, A General Curriculum in Mathematics for Colleges, stated the following:

> "The prevalence of the high-speed automatic computer affects the teaching of mathematics in a very general way. Many mathematically trained students will work closely with computers, and even those who do not should be taught to appreciate the type of algorithmic approach that enables a problem to be handled by a machine. This point of view should therefore be presented, along with the more classical one, at appropriate places in calculus, differential equations, linear algebra, etc." [6,p.14]

At the forementioned FSU-ACM-Siam Symposium, Dr. Frank Murray of Duke University suggested that a computing course be offered as part of or paralled to the student's first mathematical analysis course. If this approach were taken, then the computer could be used to develop such topics as functions, limits, series, and the definite integral. [19]

At a conference sponsored by the National Science Foundation, conducted at the Science Teaching Center of the University of Maryland on December 8-9, 1967, the topic of the computer in undergraduate curricula was examined. The conclusion was reached that "computing activities and relevant areas of numerical analysis should be integrated into the first college mathematics course and in subsequent courses when appropriate and natural." [21,p.2]

In CUMP's Recommendations on Course Content for the Training of Teachers of Mathematics, it was recommended as part of their minimum preparation for mathematic teachers, to include three semesters of calculus instruction and that it is "desirable to take advantage of the growing role of computers in introducing mathematical concepts." [9,p.17]

Philip J. Davis of Brown University had the following to say concerning introducing computers with calculus:

> "Among the various possible presentations of elementary calculus, there are those which
>
> a) develop the subject for its own sake and at its own level.
>
> b) develop the subject for subsequent utility in higher pure mathematics.
>
> c) develop the subject for subsequent utility in applied mathematics, physical science, engineering, mathematical economics, etc." [10,p.12-13]

A computer calculus course is another alternative which could teach the concepts of the above three presentations. [10]

From the above recommendations it would appear logical to introduce computing simultaneously with calculus instruction since calculus is traditionally the first course in mathematical analysis. One might ask how this is to be accomplished. It appears there are three basic approaches to the problem. The first approach could be to totally integrate calculus and computing by teaching programming techniques from the computer calculus text itself. At the other extreme, the two subjects could be taught independently with little or no connection between the calculus class and the computer class. It would be left to the student to establish the relation between calculus and computing. The third approach is to have standard calculus instruction with a coordinated laboratory, working in conjunction with the classroom assignments.

In trying to establish which approach is the most effective the following questions were considered in studying the various projects conducted:

1) Which of the three approaches or variations of these approaches were attempted?
2) How successful was the approach?
3) Which mode was used: batch processing or interactive (use of time-sharing consoles) mode?
4) What programming language was used?
5) What was the cost per student?
6) Was academic credit given?

The Center for Research in College Instruction of Science and Mathematics (CRICISAM) project dealing with the inclusion of computers into calculus was begun in 1966 as the result of the recommendation of the Committee on Undergraduate Programs in Mathematics (CUPM) to the Mathematical Association of America (MAA) and the Symposium on the Impact of Computing in Undergraduate Mathematics Instruction which was funded by the National Science Foundation. The CRICISAM approach involves total integration of calculus and computing into one course and is nondependent on language. [20] The CRICISAM project is now the most extensively used project with at least one hundred schools using it as of May, 1972. [8]

North Carolina Agricultural and Technical State University has used the CRICISAM approach in its three semester sequence of calculus on a trial basis. This course was given four semester hours credit and was taken by students on a voluntary basis. The student reaction was very favorable and Professor Octavio Diaz, the instructor, offered his personal opinion that students learned calculus as well or better than the traditional approach with their interest and comprehension increased. However, he also expressed some reactions about the inadequacies of the CRICISAM text. For example, since their sequence included analytic geometry and multivariable calculus and since the CRICISAM text was inadequate in these topics, the CRICISAM text had to be supplemented. [12]

Vanderbilt University has been using the CRICISAM approach for the past three years as an honors level course and reports that the attempt has been reasonably successful. [27]

Purdue University has also been using CRICISAM for three years as an optional course that meets five hours per week with five hours credit. Although no data was collected comparing the effectiveness of computer calculus and traditional calculus, those who have taught the course report that the students learn calculus as well and "develop a greater appreciation of infinite processes." [14,p.1] "The major strong point of the CRICISAM or other computer oriented approaches to calculus is that the students can see within one framework the power of classical calculus and the utility of approximation techniques." [14,p.2] The major weak point of the CRICISAM text was the lack of multivariable calculus. Again, student reaction was favorable. [14]

The University of Minnesota had their first computer calculus experiment in 1967-1968 using the CRICISAM approach with an honors section of fifty students. A subset of FORTRAN was used and about seven days were spent on algorithms and flow charts. Batch processing was used due to the lack of consoles. No report was given reflecting the success of the project.

An anonymous student at State University College at Fredonia said the following in an article concerning the CRICISAM text:

"The book seems to be involved as far as proofs go. This is what is confusing to me. However, the material doesn't seem too bad, just hard to understand." [5,p.6]

In a report of a panel discussion of CRICISAM, Richard Andree of the University of Oklahoma said their CRICISAM project was considered a failure because two different books were used in the calculus sections, the CRICISAM text and a standard text, which the non-experimental sections used. Difficulties arose when students dropped out of the CRICISAM course into the standard course. [5]

J. C. Wicht, Chairman of the Department of Mathematics at North Georgia College, calls the CRICISAM text "impossible." [26] Walter J. Carpenter of North Ceorgia College reported that the CRICISAM text is too theoretical for the students. [4]

The University of North Carolina at Chapel Hill has used CRICISAM extensively for the past two years. Approximately two hundred fifty students in the fall term and one hundred in the spring term registered for the course. This course was not required and met three hours a week with three hours credit. It was reported that good students learned calculus better while the poorer students learned worse, possibly due to lack of time for both computers and calculus. It was also reported that many students were "turned on" by the course and in general, students understood the topic of limits better. There was a mixed reaction from the students concerning the course. [18]

Brown University first taught computer calculus in 1968-1969 where they used three standard lectures in calculus and one or two laboratories each week dealing with programming techniques. This course was optional and approximately seventy students participated. A local interactive language known as BRUIN was used with approximately five minutes of CPU time being used by each student per semester. The experiment was reported a success from both students and faculty. Philip J. Davis of Brown University made the following remark concerning the question, "Do student learn calculus better?"

> "We cannot claim - and we think it would be foolish to do so- that a student learns calculus better in a computer calculus section than in a conventional section. We think this is barking up the wrong tree. What is important is that computer calculus offers an interesting and relevant option for the teaching of the calculus, an option which will lead to new skills, new

attitudes and new interests." [11,p.13]

Case Western Reserve University required all calculus students (750) to be taught programming in special sessions by undergraduates. They are presently using batch processing mode but plan to go to interactive mode and also develop a lab to be coordinated with course material. [7] Information was not available as to its success.

Pomona College offers an optional course in PL/1 with their second semester of calculus coordinated with infinite series. [7] Information was not available as to its success.

Claremont Men's College requires all students to take a semester of calculus which includes FORTRAN instruction in the batch mode. [7] Information was not available as to its success.

Harvey Mudd College gives a special two-week, two hours per day, seminar on methods of solution and numerical techniques. Freshmen learn FORTRAN in a required, one-semester hour course. [7] Information was not available as to its success.

At Dartmouth College computing is taught with second semester calculus and in a finite mathematics course for non-physical science students. Between two and three hundred students each tri-semester use BASIC, an interactive language, forty-five minutes per week (connect time). [7] Information was not available as to its success.

The University of Denver taught a special section computer calculus, which forty-eight students used a standard calculus text with motivation toward the algorithmic methods. The students learn FORTRAN in a separate course using batch processing mode. A later section used BASIC in the interactive mode and this mode was preferred over batch. Plans are to expand the computer calculus to all freshman calculus sections. [7]

Florida State University has several projects involving calculus with computers. One project involves students who already have a background in calculus and computers. They choose a special calculus course with computer lab, where they study numerical topics of calculus. Another project is the CRICISAM course which meets five hours per week learning and using FORTRAN with calculus. [7]

A computer laboratory was used with three semesters of calculus at the University of New Mexico. One hour per week was required in the laboratory using a subset of FORTRAN in the interactive mode. Approximately five hundred students took part in this project. [7] Information was not available as to its success.

At the University of Pennsylvania a special section of one hundred forty students was taught computer calculus. FORTRAN was taught within six lectures and never mentioned again. Batch mode was used but interactive was preferred. [7] Information was not available as to its success.

Oberlin College sets aside eight class meetings for instruction in FORTRAN. Each professor makes use of the computer as he sees fit within his classroom instruction. Some do not use it as all, while others require the writing of programs. [7] Also at Oberlin, the CRICISAM project was tried several years ago with mixed results. Now a two week block is set aside for instruction in programming, then it is left up to each instructor to the extent he wishes to use computing. [2]

The University of Utah has given an experimental calculus course during 1965-1969 with about twenty to forty students each semester. They replace one lecture per week with a two hour laboratory with the same amount of credit as the standard course. [7] In 1971 they gave three lectures and two coordinated laboratories weekly with four hours credit. [25]

The University of Iowa has experimented with a computer laboratory with both calculus and linear algebra. In 1969-1970 their project began with a special section using FORTRAN or PL/1 in batch mode at a cost of twenty one dollars per student. In 1970-1971 they offered an one hour optional laboratory which could be taken with each of the courses in the calculus - linear algebra sequence. About twenty students each semester used BASIC language at a cost of sixty dollars per student per semester. In 1971-1972 they plan to use a subset of PL/1 in the interactive mode. The student reaction was favorable and found the optional laboratory course very helpful. [17]

The Educational Research Center at Massachusetts Institute of Technology has undertaken a project which involves students working at their own speed in a computer based first-year calculus course. This project involves approximately one hundred students from M.I.T., Tufts, North Shore Community College, and the University of Massachusetts at Boston. Interactive mode using BASIC was used by the students to complete a minimum set of units which deal with a topic or group of topics. The laboratory course is ungraded but the students are given a test after each unit to determine if he continues to another unit. [15]

At Colorado College, in 1968, a random section of twenty-six students took part in a computer calculus project using BASIC in the interactive mode. The students received four hours credit and considered the course successful. [3]

At Duke University several approaches have been made toward computer calculus. In 1967-1968 there was an experiment involving twenty-three students using FORTRAN in batch mode. Computer programs were not required as part of the course and were optional but the students were encouraged to write the programs. The experiment was not successful due to several factors, of those being that self-motivation is not encouraged in the university reward system, lack of fast turnaround and insufficient number of keypunches. However, the experiment was considered a partial success "with respect to the principle goal of using numerical computations to enhance the study of calculus." [23,p.10] During 1971-1972, Duke was experimenting with an optional one hour per week supplement to calculus, where students investigated the relationship between calculus and computing. With this experiment the interactive mode was used. [24]

In using the computer to present concepts of pure mathematics and even establishing definitions, G. J. Porter of the University of Pennsylvania made the following statement:

> "In conclusion let me remark that there is a tendency on the part of some of those who use the computer in teaching calculus to use it only to compute those integrals which do not yield to analytic methods. While the technique may from time to time be useful in applications, I believe it adds nothing mathematically to the calculus course. On the other

hand I think the computer can be used effectively in motivating and helping the student understand the pure mathematics." [22,p.1001]

The following are observations of the author concerning the fore-mentioned projects involved in computer calculus.

Looking at the approaches taken to incorporate computers into calculus, all three basic approaches can be judged successful. No one method of incorporating computers into calculus can be judged to be absolutely better than any other approach mentioned in this thesis. Students learn calculus and computing no matter which approach is taken. However, there have been failures. These failures can be linked with equipment deficiences and personnel problems. The lack of an appropriate number of keypunch machines or interactive terminals caused unneccessary waiting for the students. An over-burdened computer or computer terminal also caused unneccessary waiting. In informal discussions, it was reported that some projects failed because the staff teaching the computer calculus courses were not interested in teaching the courses or were not qualified. This thesis gives some insight to the various problems confronted and possible solutions to those problems.

The CRICISAM approach, which can be considered to be the totally integrated approach, contained two basic problems. The first problem is that of content. Students using CRICISAM often did not get enough traditional calculus. For example, the topics of multivariable calculus and analytic geometry are not covered by the CRICISAM text. A future edition including these topics could remedy this problem. The second problem is that of student mobility. If the CRICISAM text is not used in all sections of calculus, then the students encounter a problem when they decide to drop out of the CRICISAM course into a traditional course using a different text. There is also a problem for transfer students who have been using a different text.

Concerning the question of the superiority of the interactive mode or the batch processing mode, the interactive mode was preferred over batch processing. The interactive mode has two basic advantages. First, the turnaround time is faster; therefore, the students do not have to wait an unreasonable amount of time for the program results. Second, cards and keypunches are no longer needed. However, there is one important disadvantage. The cost factor is much greater. For example, at the University of Iowa, there was a three to one ratio of cost of the interactive mode to batch processing. [17] There is also the cost of specialized interactive terminals. It is believed the batch processing mode may be made as successful as the interactive mode provided two conditions are met. First, make sure the student does not have to wait an unreasonable amount of time by providing a sufficient number of keypunch machines and make them easily accessible. Second, have an efficient computer or computer terminal to guarantee fast turnaround for the student.

In most instances more academic credit was given for the computer calculus than the traditional calculus course.

Many different programming languages were used at the various schools. Any easy to learn, high-level language can be used for this type of course. Following is a list of high-level languages that were used or could be used:

1) PL/1 and subsets (interactive and batch),

2) FORTRAN and subsets (interactive and batch),

3) BASIC (strictly interactive),

4) BRUIN (strictly interactive).

CHAPTER II

PROPOSED COMPUTER CALCULUS PROJECT AT UNC-G

The Mathematics Department of the University of North Carolina at Greensboro is planning to offer an experimental computer calculus course in the fall semester of 1972.

A computer laboratory is being developed which will operate in conjunction with the elementary calculus sequence. The classroom lectures and laboratory exercises will be highly coordinated so that student should learn both calculus and computing better than if each were taught separately. The content of the laboratory will include the basic ideas in computing equipment and the operation of this equipment; the algorithmic process for solving problems; essential knowledge of a programming language (PL/C); and an extensive set of exercise programs per semester which every student must write. The calculus course and laboratory will meet five hours per week, three hours for calculus lecture and two hours for computer lab. Each semester will count as five semester hours credit. The laboratory exercises will count as part of the total course grade. Students will have a great deal of mobility since the same calculus text will be used in normal sections. Students will be allowed to drop out of the experimental computer calculus course into the normal course; however, students will not be allowed to enter the experimental sections once instruction has begun.

Because of existing campus computing facilities, the laboratory exercises will be designed for the batch-processing mode using an IBM 2780 rather than conversational (interactive) mode. However, some experimentation will be conducted to determine the effectiveness of both types of operating modes.

The following list includes skills and understandings which will be expected from all students completing the computer calculus sequence;

1) The student should have a solid understanding of the traditional approach to calculus, especially from the theoretical point of view.
2) The student should have an understanding and ability to use computer equipment including card punch, card reader and printer, and teletype.
3) The student should understand and be able to develop algorithms in analyzing and solving problems.
4) The student should have sufficient mastery of a programming language.

Since there is a lack of current publications or materials which correspond to our ideas, a computer calculus laboratory manual is being developed. The manual is divided into two different parts. The first eight lessons deal basically with teaching the essentials of PL/C to the student so he will be able to write simple programs. The remainder of the lessons deal with incorporating computers into calculus. A basic assumption is that the student has no prior computer programming knowledge. A table of contents is included in Appendix I along with sample lessons from the laboratory manual.

One of the assumptions of the computer calculus experiment is that the laboratory experience will result in a better understanding of the traditional calculus. An attempt will be made to compare the achievement of those students in computer calculus with those in the traditional calculus. Standardized tests will be given to all students upon initial enrollment, then they will all be tested again at the end of the first semester and second semester of calculus. Data will then be analyzed to evaluate our assuumptions.

Further revisions and additions will be made to the laboratory manual as needed.

APPENDIX

TABLE OF CONTENTS

Lesson 14 FUNCTIONS

1. PURPOSE: The objective of this lesson is to present the basic ideas concerning functions, their graphs, and how to determine certain properties of functions.

2. SOME DEFINITIONS:

There are several ways functions may be defined. Two definitions are given below:

Defn: A FUNCTION consists of a set of ordered pairs, (x,y), such that if any two distinct ordered pairs have the same first element, x, then they also have the same second element, y.

Defn: A FUNCTION consists of a set, X, and a set, Y, and a RULE which associates every x in X with exactly one y in Y.

Frequently, functions are defined over the set of real numbers or over some subset of the real numbers, but it is not necessary that functions be defined over numbers at all. In calculus, we are primarily interested in functions defined over real numbers, hence the examples will be restricted to this type of function.

Defn: The DOMAIN or DOMAIN OF DEFINITION of a function is the set of real numbers where the function is defined. That is, the domain is the set of all x's in the ordered pairs (x,y).

Defn: The RANGE of a function is the set of all possible values that the function may assume. That is, the range is the set of all y's in the ordered pairs (x,y).

The computer may be very useful in determining the properties of a function, especially very complex functions whose values are difficult to compute. Frequently, one would like to know the domain and range of a function.

In relatively simple functions one can usually determine the domain and range by inspection or a few calculations. Functions are undefined whenever division by zero or square roots of negative values are involved. The computer cannot always determine precisely the domain or the range although it can give some hints or clues which will provide a reasonable guess. Error messages will result whenever functions are evaluated at points not in their domain.

Sometimes, a person is interested in knowing where a function has positive (or negative) values.

Defn: A function is said to be POSITIVE over an interval provided that $f(x)>0$ for all x's in the interval.

Sometimes, a person is interested in knowing at which points the function has a zero value. This is equivalent to finding where the graph of a function intersects the X-axis.

Defn: A function is said to have a ZERO at a point $x=c$ provided $f(c) = 0$.

Sometimes, a persn is interested in knowing where a function is increasing or decreasing. This is equivalent to finding where the graph of the function is rising or falling.

Defn: A function is said to be INCREASING over an interval provided if $c<d$ in this interval, then $f(c)\leq f(d)$.

Defn: A function is said to be STRICTLY INCREASING over an interval provided if $c<d$ in this interval, then $f(c)<f(d)$.

By using the computer to print a table of values for "x" and for "f(x)", one is often able to approximate the domain of the function, locate where the function is undefined, determine where the function assumes positive or negative values, approximate zero's of the function, and find where the function is increasing or decreasing.

By printing a table of values for "x" and for "f(x)", one essentially is able to plot these values to form the graph of the function. In determining the values of "x" to be selected, two decisions must be made: first, the interval where the values must range, that is, the beginning and ending values for x, and second, how many values are to be used. For example, suppose one decides to evaluate the function over the interval from -5 to +5. This determines the upper and lower boundary where the function is to be evaluated. Within this interval, one may decide to evaluate the function at the values X=-5,-4,-3,-2,-1,0,1,2,3,4, and 5. In this case, the points are one unit apart, and we say that $\Delta X = 1$ (or DX = 1). This provides eleven values to be evaluated. Possibly the gaps are too far apart, and a DX = 0.5 would give better results. In this case, the function would be evaluated at x = -5,-4.5,-4,-3.5,...,3.5,4,4.5, and 5, this time at 21 points. If this is not sufficient, then a DX = .1 might be selected and the function would be evaluated at x = -5,-4.9,-4.8,-4.7,...,4.7,4.8,4.9, and 5.0. <u>Any value for DX may be used provided it is POSITIVE</u>. Large values of DX result in a coarse approximation of the graph of the function and small values of DX provide a reasonable approximation. However, very small values of DX cause the computer to execute many computations which can become very expensive.

3. EXAMPLE 1: Write a program which will print a table of values for "x" and for "f(x)" defined by the function: $f(x) = \sqrt{x^2 - 9}$. Let the variable "x" cover the interval from -10 to +10. Let DX = 0.5. By examining the output, answer the following questions.

a. What is the domain of the function?

b. What is the range of the function?

c. Where is the function undefined?

d. Where is the function positive?

e. Where is the function negative?

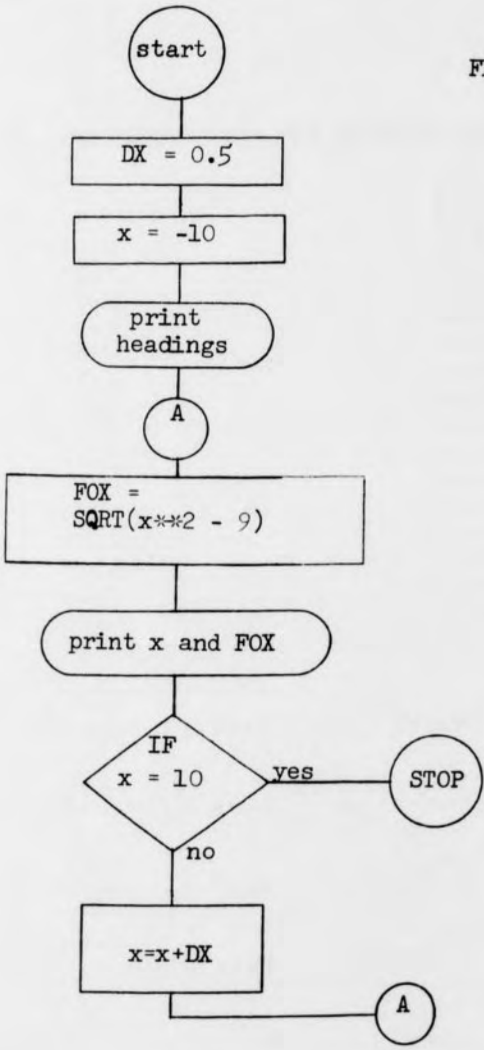
f. Where does the function have zeros, if any?

g. Where is the function increasing?

h. Where is the function decreasing?

A. One possible flowchart for this problem might be as shown below.

FLOWCHART Lesson 14 Example 1



B. A PL/C program based on the above flowchart could be written:

```
//EXAMPLE1 PLC ECS.UNCG.MA600237,LESSON14,OPT=NOBOUNDARY
 FCN:       PROCEDURE OPTIONS(MAIN);
            DECLARE (DX,                          /* DELTA X    */
                     X,                           /* CURRENT X  */
                     FOX) FLOAT(16);              /* F(X)       */
            DX=0.5;    /* INITIALIZE DELTA X  AND X */
            X=-10.0;
 /* PRINT OUT HEADING */
            PUT LIST('  X                    F(X)');
 EVFOX:     FOX=SQRT((X**2)-9);  /* EVALUATE F(X) */
 /* PRINT OUT X AND F(X) */
            PUT EDIT(X,FOX)  (F(5,1),X(10),F(10,5)) SKIP;
            IF X=10 THEN STOP;   /* TEST TO SEE IF X HAS REACHED 10 */
            X=X+DX;   /* ADD DELTA X  TO X */
            GO TO EVFOX;  /* GO EVALUATE NEW F(X) */
 END FCN;
 *DATA
 //
```

C. The output from the above program is shown below:

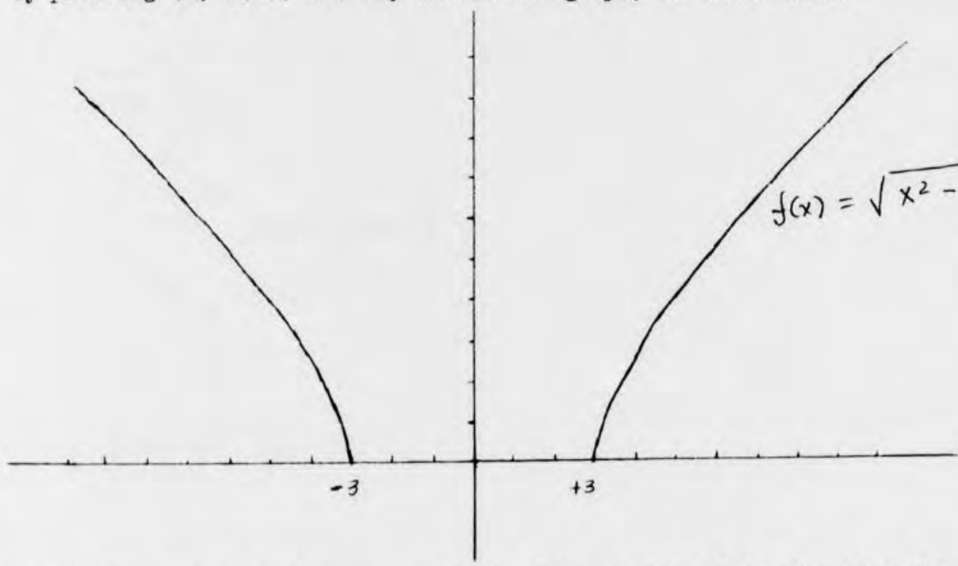
```
  X                     F(X)
-10.0                  9.53939
 -9.5                  9.01387
 -9.0                  8.48528
 -8.5                  7.95298
 -8.0                  7.41619
 -7.5                  6.87386
 -7.0                  6.32456
 -6.5                  5.76628
 -6.0                  5.19615
 -5.5                  4.60977
 -5.0                  4.00000
 -4.5                  3.35410
 -4.0                  2.64575
 -3.5                  1.80277
 -3.0                  0.00000
****** IN STMT   6  ERROR  EX8D    THE ARGUMENT IS NEGATIVE (###).
                       1.65831
 -2.5
****** IN STMT   6  ERROR  EX8D    THE ARGUMENT IS NEGATIVE (###).
                       2.23606
 -2.0
****** IN STMT   6  ERROR  EX8D    THE ARGUMENT IS NEGATIVE (###).
                       2.59807
 -1.5
****** IN STMT   6  ERROR  EX8D    THE ARGUMENT IS NEGATIVE (###).
                       2.82842
 -1.0
****** IN STMT   6  ERROR  EX8D    THE ARGUMENT IS NEGATIVE (###).
```

Lesson 14 (cont)

```
   X                    F(X)

 -0.5                 2.95803
****** IN STMT    6  ERROR  EX8D   THE ARGUMENT IS NEGATIVE (###).

  0.0                 3.00000
****** IN STMT    6  ERROR  EX8D   THE ARGUMENT IS NEGATIVE (###).

  0.5                 2.95803
****** IN STMT    6  ERROR  EX8D   THE ARGUMENT IS NEGATIVE (###).

  1.0                 2.82842
****** IN STMT    6  ERROR  EX8D   THE ARGUMENT IS NEGATIVE (###).

  1.5                 2.59807
****** IN STMT    6  ERROR  EX8D   THE ARGUMENT IS NEGATIVE (###).

  2.0                 2.23606
****** IN STMT    6  ERROR  EX8D   THE ARGUMENT IS NEGATIVE (###).

  2.5                 1.65831
  3.0                 0.00000
  3.5                 1.80277
  4.0                 2.64575
  4.5                 3.35410
  5.0                 4.00000
  5.5                 4.60977
  6.0                 5.19615
  6.5                 5.76628
  7.0                 6.32455
  7.5                 6.87386
  8.0                 7.41619


  8.5                 7.95298
  9.0                 8.48528
  9.5                 9.01387
 10.0                 9.53939
```

Carefully observe the output from this program. Error messages are given when the computer attempts to evaluate the function for values of x from -2.5 to +2.5. Here the value under the radical is negative and the function is undefined, hence these values are not in the domain of the function. Obviously the domain should be values of x less than or equal to -3 and values of x greater than or equal to +3.

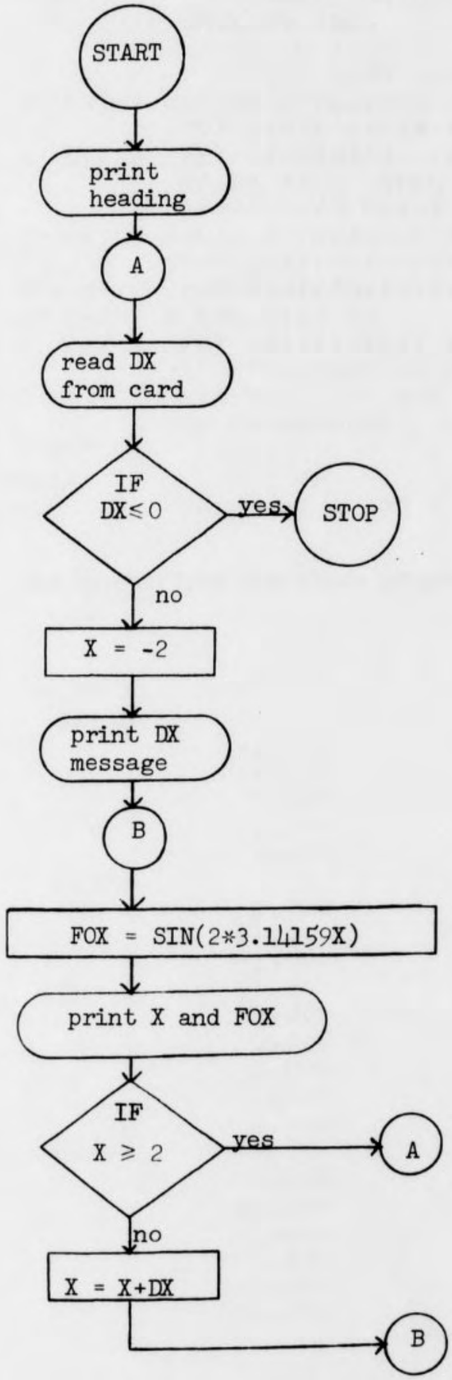
Answer questions b through h for yourself by examination of the output. By plotting (X, F(X)) one may obtain the graph, as shown below:



4. EXAMPLE 2: Write a program which will print a table of values for "x" and for "f(x)" defined by the function: $f(x) = \text{SIN } 2\pi x$. Let the variable "x" cover the interval from -2 to +2. Let DX have values of 1.0, 0.5, 0.25, 0.10, and 0.05. (This example illustrates how false conclusions may be made about a function whenever too few points are evaluated). Answer questions a through h from Example 1 for this function by examination of the output.

A. One possible flowchart for this problem might be as shown on the next page:

FLOWCHART Lesson 14 Example 2



B. **A PL/C program based on the above flowchart could be written:**

```
//EXAMPLE2 PLC ECS.UNCG.MA600237,LESSON14,OPT=NOBOUNDARY
 FCN:       PROCEDURE OPTIONS(MAIN);
            DECLARE (DX,                    /* DELTA X   */
                     X,                     /* CURRENT X */
                     FOX) FLOAT(16);        /* F(X)      */
 /* PRINT OUT MAIN HEADING */
            PUT LIST('DELTA X         X                        F(X)');
 NEWDX:     GET LIST(DX);    /* GET DELTA X  FORM CARD */
            IF DX <= 0  THEN STOP;  /* TERMINATION CHECK */
            X=-2;   /* POINT X */
 /* PRINT DELTA X  MESSAGE */
            PUT EDIT(DX)   (X(2),F(5,3)) SKIP;
 NEWFOX:    FOX=SIN(2*3.1415926 * X);  /* CALULATE F(X) */
 /* PRINT X AND F(X) */
            PUT EDIT(X,FOX) (X(13),F(6,3),X(9),F(16,6)) SKIP;
            IF X>=2 THEN GO TO NEWDX;    /* GO GET NEW DELTA X */
            X=X+DX;    /* ADD DELTA X TO X */
            GO TO NEWFOX;    /* GO EVALUATE NEW F(X) */
 END FCN;
*DATA
1.0    .5    .25    .1    .05    0
```

C. **The output from the above program is show below:**

```
 DELTA X          X                       F(X)
  1.000
               -2.000                   0.000000
               -1.000                   0.000000
                0.000                   0.000000
                1.000                  -0.000000
                2.000                  -0.000000
  0.500
               -2.000                   0.000000
               -1.500                  -0.000000
               -1.000                   0.000000
               -0.500                  -0.000000
                0.000                   0.000000
                0.500                   0.000000
                1.000                  -0.000000
                1.500                   0.000000
                2.000                  -0.000000
  0.250
               -2.000                   0.000000
               -1.750                   0.999999
               -1.500                  -0.000000
               -1.250                  -0.999999
               -1.000                   0.000000
               -0.750                   0.999999
```

Delta X	X	F(X)
	-0.500	-0.000000
	-0.250	-0.999999
	0.000	0.000000
	0.250	0.999999
	0.500	0.000000
	0.750	-0.999999
	1.000	-0.000000
	1.250	0.999999
	1.500	0.000000
	1.750	-0.999999
	2.000	-0.000000
0.100		
	-2.000	0.000000
	-1.900	0.587785
	-1.800	0.951056
	-1.700	0.951056
	-1.600	0.587785
	-1.500	-0.000000
	-1.400	-0.587785
	-1.299	-0.951056
	-1.199	-0.951056
	-1.099	-0.587785
	-0.999	0.000000
	-0.899	0.587785
	-0.799	0.951056
	-0.699	0.951056
	-0.599	0.587785
	-0.499	-0.000000
	-0.399	-0.587785
	-0.299	-0.951056
	-0.199	-0.951056
	-0.099	-0.587785
	0.000	0.000000
	0.100	0.587785
	0.200	0.951056
	0.300	0.951056
	0.400	0.587785
	0.500	0.000000
	0.600	-0.587785
	0.700	-0.951056
	0.800	-0.951056
	0.900	-0.587785
	1.000	-0.000000
	1.100	0.587785
	1.200	0.951056
	1.300	0.951056
	1.400	0.587785
	1.500	0.000000
	1.600	-0.587785
	1.700	-0.951056
	1.799	-0.951056
	1.899	-0.587785
	1.999	-0.000000
	2.099	0.587785

Delta X	X	F(X)
	0.050	0.309016
	0.100	0.587785
	0.150	0.809016
	0.200	0.951056
	0.250	0.999999
	0.300	0.951056
	0.350	0.809017
	0.400	0.587785
	0.450	0.309017
	0.500	0.000000
	0.550	-0.309016
	0.600	-0.587785
	0.650	-0.809016
	0.700	-0.951056
	0.750	-0.999999
	0.800	-0.951056
	0.850	-0.809017
	0.900	-0.587785
	0.950	-0.309017
	1.000	-0.000000
	1.050	0.309016
	1.100	0.587785
	1.150	0.809016
	1.200	0.951056
	1.250	0.999999
	1.300	0.951056
	1.350	0.809017
	1.399	0.587785
	1.449	0.309017
	1.499	0.000000
	1.549	-0.309016
	1.599	-0.587785
	1.649	-0.809016
	1.699	-0.951056
	1.749	-0.999999
	1.799	-0.951056
	1.849	-0.809017
	1.899	-0.587785
	1.949	-0.309017
	1.999	-0.000000
	2.049	0.309016
0.050	-2.000	0.000000
	-1.950	0.309017
	-1.900	0.587785
	-1.850	0.809017
	-1.800	0.951056
	-1.750	0.999999
	-1.700	0.951056
	-1.650	0.809016

Delta X	X	F(X)
	-1.600	0.587785
	-1.550	0.309016
	-1.500	-0.000000
	-1.450	-0.309017
	-1.400	-0.587785
	-1.350	-0.809017
	-1.299	-0.951056
	-1.249	-0.999999
	-1.199	-0.951056
	-1.149	-0.809016
	-1.099	-0.587785
	-1.049	-0.309016
	-0.999	0.000000
	-0.949	0.309017
	-0.899	0.587785
	-0.849	0.809017
	-0.799	0.951056
	-0.749	0.999999
	-0.699	0.951056
	-0.649	0.809016
	-0.599	0.587785
	-0.549	0.309016
	-0.499	-0.000000
	-0.449	-0.309017
	-0.399	-0.587785
	-0.349	-0.809017
	-0.299	-0.951056
	-0.249	-0.999999
	-0.199	-0.951056
	-0.149	-0.809016
	-0.099	-0.587785
	-0.049	-0.309016
	0.000	0.000000

D. Explanation of the program in Example 2:

Card 1 is the job card.

Card 2 is the procedure card.

Cards 3 - 5 are the declare statement as in Example 1.

Card 6 is a comment.

Card 7 is a PUT statement which is the main heading. Observe the spacing between the three columns.

Card 8 is labeled NEWDX. It represents position "A" in the flowchart. This is a GET statement, instructing the computer to get a value for DX from the data deck.

Card 9 is the termination check. When the value DX=0 is read, the program transfers to Card 19 and stops.

Card 10 initializes the value of X to -2.

Card 11 is a comment referring to card 12.

Card 12 is a PUT EDIT statement. It instructs the computer to first SKIP a new line, skip 2 spaces, then print the value of DX using the fixed decimal format.

Card 13 is labeled NEWFOX. It represents position "B" in the flowchart. The function is evaluated for the given X.

Card 14 is a comment referring to card 15.

Card 15 is a PUT EDIT statement. It instructs the computer to first skip a new line; skip 13 spaces (to align under the X column); print the value of X using F(6,3) format; skip 9 spaces; print the value of FOX using F(16,6) format.

Card 16 is a decision statement. If the value for X has exceeded 2, the tables are complete and the program loops to NEWDX to get a new value for DX. If X is less than 2, a new value for X is obtained in card 17.

Card 17 is the new value of X obtained by adding DX to the old value.

Card 18 is a transfer statement, looping back to card 13.

Card 21 is the data card, containing possible values for DX. Observe the final value for DX is 0 which is used to terminate the program.

Observe the output from the program and plot the graphs for each value of DX. When DX = 1.0 and 0.5 the graph is a straight line. When DX = 0.25 the graph appears to be a sawtooth. When DX = 0.1 the graph is becoming more realistic. Only when DX = 0.05 does a more accurate picture of the graph appear.

5. STUDENT EXERCISES:

a. Write a program which will print a set of tables for thes basic functions: x^2
x^3
$\sqrt{x}$
$|x|$
$1/x$
$[x]$
$\sin x$
$\exp x$
$\ln x$

Arrange the output into ten columns, select your interval, and indicate what values for DX. For each function, determine the domain, range, and sketch the graph.

b. For any of the functions below, write a program which will print a set of values. Answer the following questions:

1. What is the domain of the function?
2. What is the range of the function?
3. Where is the function undefined?
4. Where is the function positive?
5. Where is the function negative?
6. Where does the function have zeros?
7. Where is the function increasing?
8. Where is the function decreasing?

Lesson 14 (cont)

$f(x) = x^3-x^2-8x+12$

$g(x) = \sqrt{25-x^2}$

$h(x) = 1/|x-3|$

$k(x) = x^3-x^2$

$p(x) = (x^2-x-6)/(x-3)$

$q(x) = |x| - [x]$

Lesson 16 Limits of Functions: Estimating Limits

1. PURPOSE: One of the most important concepts in calculus is the limit of a function. Frequently, one must determine if a given function has a limit, and if so, find its value. The computer can NOT be used to PROVE that a function has a limit and it can NOT determine the value of a limit if it exists, but it can be used to help make a "guess" as to the value of a limit. The purpose of this lesson is to show how the computer may be used to approximate values of limits or to estimate whether limits exist or not.

2. THEORY: The meaning of the symbol "$\lim_{x \to a} f(x) = L$" is carefully defined in your calculus text. It will also be discussed more fully in lesson 17. At this point, only an intuitive discussion of limits will be presented.

When one writes "$\lim_{x \to a} f(x) = ?$", one is actually asking a very simple question: What happens to the function when the value of x approaches the value "a"? One is not really interested in the value of the function at the value $x=a$, since $f(a)$ may not exist. How does the function behave for values slightly larger than a and for values slightly smaller than a?

For example, suppose one is interested in determining:

$$\lim_{x \to 0} (\sin x)/x = ?$$

Clearly, this function is undefined at $x=0$ since this would require division by zero. But what is the behavior of the function for values VERY NEAR zero? Does the function fluctuate radically? Does it tend toward positive or negative infinity? Does it approach a particular finite value? Does it behave the same way when x is slightly POSITIVE as when x is slightly NEGATIVE? One way to determine the behavior of this function is to print a table of values for "x" and for "(sin x)/x" then directly observe the results. Care must be taken in two respects:

1. The variable x must be exceedingly close to zero, but not actually

equal zero. If the difference between x and 0 is too small, the computer may exhibit round-off errors.

2. The variable x must be close to zero from both the positive side AND the negative side. That is, it must approach zero from above and below. Approaching 0 from only the positive side or only the negative side does not provide enough information to suggest if the limit exists.

The limit of a function may only be evaluated at one point at a time. Thus, $\lim_{x \to a} f(x) = ?$ and $\lim_{x \to b} f(x) = ?$ are two very different questions. How the function behaves near the point x=a may be very different than how it behaves near the point x=b. In fact, the limit may not exist at one point and exist at the other.

How does one use the computer to evaluate the function for values of x near a point? Let us use the example where f(x) = (sin x)/x at the point x = 0. First one must examine values of x which approach the value of zero from the positive side. That is, one may select these values for x: 0+1/2 , 0+1/4, 0+1/8, 0+1/16, ..., $0+1/2^n$. These values are all positive and get progressively closer to zero. In this case, we say "that x is approaching 0 from above." Second, one must examine values of x which approach the value of zero from the negative side. That is, one may select these values for x: 0-1/2, 0-1/4, 0-1/8, 0-1/16, ..., $0-1/2^n$. These values are all negative and get progressively closer to zero. In this case, we say "that x is approaching 0 from below".

Therefore, when given a function, f(x), and some point x=a, and asked to determine the limit of f(x) as x approaches a, if this limit exists, one must observe the behavior of f(x) as x approaches a from ABOVE, and also, the behavior of f(x) as x approaches a from BELOW.

Usually, the limit of f(x) as x approaches a from above is symbolized by: $\lim_{x \to a^+} f(x)$ and the limit of f(x) as x approaches a from below is symbolized by: $\lim_{x \to a^-} f(x)$ but for the sake of simplicity, IN THIS MANUAL ONLY, let us use the symbols L^+ and L^- respectively.

By using the computer, one can see if L^+ exists by printing a table of values for "x" and for "f(x)" where x approaches a from above. There are three possibilities for L^+:

1. L^+ may approach a fixed value (finite).
2. L^+ may fluctuate so radically that it does not converge to any particluar value.
3. L^+ may become exceedingly large and tend toward positive or negative infinity.

For case 1, we say that L^+ exists and has the value which it approaches, but for case 2 and case 3, we say that L^+ does not exist.

The same three possibilities also hold for L^-.

NOTE THE FOLLOWING SYMBOLISM: Let f(x) and a point x=a be given.

Let F = f(a)

Let L^+ = limit of f(x) as x approaches a from above.

Let L^- = limit of f(x) as x approaches a from below.

Let L = limit of f(x) as x approaches a.

NOTE THE FOLLOWING RULES FOR LIMITS:

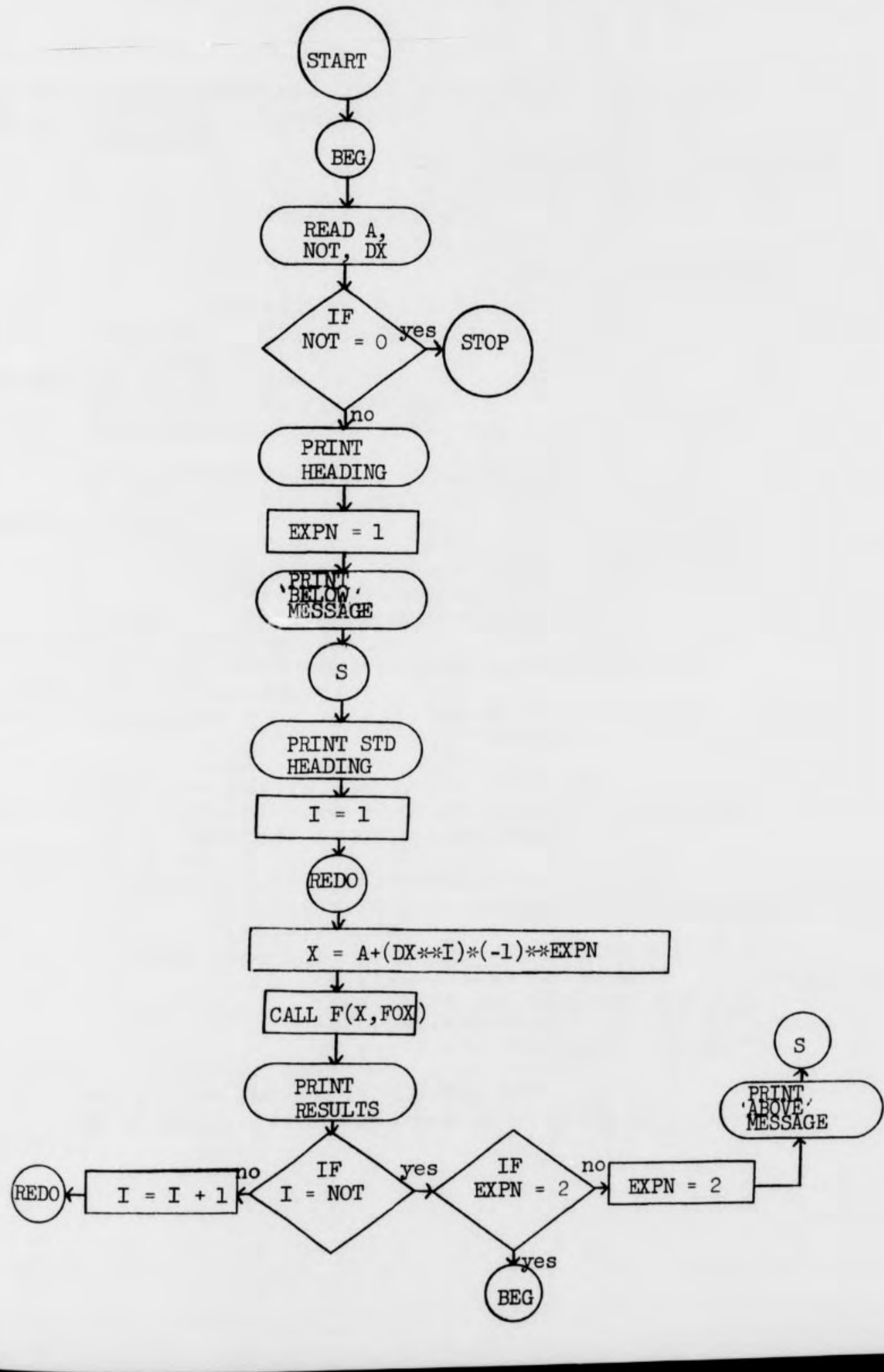
1. If L^+ or L^- fails to exist, the L does not exist.
2. If L^+ and L^- both exist, but $L^+ \neq L^-$, then L does not exist.
3. If L^+ and L^- both exist and are equal, then L exists, and has the same value as L^+ and L^-.

4. If $f(x)$ is undefined for some interval (b,a), then L^- does not exist and similarly L^+ does not exist if $f(x)$ is undefined for some interval (a,c).

3. EXAMPLE 1: Write a program which will help you guess if the limit of the function exists or not, and if so, approximate its value. Let $f(x) = x^{3/2} + x^{4/5}$ and let $a = 3$. First determine if L^+ exists, second, determine if L^- exists, then make a conclusion about L.

A. One possible flowchart for this problem is shown on the next page:

BASIC FLOWCHART FOR ALL EXAMPLES IN LESSON 16



B. A possible PL/C program based on the flowchart is shown below:

```
//EXAMPLE1 PLC ECS.UNCG.MA600237,LESSON16,OPT=NOBOUNDARY
LIMIT:     PROCEDURE OPTIONS(MAIN);
           DECLARE(A,                    /* POINT APPROACHED                    */
                  DX,                    /* CONSTANT RAISED TO POWERS           */
                   X,                    /* POINT APPROACHING                   */
                   FOX)FLOAT(16),/* F(X)                                        */
                 (I,                     /* COUNTER                             */
                   EXPN,                 /* EXPONENT FOR APPROACHING FROM
                                            ABOVE OR BELOW                      */
                  NOT)FIXED;             /* NUMBER OF TERMS APPROACHING A       */
           DECLARE F ENTRY(FLOAT(16),FLOAT(16));
F:         PROCEDURE(X,FOX);
/* DEFINITION OF FUNCTION */
           DECLARE(X,FOX) FLOAT(16);
           FOX=(X**(3/2))+X**(4/5);    /* CALCULATE F(X) */
END F;
BEG:       GET LIST(A,NOT,DX);    /* READ INPUT */
           IF NOT = 0 THEN STOP;    /* TEST FOR TERMINATION */
/* PRINT HEADING */
           PUT EDIT('ESTIMATING THE LIMIT AS X APPROACHES ',A,' FOR THE'
                 ,' FUNCTION F(X)=(X**(3/2))+(X**(4/5))')
                  (A,F(10,5),A,SKIP,A) PAGE;
           EXPN=1;    /* SET EXPONENT FOR BELOW    */
/* PRINT 'BELOW' MESSAGE  */
           PUT LIST('APPROACHING FROM BELOW') SKIP(2);
/* PRINT STANDARD HEADING */
START:     PUT EDIT(' I          A     +(  DX        ** I)=            X',
                    '                            F(X)')

                 (A,A) SKIP(2);
           I=1;    /* INITIALIZE I */
REDO:      X=A+(DX**I)*(-1)**EXPN;  /* CALCULATE NEW POINT */
           CALL F(X,FOX);  /* CALCULATE F(X)  */
/* PRINT RESULTS */
           PUT EDIT(I,A,'+(',DX,'**',I,')= ',X,FOX)
                  (F(2),X(3),F(9,5),A,F(9,5),A,F(2),A,F(13,9),X(5),
                   F(13,9)) SKIP;
           IF I=NOT THEN DO;    /* HAVE WE REACHED NEAR A */
                            IF EXPN=2 THEN GO TO BEG;    /* GO GET DATA */
                            EXPN=2; /* SET EXPONENT FOR ABOVE */
                            PUT LIST('APPROACHING FROM ABOVE') SKIP(5);
                            GO TO START; /* RETURN FOR ABOVE DATA */
                         END;
           I=I+1;  /* INCREMENT COUNTER */
           GO TO REDO;  /* RETURN FOR NEW POINTS */
END LIMIT;
*DATA
5    20         .5          2           20          .5         0      0
//
```

C. The output from the above program is shown below:

```
ESTIMATING THE LIMIT AS X APPROACHES    3.00000 FOR THE
 FUNCTION F(X)=(X**(3/2))+(X**(4/5))

APPROACHING FROM BELOW

 I        A      +( DX         ** I)=          X                F(X)
 1     3.00000+(  0.50000** 1)=    2.500000000      6.034250093
 2     3.00000+(  0.50000** 2)=    2.750000000      6.806650800
 3     3.00000+(  0.50000** 3)=    2.875000000      7.202410022
 4     3.00000+(  0.50000** 4)=    2.937500000      7.402624650
 5     3.00000+(  0.50000** 5)=    2.968750000      7.503509495
 6     3.00000+(  0.50000** 6)=    2.984375000      7.555795504
 7     3.00000+(  0.50000** 7)=    2.992187500      7.579074415
 8     3.00000+(  0.50000** 8)=    2.996093750      7.591722782
 9     3.00000+(  0.50000** 9)=    2.998046875      7.598049201
10     3.00000+(  0.50000**10)=    2.999023437      7.601212966
11     3.00000+(  0.50000**11)=    2.999511718      7.602794951
12     3.00000+(  0.50000**12)=    2.999755859      7.603586038
13     3.00000+(  0.50000**13)=    2.999877929      7.603981570
14     3.00000+(  0.50000**14)=    2.999938964      7.604179338
15     3.00000+(  0.50000**15)=    2.999969482      7.604278222
16     3.00000+(  0.50000**16)=    2.999984741      7.604327665
17     3.00000+(  0.50000**17)=    2.999992370      7.604352386
18     3.00000+(  0.50000**18)=    2.999996185      7.604364747
19     3.00000+(  0.50000**19)=    2.999998092      7.604370927
20     3.00000+(  0.50000**20)=    2.999999046      7.604374017

APPROACHING FROM ABOVE

 I        A      +( DX         ** I)=          X                F(X)
 1     3.00000+(  0.50000** 1)=    3.500000000      9.272197522
 2     3.00000+(  0.50000** 2)=    3.250000000      8.426496573
 3     3.00000+(  0.50000** 3)=    3.125000000      8.012441543
 4     3.00000+(  0.50000** 4)=    3.062500000      7.807658833
 5     3.00000+(  0.50000** 5)=    3.031250000      7.705625776
 6     3.00000+(  0.50000** 6)=    3.015625000      7.655055915
 7     3.00000+(  0.50000** 7)=    3.007812500      7.629703616
 8     3.00000+(  0.50000** 8)=    3.003906250      7.617037387
 9     3.00000+(  0.50000** 9)=    3.001953125      7.610706503
10     3.00000+(  0.50000**10)=    3.000976562      7.607541619
11     3.00000+(  0.50000**11)=    3.000488281      7.605959317
12     3.00000+(  0.50000**12)=    3.000244140      7.605168201
13     3.00000+(  0.50000**13)=    3.000122070      7.604772651
14     3.00000+(  0.50000**14)=    3.000061035      7.604574879
15     3.00000+(  0.50000**15)=    3.000030517      7.604475993
16     3.00000+(  0.50000**16)=    3.000015258      7.604426550
17     3.00000+(  0.50000**17)=    3.000007629      7.604401829
18     3.00000+(  0.50000**18)=    3.000003814      7.604389466
19     3.00000+(  0.50000**19)=    3.000001907      7.604383288
20     3.00000+(  0.50000**20)=    3.000000953      7.604380198
```

ESTIMATING THE LIMIT AS X APPROACHES 2.00000 FOR THE FUNCTION F(X)=(X**(3/2))+(X**(4/5))

APPROACHING FROM BELOW

I	A +(DX ** I)=	X	F(X)
1	2.00000+(0.50000** 1)=	1.500000000	3.220279174
2	2.00000+(0.50000** 2)=	1.750000000	3.879730078
3	2.00000+(0.50000** 3)=	1.875000000	4.220937374
4	2.00000+(0.50000** 4)=	1.937500000	4.394321708
5	2.00000+(0.50000** 5)=	1.968750000	4.481698634
6	2.00000+(0.50000** 6)=	1.984375000	4.525557052
7	2.00000+(0.50000** 7)=	1.992187500	4.547528561
8	2.00000+(0.50000** 8)=	1.996093750	4.558524888
9	2.00000+(0.50000** 9)=	1.998046875	4.564025691
10	2.00000+(0.50000**10)=	1.999023437	4.566776751
11	2.00000+(0.50000**11)=	1.999511718	4.568152446
12	2.00000+(0.50000**12)=	1.999755859	4.568840355
13	2.00000+(0.50000**13)=	1.999877929	4.569184289
14	2.00000+(0.50000**14)=	1.999938964	4.569356269
15	2.00000+(0.50000**15)=	1.999969482	4.569442260
16	2.00000+(0.50000**16)=	1.999984741	4.569485255
17	2.00000+(0.50000**17)=	1.999992370	4.569506753
18	2.00000+(0.50000**18)=	1.999996185	4.569517502
19	2.00000+(0.50000**19)=	1.999998092	4.569522876
20	2.00000+(0.50000**20)=	1.999999046	4.569525564

APPROACHING FROM ABOVE

I	A +(DX ** I)=	X	F(X)
1	2.00000+(0.50000** 1)=	2.500000000	6.034230095
2	2.00000+(0.50000** 2)=	2.250000000	5.288156750
3	2.00000+(0.50000** 3)=	2.125000000	4.925316435
4	2.00000+(0.50000** 4)=	2.062500000	4.740554617
5	2.00000+(0.50000** 5)=	2.031250000	4.657807760
6	2.00000+(0.50000** 6)=	2.015625000	4.613611942
7	2.00000+(0.50000** 7)=	2.007812500	4.591556059
8	2.00000+(0.50000** 8)=	2.003906250	4.580536643
9	2.00000+(0.50000** 9)=	2.001953125	4.575032568
10	2.00000+(0.50000**10)=	2.000976562	4.572280190
11	2.00000+(0.50000**11)=	2.000488281	4.570904166
12	2.00000+(0.50000**12)=	2.000244140	4.570216194
13	2.00000+(0.50000**13)=	2.000122070	4.569872219
14	2.00000+(0.50000**14)=	2.000061035	4.569700234
15	2.00000+(0.50000**15)=	2.000030517	4.569614242
16	2.00000+(0.50000**16)=	2.000015258	4.569571247
17	2.00000+(0.50000**17)=	2.000007629	4.569549749
18	2.00000+(0.50000**18)=	2.000003814	4.569539000
19	2.00000+(0.50000**19)=	2.000001907	4.569533625
20	2.00000+(0.50000**20)=	2.000000953	4.569530938

D. Below is a partial explanation for the above program:

Card 11 declares a subroutine called "F", with two variables, both with 16 places and floating decimal point.

Cards 12-16 are the subroutine "F".

Cards 23 and 31 can best be understood by examining what is needed.

From Above	From Below
$X = 3 + (1/2)^1$	$X = 3 - (1/2)^1$
$X = 3 + (1/2)^2$	$X = 3 - (1/2)^2$
$X = 3 + (1/2)^3$	$X = 3 - (1/2)^3$

The values of X have a general form: X = A + (DX**I)*(-1)**EXPN. Where A = 3 in this case, DX = 1/2, I is the exponent for 1/2, and -1**EXPN gives the + or - sign for either ABOVE or BELOW. Hence, in card 23, EXPN is set to 1, to produce the negative signs for the approach from below.

Card 30 initializes I, the exponent for 1/2 to I = 1.

Card 31 calculates the values for X.

Cards 37 - 42 contain a DO group. If I=NOT (number of terms, 20 in this case), then the DO group is executed, which then checks to see if the approach was from below or above (EXPN = 1 or 2). If EXPN = 1, then it proceeds to evaluate the limit from above. If EXPN = 2, then it returns to the label BEG to get new data.

Card 47 contains two sets of data: A=3, NOT=20, DX=.5 and A=2, NOT=20, DX=.5. Thus the limit is evaluated at two places.

Examine the output from the example. In order to determine L^-, twenty values for X were used, beginning with 2.5000000000 and finally, 2.999999046, so that clearly the values of X are approaching 3 from below. In the f(x) column, the values begin to "converge" toward 7.6043. Thus one would guess that L^- does indeed exist, and that its value is approximately $L^- = 7.6043$.

In the second part where X is approaching 3 from above, twenty values for X were used, beginning with 3.5000000000 and finally, 3.000000935. In the f(x) column, the values again begin to "converge" toward 7.6043. Thus one would guess that L^+ exists and has a value of approximately $L^+ = 7.6043$.

Since both L^+ and L^- exist (by approximation) and have the same (again by approximation) value, then one would conclude that

$$\lim_{x \to 3} x^{3/2} + x^{4/5} = 7.6043 \text{ (approximately)}$$

In this case, f(3) does exist, and equals approximately 7.6043.

4. For the following examples, the flowcharts and programs are essentially the same as in Example 1.

In each example, examine the output and answer these questions:

i. Does L^- exist, and if so, what is its value?

ii. Does L^+ exist, and if so, what is its value?

iii. Does $L^+ = L^-$?

iv. Does L exist, and if so, what is its value?

5. **EXAMPLE 2:** Estimate $\lim_{x \to a}$ [[sin x]] where a = π and $\pi/2$

ESTIMATING THE LIMIT AS X APPROACHES 3.14159 FOR THE
FUNCTION F(X)=STEP(SIN(X))

APPROACHING FROM BELOW

I	A +(DX ** I)=	X	F(X)
1	3.14159+(0.50000** 1)=	2.641592600	0.000000000
2	3.14159+(0.50000** 2)=	2.891592600	0.000000000
3	3.14159+(0.50000** 3)=	3.016592600	0.000000000
4	3.14159+(0.50000** 4)=	3.079092600	0.000000000
5	3.14159+(0.50000** 5)=	3.110342600	0.000000000
6	3.14159+(0.50000** 6)=	3.125967600	0.000000000
7	3.14159+(0.50000** 7)=	3.133780100	0.000000000
8	3.14159+(0.50000** 8)=	3.137686350	0.000000000
9	3.14159+(0.50000** 9)=	3.139639475	0.000000000

10	3.14159+(	0.50000**10)=	3.140616037	0.000000000
11	3.14159+(	0.50000**11)=	3.141104318	0.000000000
12	3.14159+(	0.50000**12)=	3.141348459	0.000000000
13	3.14159+(	0.50000**13)=	3.141470529	0.000000000
14	3.14159+(	0.50000**14)=	3.141531564	0.000000000
15	3.14159+(	0.50000**15)=	3.141562082	0.000000000
16	3.14159+(	0.50000**16)=	3.141577341	0.000000000
17	3.14159+(	0.50000**17)=	3.141584970	0.000000000
18	3.14159+(	0.50000**18)=	3.141588785	0.000000000
19	3.14159+(	0.50000**19)=	3.141590692	0.000000000
20	3.14159+(	0.50000**20)=	3.141591646	0.000000000

APPROACHING FROM ABOVE

I	A +(	DX ** I)=	X	F(X)
1	3.14159+(	0.50000** 1)=	3.641592600	-1.000000000
2	3.14159+(	0.50000** 2)=	3.391592600	-1.000000000
3	3.14159+(	0.50000** 3)=	3.266592600	-1.000000000
4	3.14159+(	0.50000** 4)=	3.204092600	-1.000000000
5	3.14159+(	0.50000** 5)=	3.172842600	-1.000000000
6	3.14159+(	0.50000** 6)=	3.157217600	-1.000000000
7	3.14159+(	0.50000** 7)=	3.149405100	-1.000000000
8	3.14159+(	0.50000** 8)=	3.145498850	-1.000000000
9	3.14159+(	0.50000** 9)=	3.143545725	-1.000000000
10	3.14159+(	0.50000**10)=	3.142569162	-1.000000000
11	3.14159+(	0.50000**11)=	3.142080881	-1.000000000
12	3.14159+(	0.50000**12)=	3.141836740	-1.000000000
13	3.14159+(	0.50000**13)=	3.141714670	-1.000000000
14	3.14159+(	0.50000**14)=	3.141653655	-1.000000000
15	3.14159+(	0.50000**15)=	3.141623117	-1.000000000
16	3.14159+(	0.50000**16)=	3.141607858	-1.000000000
17	3.14159+(	0.50000**17)=	3.141600229	-1.000000000
18	3.14159+(	0.50000**18)=	3.141596414	-1.000000000
19	3.14159+(	0.50000**19)=	3.141594507	-1.000000000
20	3.14159+(	0.50000**20)=	3.141593553	-1.000000000

EXAMPLE 3: Estimate $\lim_{x \to b} \sqrt{x^2 - 1}$ **where b = -1**

```
ESTIMATING THE LIMIT AS X APPROACHES   -1.00000 FOR THE
 FUNCTION F(X)= SQRT((X**2) -1)

APPROACHING FROM BELOW

 I        A    +( DX        ** I)=          X                    F(X)
 1   -1.00000+(  0.50000** 1)=  -1.500000000          1.118033988
 2   -1.00000+(  0.50000** 2)=  -1.250000000          0.750000000
 3   -1.00000+(  0.50000** 3)=  -1.125000000          0.515388203
 4   -1.00000+(  0.50000** 4)=  -1.062500000          0.359035165
 5   -1.00000+(  0.50000** 5)=  -1.031250000          0.251945554
 6   -1.00000+(  0.50000** 6)=  -1.015625000          0.177465885
 7   -1.00000+(  0.50000** 7)=  -1.007812500          0.125243902
 8   -1.00000+(  0.50000** 8)=  -1.003906250          0.088474622
 9   -1.00000+(  0.50000** 9)=  -1.001953125          0.062530510
10   -1.00000+(  0.50000**10)=  -1.000976562          0.044204962
11   -1.00000+(  0.50000**11)=  -1.000488281          0.031253814
12   -1.00000+(  0.50000**12)=  -1.000244140          0.022098435
13   -1.00000+(  0.50000**13)=  -1.000122070          0.015625476
14   -1.00000+(  0.50000**14)=  -1.000061035          0.011048712
15   -1.00000+(  0.50000**15)=  -1.000030517          0.007812559
16   -1.00000+(  0.50000**16)=  -1.000015258          0.005524292
17   -1.00000+(  0.50000**17)=  -1.000007629          0.003906257
18   -1.00000+(  0.50000**18)=  -1.000003814          0.002762156
19   -1.00000+(  0.50000**19)=  -1.000001907          0.001953125
20   -1.00000+(  0.50000**20)=  -1.000000953          0.001381068

APPROACHING FROM ABOVE

 I        A    +( DX        ** I)=          X                    F(X)
****** IN STMT    6  ERROR   EX8D   THE ARGUMENT IS NEGATIVE ([illegible]).

 1   -1.00000+(  0.50000** 1)=  -0.500000000         0.866025405
****** IN STMT    6  ERROR   EX8D   THE ARGUMENT IS NEGATIVE ([illegible]).

 2   -1.00000+(  0.50000** 2)=  -0.750000000         0.661437827
****** IN STMT    6  ERROR   EX8D   THE ARGUMENT IS NEGATIVE ([illegible]).

 3   -1.00000+(  0.50000** 3)=  -0.875000000         0.484122291
****** IN STMT    6  ERROR   EX8D   THE ARGUMENT IS NEGATIVE ([illegible]).

 4   -1.00000+(  0.50000** 4)=  -0.937500000         0.347985272
****** IN STMT    6  ERROR   EX8D   THE ARGUMENT IS NEGATIVE ([illegible]).

 5   -1.00000+(  0.50000** 5)=  -0.968750000         0.248039185
****** IN STMT    6  ERROR   EX8D   THE ARGUMENT IS NEGATIVE ([illegible]).
```

```
 6    -1.00000+(  0.50000** 6)=  -0.984375000        0.176084807
****** IN STMT    6  ERROR  EX8D  THE ARGUMENT IS NEGATIVE (###).

 7    -1.00000+(  0.50000** 7)=  -0.992187500        0.124755620
****** IN STMT    6  ERROR  EX8D  THE ARGUMENT IS NEGATIVE (###).

 8    -1.00000+(  0.50000** 8)=  -0.996093750        0.088301988
****** IN STMT    6  ERROR  EX8D  THE ARGUMENT IS NEGATIVE (###).

 9    -1.00000+(  0.50000** 9)=  -0.998046875        0.062469474
****** IN STMT    6  ERROR  EX8D  THE ARGUMENT IS NEGATIVE (###).

10    -1.00000+(  0.50000**10)=  -0.999023437        0.044163382
****** IN STMT    6  ERROR  EX8D  THE ARGUMENT IS NEGATIVE (###).

11    -1.00000+(  0.50000**11)=  -0.999511718        0.031246185
****** IN STMT    6  ERROR  EX8D  THE ARGUMENT IS NEGATIVE (###).

12    -1.00000+(  0.50000**12)=  -0.999755859        0.022095738
****** IN STMT    6  ERROR  EX8D  THE ARGUMENT IS NEGATIVE (###).

13    -1.00000+(  0.50000**13)=  -0.999877929        0.015624523
****** IN STMT    6  ERROR  EX8D  THE ARGUMENT IS NEGATIVE (###).

14    -1.00000+(  0.50000**14)=  -0.999938964        0.011048374
****** IN STMT    6  ERROR  EX8D  THE ARGUMENT IS NEGATIVE (###).

15    -1.00000+(  0.50000**15)=  -0.999969482        0.007812440
****** IN STMT    6  ERROR  EX8D  THE ARGUMENT IS NEGATIVE (###).

16    -1.00000+(  0.50000**16)=  -0.999984741        0.005524250
****** IN STMT    6  ERROR  EX8D  THE ARGUMENT IS NEGATIVE (###).

17    -1.00000+(  0.50000**17)=  -0.999992370        0.003906242
****** IN STMT    6  ERROR  EX8D  THE ARGUMENT IS NEGATIVE (###).

18    -1.00000+(  0.50000**18)=  -0.999996185        0.002762155
****** IN STMT    6  ERROR  EX8D  THE ARGUMENT IS NEGATIVE (###).

19    -1.00000+(  0.50000**19)=  -0.999998092        0.001953124
****** IN STMT    6  ERROR  EX8D  THE ARGUMENT IS NEGATIVE (###).

20    -1.00000+(  0.50000**20)=  -0.999999046        0.001381067
```

7. **EXAMPLE 4:** **Estimate** $\lim_{x \to a}$ **(sin 2x)/(sin x)** where a = 0

```
ESTIMATING THE LIMIT AS X APPROACHES    0.00000 FOR THE
 FUNCTION F(X)= (SIN(2*X))/(SIN(X))

APPROACHING FROM BELOW

 I        A     +(  DX        ** I)=            X                    F(X)
 1     0.00000+(  0.50000** 1)=   -0.500000000           1.755165123
 2     0.00000+(  0.50000** 2)=   -0.250000000           1.937824843
 3     0.00000+(  0.50000** 3)=   -0.125000000           1.984395334
 4     0.00000+(  0.50000** 4)=   -0.062500000           1.996095021
 5     0.00000+(  0.50000** 5)=   -0.031250000           1.999023516
 6     0.00000+(  0.50000** 6)=   -0.015625000           1.999755864
 7     0.00000+(  0.50000** 7)=   -0.007812500           1.999938965
 8     0.00000+(  0.50000** 8)=   -0.003906250           1.999984741
 9     0.00000+(  0.50000** 9)=   -0.001953125           1.999996185
10     0.00000+(  0.50000**10)=   -0.000976562           1.999999046
11     0.00000+(  0.50000**11)=   -0.000488281           1.999999761
12     0.00000+(  0.50000**12)=   -0.000244140           1.999999940
13     0.00000+(  0.50000**13)=   -0.000122070           1.999999985
14     0.00000+(  0.50000**14)=   -0.000061035           1.999999996
15     0.00000+(  0.50000**15)=   -0.000030517           1.999999999
16     0.00000+(  0.50000**16)=   -0.000015258           1.999999999
17     0.00000+(  0.50000**17)=   -0.000007629           1.999999999
18     0.00000+(  0.50000**18)=   -0.000003814           1.999999999
19     0.00000+(  0.50000**19)=   -0.000001907           1.999999999
20     0.00000+(  0.50000**20)=   -0.000000953           1.999999999

APPROACHING FROM ABOVE

 I        A     +(  DX        ** I)=            X                    F(X)
 1     0.00000+(  0.50000** 1)=    0.500000000           1.755165123
 2     0.00000+(  0.50000** 2)=    0.250000000           1.937824843
 3     0.00000+(  0.50000** 3)=    0.125000000           1.984395334
 4     0.00000+(  0.50000** 4)=    0.062500000           1.996095021
 5     0.00000+(  0.50000** 5)=    0.031250000           1.999023516
 6     0.00000+(  0.50000** 6)=    0.015625000           1.999755864
 7     0.00000+(  0.50000** 7)=    0.007812500           1.999938965
 8     0.00000+(  0.50000** 8)=    0.003906250           1.999984741
 9     0.00000+(  0.50000** 9)=    0.001953125           1.999996185
10     0.00000+(  0.50000**10)=    0.000976562           1.999999046
11     0.00000+(  0.50000**11)=    0.000488281           1.999999761
12     0.00000+(  0.50000**12)=    0.000244140           1.999999940
13     0.00000+(  0.50000**13)=    0.000122070           1.999999985
14     0.00000+(  0.50000**14)=    0.000061035           1.999999996
15     0.00000+(  0.50000**15)=    0.000030517           1.999999999
16     0.00000+(  0.50000**16)=    0.000015258           1.999999999
17     0.00000+(  0.50000**17)=    0.000007629           1.999999999
18     0.00000+(  0.50000**18)=    0.000003814           1.999999999
19     0.00000+(  0.50000**19)=    0.000001907           1.999999999
20     0.00000+(  0.50000**20)=    0.000000953           1.999999999
```

8. **EXAMPLE 5: Estimate $\lim_{x \to b} LOG_{10}x$ where b = 0**

```
ESTIMATING THE LIMIT AS X APPROACHES     0.00000 FOR THE
 FUNCTION F(X)= LOG10(X)

APPROACHING FROM BELOW

 I        A      +(  DX        **  I)=              X
****** IN STMT     6   ERROR   EX8F LOG10 ARGUMENT (###)

 1      0.00000+(   0.50000**  1)=  -0.500000000        1
****** IN STMT     6   ERROR   EX8F LOG10 ARGUMENT (###)

 2      0.00000+(   0.50000**  2)=  -0.250000000        1
****** IN STMT     6   ERROR   EX8F LOG10 ARGUMENT (###)

 3      0.00000+(   0.50000**  3)=  -0.125000000        1
****** IN STMT     6   ERROR   EX8F LOG10 ARGUMENT (###)

 4      0.00000+(   0.50000**  4)=  -0.062500000        1
****** IN STMT     6   ERROR   EX8F LOG10 ARGUMENT (###)

 5      0.00000+(   0.50000**  5)=  -0.031250000        1
****** IN STMT     6   ERROR   EX8F LOG10 ARGUMENT (###)

 6      0.00000+(   0.50000**  6)=  -0.015625000        1
****** IN STMT     6   ERROR   EX8F LOG10 ARGUMENT (###)

 7      0.00000+(   0.50000**  7)=  -0.007812500        1
****** IN STMT     6   ERROR   EX8F LOG10 ARGUMENT (###)

 8      0.00000+(   0.50000**  8)=  -0.003906250        1
****** IN STMT     6   ERROR   EX8F LOG10 ARGUMENT (###)

 9      0.00000+(   0.50000**  9)=  -0.001953125        1
****** IN STMT     6   ERROR   EX8F LOG10 ARGUMENT (###)

10      0.00000+(   0.50000** 10)=  -0.000976562        1
****** IN STMT     6   ERROR   EX8F LOG10 ARGUMENT (###)

11      0.00000+(   0.50000** 11)=  -0.000488281        1
****** IN STMT     6   ERROR   EX8F LOG10 ARGUMENT (###)

12      0.00000+(   0.50000** 12)=  -0.000244140        1
****** IN STMT     6   ERROR   EX8F LOG10 ARGUMENT (###)

13      0.00000+(   0.50000** 13)=  -0.000122070        1
****** IN STMT     6   ERROR   EX8F LOG10 ARGUMENT (###)

14      0.00000+(   0.50000** 14)=  -0.000061035        1
****** IN STMT     6   ERROR   EX8F LOG10 ARGUMENT (###)
```

```
15      0.00000+(   0.50000**15)=  -0.000030517         1
****** IN STMT      6  ERROR   EX8F LOG10 ARGUMENT (###)

16      0.00000+(   0.50000**16)=  -0.000015258         1
****** IN STMT      6  ERROR   EX8F LOG10 ARGUMENT (###)

17      0.00000+(   0.50000**17)=  -0.000007629         1
****** IN STMT      6  ERROR   EX8F LOG10 ARGUMENT (###)

18      0.00000+(   0.50000**18)=  -0.000003814         1
****** IN STMT      6  ERROR   EX8F LOG10 ARGUMENT (###)

19      0.00000+(   0.50000**19)=  -0.000001907         1
****** IN STMT      6  ERROR   EX8F LOG10 ARGUMENT (###)

20      0.00000+(   0.50000**20)=  -0.000000953         1
```

APPROACHING FROM ABOVE

I	A +(	DX ** I)=	X	F(X)
1	0.00000+(	0.50000** 1)=	0.500000000	-0.301029995
2	0.00000+(	0.50000** 2)=	0.250000000	-0.602059991
3	0.00000+(	0.50000** 3)=	0.125000000	-0.903089986
4	0.00000+(	0.50000** 4)=	0.062500000	-1.204119982
5	0.00000+(	0.50000** 5)=	0.031250000	-1.505149978
6	0.00000+(	0.50000** 6)=	0.015625000	-1.806179973
7	0.00000+(	0.50000** 7)=	0.007812500	-2.107209969
8	0.00000+(	0.50000** 8)=	0.003906250	-2.408239965
9	0.00000+(	0.50000** 9)=	0.001953125	-2.709269960
10	0.00000+(	0.50000**10)=	0.000976562	-3.010299956
11	0.00000+(	0.50000**11)=	0.000488281	-3.311329952
12	0.00000+(	0.50000**12)=	0.000244140	-3.612359947
13	0.00000+(	0.50000**13)=	0.000122070	-3.913389943
14	0.00000+(	0.50000**14)=	0.000061035	-4.214419939
15	0.00000+(	0.50000**15)=	0.000030517	-4.515449934
16	0.00000+(	0.50000**16)=	0.000015258	-4.816479930
17	0.00000+(	0.50000**17)=	0.000007629	-5.117509926
18	0.00000+(	0.50000**18)=	0.000003814	-5.418539921
19	0.00000+(	0.50000**19)=	0.000001907	-5.719569917
20	0.00000+(	0.50000**20)=	0.000000953	-6.020599913

9. STUDENT EXERCISES: For the following functions and values of a, write a program which will help you guess if the limits exist or not, and if so, determine their values. For each function, examine the output and make an observation about L^+, L^-, L, and F(a).

a. $f(x) = x^{2/3} + x^{3/2}$ a = 2

b. $f(x) = 2^{x/2}$ a = 1/2

c. $f(x) = x - [[x]]$ a = 0

d. $f(x) = x^x$ a = 0

e. $f(x) = (1 + x)^{1/x}$ a = 0

f. $f(x) = \sqrt{6-x-x^2}$ a = 0

g. $f(x) = \sqrt{6-x-x^2}$ a = 2

h. $f(x) = 1/(x^2-2x+1)$ a = 1

i. $f(x) = (x^2-7x+12)/(x-3)$ a = 3

j. $f(x) = (x^2-5x+6)/(x^2-4)$ a = 2

Lesson 17 LIMITS OF FUNCTIONS: DELTA - EPSILON APPROACH

1. PURPOSE: The objective of this lesson is to understand the delta-epsilon approach to the limit of a function. In the previous lessons, an intuitive approach to limits was given, but this was very imprecise. The delta-epsilon definition of a limit provides a very precise method for determining whether a function has a limit or not. One must always keep in mind that the computer can not always do in practice what can be done in theory, hence it can not actually PROVE that a function has a limit. It can provide some powerful evidence, and in most cases the conclusions you make based on this evidence will be correct.

2. THEORY: Remember from previous lessons that when one writes

$$\lim_{x \to a} f(x) = L$$

intuitively one means that when the variable "x" gets close to the value "a", then f(x) approaches the value "L". But how close must f(x) get to L before one is satisfied they are "close enough"? The intuitive understanding of limits leaves many questions unanswered, hence the delta-epsilon definition is presented.

Consider the following definitions. The symbolism applies to this manual only.

AE = the "actual error". This is the difference between the value of the function at x and the given limit, L. Hence, AE = $|f(x)-L|$. To say that f(x) approaches the value L is equivalent to saying that AE gets very small (approaches zero).

E (epsilon) = the "specified error". E is simply a positive number, which is selected by an arbitrary choice by the person. Usually E is very small, and the only restriction is that E>0. One usually demands that the "actual error" be smaller than the "specified error", that is, $|f(x) - L| < E$.

AS = the "actual separation". This is the difference between the variable "x" and the value "a". Hence, AS = $|x - a|$. There is one restriction on AS. Since the function may be undefined at x=a, the condition $x \neq a$ is imposed, which is equivalent to 0<AS.

D(delta) = the "critical separation". This is simply a positive number which must be discovered or claculated by the person. D depends upon the function, the point a, and upon the choice of E. Usually, one wants the "actual separation" smaller than the "critical separation", that is, $0<|x - a|<D$.

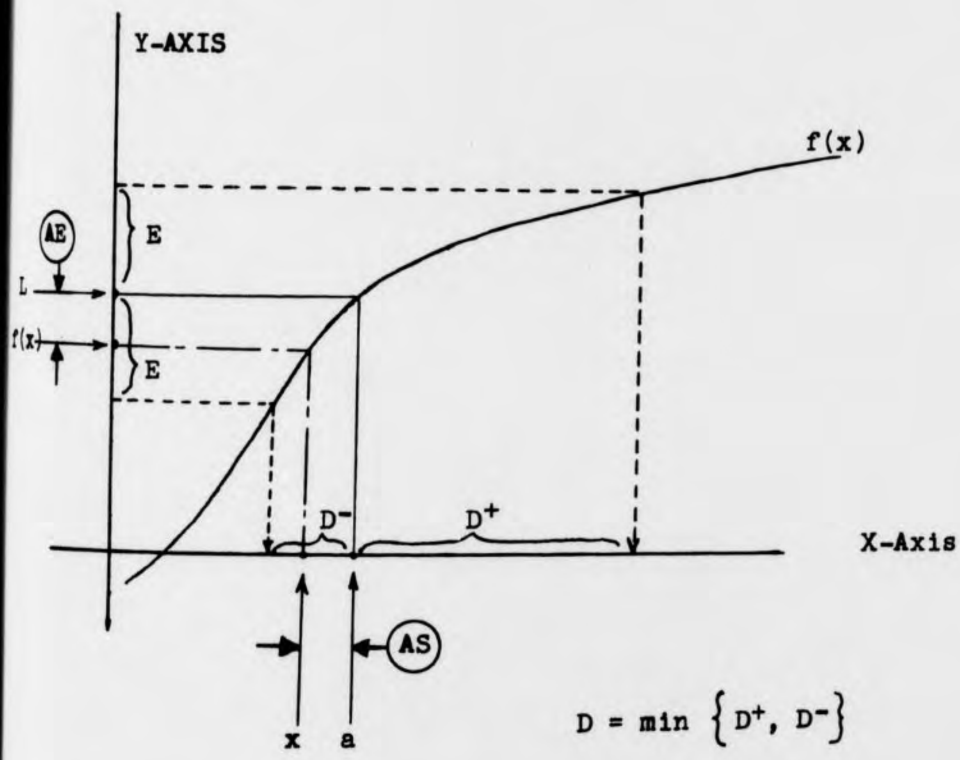
DELTA - EPSILON DEFINITION FOR THE LIMIT OF A FUNCTION

$\lim_{x \to a} f(x) = L$ means that given any specified error, E > 0, no matter how small, it is always possible to find a critical separation, D > 0, so that the actual error will be less than the specified error, $|f(x) - L|<E$, whenever the actual separation is smaller than the critical separation, $0<|x - a|<D$. In actual practice, f(x), a, L, and E are given values. One must use algebraic processes to find a "D" which will satisfy the definition. For a particular value of E, the computer can determine a "D" which will work, but the difficulty arises since the definition spacifies that a "D" can be found for ALL E>0, hence the computer cannot prove that a limit really exists.

In less mathematical language, E represents the "tolerance" or "error" which one will allow between f(x) and L. In order for the difference to be smaller than E, then x must be "close" to a. D represents how "close" it must be, that is, D is the maximum allowable separation in order that the difference be within the tolerence specified.

When trying to determine D, one may discover that separation "above" the point "a" and the separation "below" the point "a" may be different distances. The symbols D^+ and D^- will be used to represent the "critical separation" above or below the value "a". Then D = minimum $\left\{ D^+, D^- \right\}$.

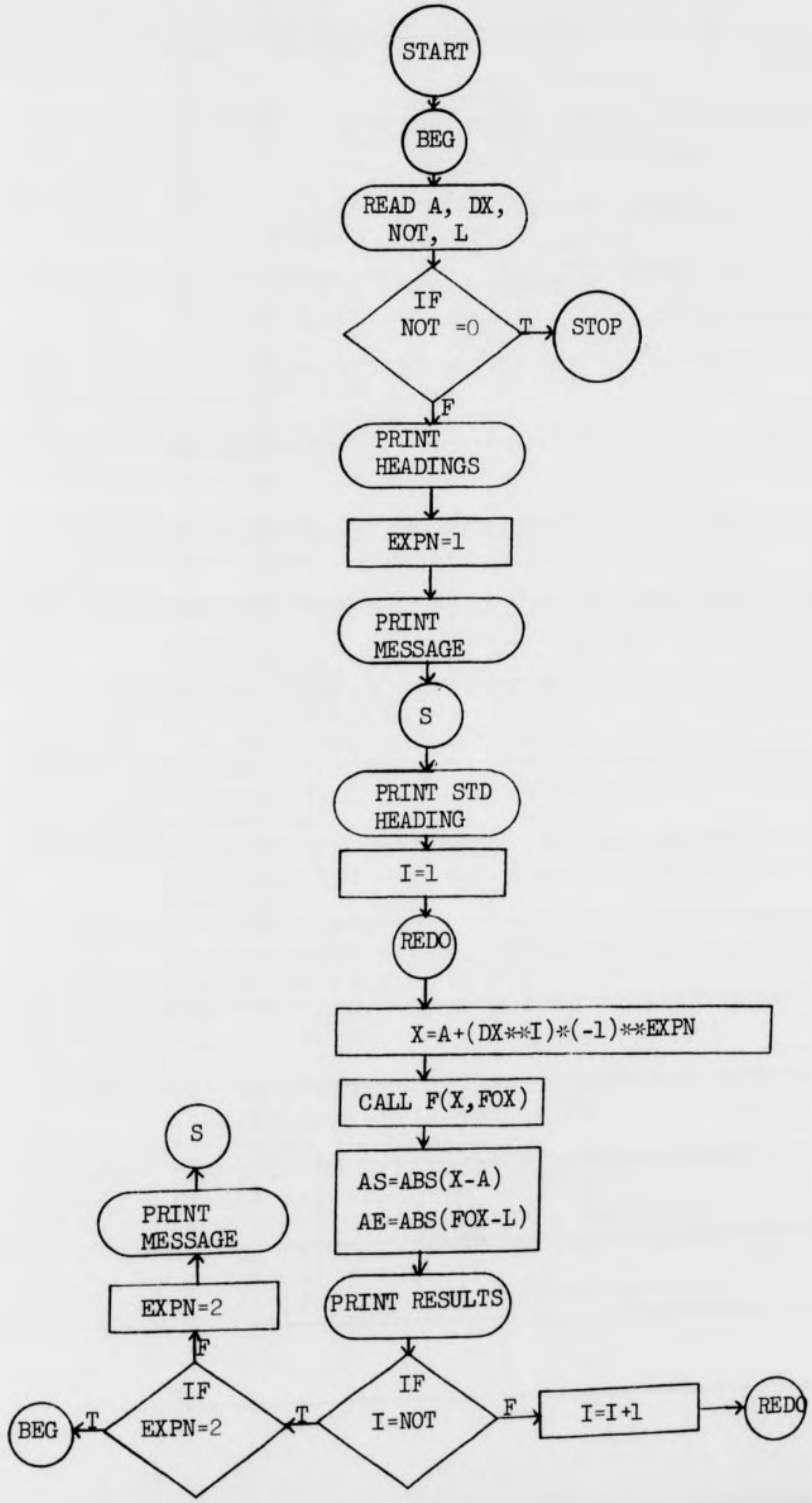
The geometric interpretation of the various symbols used in evaluating limits is shown in the figure below.



Geometric interpretation of E, AE, AS, D^+, D^-, and D.

3. EXAMPLE 1: Consider the function f(x) = (sin x)/x + 1/(x+1). This function is continuous and converges to the limit of 2 as x approaches zero. Write a program which will assist you in finding a D (delta) for a specific E (epsilon). This may be done by printing out tables for X, |X - A|, f(x), and |f(X)-L|. Choose a=0 and L=2. From this table, pick a value for E, then find the corresponding D+ and D^-, then determine D.

A. One flowchart for Example 1 is shown below.



B. A possible PL/C program based upon the flowchart is shown below:

```
//EXAMPLE1 PLC ECS.UNCG.MA600237,LESSON17,OPT=NOBOUNDARY
EPDL:      PROCEDURE OPTIONS(MAIN);
           DECLARE (A,                /* POINT APPROACHED                    */
                    DX,               /* CONSTANT RAISED TO POWERS           */
                    L,                /* GUESS OF LIMIT                      */
                    X,                /* POINT APPROACHING                   */
                    FOX,              /* F(X)                                */
                    AS ,              /* ACTUAL SEPARATION                   */
                    AE)FLOAT(16),     /* ACTUAL ERROR                        */
                   (NOT,              /* NUMBER OF POINTS APPROACHING A      */
                    I,                /* COUNTER                             */
                    EXPN)FIXED;       /* EXPONENT FOR APPROACHING FROM
                                         ABOVE OR BELOW                      */
           DECLARE F ENTRY(FLOAT(16),FLOAT(16));
F:         PROCEDURE(X,FOX);
           DECLARE(X,FOX)FLOAT(16);
/* DEFINITION OF FUNCTION */
           FOX=(SIN(X)/X)+(1/(X+ 1)) ;
END F;
BEG:       GET LIST(A,DX,NOT,L);  /* READ INPUT VARIABLES */
           IF NOT=0 THEN STOP;    /* TERMINATION CHECK */
/* PRINT HEADING */
           PUT EDIT('USING EPSILON-DELTA DEFINITION TO DETERMINE THE',
                  ' LIMIT AS X APPROACHES',A,' FOR THE FUNCTION ',
                  'F(X)=(SIN(X)/X)+(1/(X+ 1)).')
                  (PAGE,A,SKIP,A,F(10,5),SKIP,A,A);
           PUT EDIT(' THE GUESS OF THE LIMIT IS ',L)  (A,F(7,5))SKIP(2);
           EXPN=1;    /* SET EXPONENT FOR BELOW    */
/* PRINT 'BELOW' MESSAGE */
           PUT LIST('APPROACHING FROM BELOW') SKIP(2);
/* PRINT STANDARD HEADING */
START:     PUT EDIT('I','X','|X-A|','F(X)','|F(X)-',L,'|')
                 (X(1),A,X(11),A,X(10),A,X(10),A,X(10),A,F(7,5),A)
                 SKIP(2);
           I=1;    /* INITIALIZE I */
REDO:      X=A+(DX**I)*(-1)**EXPN;  /* CALCULATE NEW POINT */
           CALL F(X,FOX);  /* CALCULATE F(X)  */
           AS =ABS(X-A);   /* CALCULATE ACTUAL SEPARATION */
           AE=ABS(FOX-L);  /* CALCULATE ACTUAL ERROR */
/* PRINT RESULTS */
           PUT EDIT(I,X,AS ,FOX,AE) (F(2),X(2),F(13,8),X(1),F(14,8),X(2)
                 ,F(14,8),X(2),F(14,8)) SKIP;
           IF I¬= NOT THEN GO TO INCR;
           IF EXPN=2 THEN GO TO BEG;  /* GO GET MORE DATA */
           EXPN=2;  /* SET EXPONENT FOR ABOVE */
           PUT LIST('APPROACHING FROM ABOVE') SKIP(3);
           GO TO START; /* RETURN FOR ABOVE CALCULATIONS */
INCR:      I=I+1; /* INCREMENT I */
           GO TO REDO;  /* CALCULATE NEXT POINT */
END EPDL;
*DATA
0  .5  20  2  0  0  0  0
//
```

Lesson 17 (cont)

C. The output from the above program is shown below:

USING EPSILON-DELTA DEFINITION TO DETERMINE THE
LIMIT AS X APPROACHES 0.00000
FOR THE FUNCTION F(X)=(SIN(X)/X)+(1/(X+ 1)).

THE GUESS OF THE LIMIT IS 2.00000

APPROACHING FROM BELOW

I	X	|X-A|	F(X)	|F(X)-2.00000|
1	-0.50000000	0.50000000	2.95865107	0.95865107
2	-0.25000000	0.25000000	2.32294917	0.32294917
3	-0.12500000	0.12500000	2.14025500	0.14025500
4	-0.06250000	0.06250000	2.06601575	0.06601575
5	-0.03125000	0.03125000	2.03209531	0.03209531
6	-0.01562500	0.01562500	2.01583232	0.01583232
7	-0.00781250	0.00781250	2.00786384	0.00786384
8	-0.00390625	0.00390625	2.00391907	0.00391907
9	-0.00195312	0.00195312	2.00195651	0.00195651
10	-0.00097656	0.00097656	2.00097755	0.00097755
11	-0.00048828	0.00048828	2.00048848	0.00048848
12	-0.00024414	0.00024414	2.00024419	0.00024419
13	-0.00012207	0.00012207	2.00012208	0.00012208
14	-0.00006103	0.00006103	2.00006103	0.00006103
15	-0.00003051	0.00003051	2.00003051	0.00003051
16	-0.00001525	0.00001525	2.00001525	0.00001525
17	-0.00000762	0.00000762	2.00000762	0.00000762
18	-0.00000381	0.00000381	2.00000381	0.00000381
19	-0.00000190	0.00000190	2.00000190	0.00000190
20	-0.00000095	0.00000095	2.00000095	0.00000095

APPROACHING FROM ABOVE

I	X	|X-A|	F(X)	|F(X)-2.00000|
1	0.50000000	0.50000000	1.62551774	0.37448225
2	0.25000000	0.25000000	1.78961585	0.21038416
3	0.12500000	0.12500000	1.88628675	0.11371324
4	0.06250000	0.06250000	1.94052555	0.05947444
5	0.03125000	0.03125000	1.96953421	0.03046578
6	0.01562500	0.01562500	1.98457469	0.01542530
7	0.00781250	0.00781250	1.99223788	0.00776211
8	0.00390625	0.00390625	1.99610640	0.00389359
9	0.00195312	0.00195312	1.99805004	0.00194995
10	0.00097656	0.00097656	1.99902423	0.00097576
11	0.00048828	0.00048828	1.99951191	0.00048808
12	0.00024414	0.00024414	1.99975590	0.00024409
13	0.00012207	0.00012207	1.99987796	0.00012203
14	0.00006103	0.00006103	1.99993896	0.00006103
15	0.00003051	0.00003051	1.99996946	0.00003051
16	0.00001525	0.00001525	1.99998476	0.00001525
17	0.00000762	0.00000762	1.99999237	0.00000762
18	0.00000381	0.00000381	1.99999618	0.00000381
19	0.00000190	0.00000190	1.99999809	0.00000190
20	0.00000095	0.00000095	1.99999904	0.00000095

D. A partial explanation of EXAMPLE 1 is given below.

Cards 15-19 contain the subroutine, F, which defines the function.

Cards 22-27 contain instructions for printing the headings.

Cards 32-34 print the column headings.

Card 36 calculates a new point for x. This is done exactly as the points in lesson 16.

Cards 41 and 42 are the output format.

Card 52 is the data card, where a=0, EX=.5, NOT=20, and L=2.

Observe the output from EXAMPLE 1. Suppose one decides on a value of E = .25 as a start. In the column where the limit is approaching from BELOW, one looks for |f(x)-2.000000| less than E=.25. In this case, it is the third number down the list, .14025500, then corresponding to this value, look across under the |x-A| column for the D^- value 0.125.

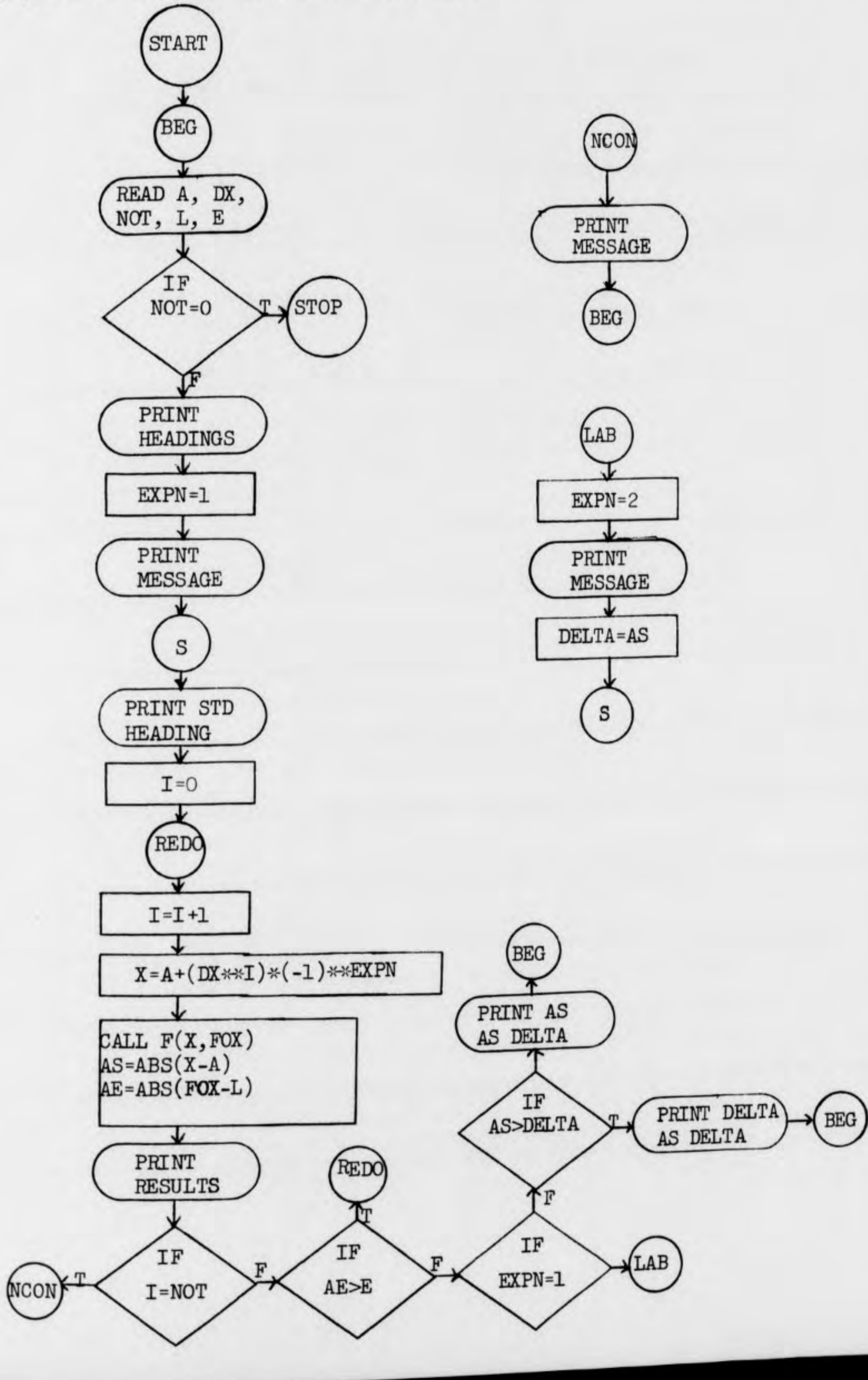
Then in the part where the limit is approaching from ABOVE, one looks for |f(x)-2.00000| less than E=.25. In this case, it is the second number down the list, 0.21038416, and the corresponding value in the |x-A| column for D^+ is 0.25000.

Therefore, for an E=.25, the corresponding $D=\min\left\{.125, .250\right\}$ which is 0.125. Hence, whenever the actual separation is AS<.125, then the actual error, AE<.25.

If one should pick another value for epsilon, say E=.0001, then using the output, one could find the corresponding D=.00006103 will work (or any value smaller than this). Observe that as |f(x)-2.000| gets very small, D^+ and D^- have approximately the same values.

4. EXAMPLE 2: Consider the function $f(x)=\sqrt{x+1}$ as x approaches 3.00000. This function is continuous and the limit converges to a value of L=2.00000. Write a program which will find a possible value for DELTA when given values of EPSILON as data. In this case, use E=.01, .0001, and .000001 and determine the corresponding values for D.

A. One flowchart for EXAMPLE 2 is shown below.



B. A possible PL/C program based upon the above flowchart is shown below:

```
//EXAMPLE2 PLC ECS.UNCG.MA600237,LESSON17,OPT=NOBOUNDARY
EPDL:      PROCEDURE OPTIONS(MAIN);
           DECLARE (A,                 /* POINT APPROACHED               */
                    DX,                /* CONSTANT RAISED TO POWERS      */
                    L,                 /* GUESS OF LIMIT                 */
                    X,                 /* POINT APPROACHING              */
                    FOX,               /* F(X)                           */
                    DELTA,             /* DELTA FOR BELOW THEN FINAL     */
                    E,                 /* EPSILON                        */
                    AS ,               /* ACTUAL SEPARATION              */
                    AE)FLOAT(16),      /* ACTUAL ERROR                   */
                   (NOT,               /* NUMBER OF ITERATIONS           */
                    I,                 /* COUNTER                        */
                    EXPN)FIXED;        /* EXPONENT FOR APPROACHING FROM
                                          ABOVE OR BELOW                 */
           DECLARE F ENTRY(FLOAT(16),FLOAT(16));
F:         PROCEDURE(X,FOX);
           DECLARE(X,FOX)FLOAT(16);
/* DEFINITION OF FUNCTION */
           FOX=SQRT(X+1);
END F;
BEG:       GET LIST(A,DX,NOT,L,E);  /* READ INPUT VARIABLES */
           IF NOT=0 THEN STOP;   /* TERMINATION CHECK */
/* PRINT HEADING */
           PUT EDIT('USING EPSILON-DELTA DEFINITION TO DETERMINE THE',
                    ' LIMIT AS X APPROACHES',A,' FOR THE FUNCTION ',
                    'F(X)=SQRT(X+1)')
                    (PAGE,A,SKIP,A,F(10,5),SKIP,A,A);
           PUT EDIT('THE GUESS OF THE LIMIT IS ',L,' WITH EPSILON=',E)
                    (A,F(7,5),A,F(7,5)) SKIP;
           EXPN=1;    /* SET EXPONENT FOR BELOW   */
/* PRINT 'BELOW' MESSAGE  */
           PUT LIST('APPROACHING FROM BELOW') SKIP(2);
/* PRINT STANDARD HEADING */
START:     PUT EDIT('I','X','|X-A|','F(X)','|F(X)-',L,'|')
                    (X(1),A,X(11),A,X(10),A,X(10),A,X(10),A,F(7,5),A)
                    SKIP(2);
           I=0;    /* INITIALIZE I */
REDO:      I=I+1; /* UPDATE I */
           X=A+(DX**I)*(-1)**EXPN;  /* CALCULATE NEW POINT */
           CALL F(X,FOX);  /* CALCULATE F(X)  */
           AS =ABS(X-A);    /* CALCULATE ACTUAL SEPARATION */
           AE=ABS(FOX-L);  /* CALCULATE ACTUAL ERROR */
/* PRINT RESULTS */
           PUT EDIT(I,X,AS ,FOX,AE) (F(2),X(2),F(13,8),X(1),F(14,8),X(2
                    ,F(14,8),X(2),F(14,8)) SKIP;
           IF I=NOT THEN GO TO NCONV; /* HAVE NOT CONVERGED YET */
           IF AE>E THEN GO TO REDO;  /* TEST EPSILON */
```

```
          IF EXPN=1 THEN DO;   /* ABOVE OR BELOW */
                             EXPN=2;   /* RESET EXPONENT FOR ABOVE */
                             PUT LIST('APPROACHING FROM ABOVE') SKIP(3);
                             DELTA=AS; /* SAVE DELTA FOR BELOW */
                             GO TO START ; /* RETURN FOR ABOVE */
                             END;
          IF AS>DELTA THEN DO;
                               PUT EDIT('DELTA= ',DELTA,' FOR EPSILON=',E)
                                         (A,F(15,10),A,F(15,10)) SKIP(4);
                               GO TO BEG; /* RETURN FOR MORE DATA */
                              END;
          PUT EDIT('DELTA= ',AS,' FOR EPSILON= ',E)
                                         (A,F(15,10),A,F(15,10)) SKIP(4);
          GO TO BEG;
NCONV:    PUT EDIT('DID NOT CONVERGE WITHIN ',NOT,' ITERATIONS')
                    (SKIP,A,F(3),A);
          GO TO BEG;
END EPDL;
*DATA
3  .5  25  2.0000000    .01
3  .5  25  2.0000000    .0001
3  .5  25  2.0000000    .000001
3  .5  25  3.0000000    .01
0 0 0 0 0
//
```

C. The computer output from the above program is shown on the next page:

USING EPSILON-DELTA DEFINITION TO DETERMINE THE
LIMIT AS X APPROACHES 3.00000
FOR THE FUNCTION F(X)=SQRT(X+1)
THE GUESS OF THE LIMIT IS 2.00000 WITH EPSILON=0.01000

APPROACHING FROM BELOW

I	X	|X-A|	F(X)	|F(X)-2.00000|
1	2.50000000	0.50000000	1.87082869	0.12917130
2	2.75000000	0.25000000	1.93649167	0.06350832
3	2.87500000	0.12500000	1.96850196	0.03149803
4	2.93750000	0.06250000	1.98431348	0.01568651
5	2.96875000	0.03125000	1.99217218	0.00782781

APPROACHING FROM ABOVE

I	X	|X-A|	F(X)	|F(X)-2.00000|
1	3.50000000	0.50000000	2.12132034	0.12132034
2	3.25000000	0.25000000	2.06155281	0.06155281
3	3.12500000	0.12500000	2.03100960	0.03100960
4	3.06250000	0.06250000	2.01556443	0.01556443
5	3.03125000	0.03125000	2.00779730	0.00779730

DELTA= 0.0312500000 FOR EPSILON= 0.0100000000

USING EPSILON-DELTA DEFINITION TO DETERMINE THE
LIMIT AS X APPROACHES 3.00000
FOR THE FUNCTION F(X)=SQRT(X+1)
THE GUESS OF THE LIMIT IS 2.00000 WITH EPSILON=0.00010

APPROACHING FROM BELOW

I	X	|X-A|	F(X)	|F(X)-2.00000|
1	2.50000000	0.50000000	1.87082869	0.12917130
2	2.75000000	0.25000000	1.93649167	0.06350832
3	2.87500000	0.12500000	1.96850196	0.03149803
4	2.93750000	0.06250000	1.98431348	0.01568651
5	2.96875000	0.03125000	1.99217218	0.00782781
6	2.98437500	0.01562500	1.99608992	0.00391007
7	2.99218750	0.00781250	1.99804592	0.00195407
8	2.99609375	0.00390625	1.99902319	0.00097680
9	2.99804687	0.00195312	1.99951165	0.00048834
10	2.99902343	0.00097656	1.99975586	0.00024415
11	2.99951171	0.00048828	1.99987792	0.00012207
12	2.99975585	0.00024414	1.99993896	0.00006103

APPROACHING FROM ABOVE

I	X	|X-A|	F(X)	|F(X)-2.00000|
1	3.50000000	0.50000000	2.12132034	0.12132034
2	3.25000000	0.25000000	2.06155281	0.06155281
3	3.12500000	0.12500000	2.03100960	0.03100960
4	3.06250000	0.06250000	2.01556443	0.01556443
5	3.03125000	0.03125000	2.00779730	0.00779730
6	3.01562500	0.01562500	2.00390244	0.00390244
7	3.00781250	0.00781250	2.00195217	0.00195217
8	3.00390625	0.00390625	2.00097632	0.00097632
9	3.00195312	0.00195312	2.00048822	0.00048822
10	3.00097656	0.00097656	2.00024412	0.00024412
11	3.00048828	0.00048828	2.00012206	0.00012206
12	3.00024414	0.00024414	2.00006103	0.00006103

DELTA= 0.0002441406 FOR EPSILON= 0.0001000000

USING EPSILON-DELTA DEFINITION TO DETERMINE THE
LIMIT AS X APPROACHES 3.00000
FOR THE FUNCTION F(X)=SQRT(X+1)
THE GUESS OF THE LIMIT IS 2.00000 WITH EPSILON=0.000001

APPROACHING FROM BELOW

I	X	|X-A|	F(X)	|F(X)-2.00000|
1	2.50000000	0.50000000	1.87082869	0.12917130
2	2.75000000	0.25000000	1.93649167	0.06350832
3	2.87500000	0.12500000	1.96850196	0.03149803
4	2.93750000	0.06250000	1.98431348	0.01568651
5	2.96875000	0.03125000	1.99217218	0.00782781
6	2.98437500	0.01562500	1.99608992	0.00391007
7	2.99218750	0.00781250	1.99804592	0.00195407
8	2.99609375	0.00390625	1.99902319	0.00097680
9	2.99804687	0.00195312	1.99951165	0.00048834
10	2.99902343	0.00097656	1.99975584	0.00024415
11	2.99951171	0.00048828	1.99987792	0.00012207
12	2.99975585	0.00024414	1.99993896	0.00006103
13	2.99987792	0.00012207	1.99996948	0.00003051
14	2.99993896	0.00006103	1.99998474	0.00001525
15	2.99996948	0.00003051	1.99999237	0.00000762
16	2.99998474	0.00001525	1.99999618	0.00000381
17	2.99999237	0.00000762	1.99999809	0.00000190
18	2.99999618	0.00000381	1.99999904	0.00000095

APPROACHING FROM ABOVE

I	X	|X-A|	F(X)	|F(X)-2.00000|
1	3.50000000	0.50000000	2.12132034	0.12132034
2	3.25000000	0.25000000	2.06155281	0.06155281
3	3.12500000	0.12500000	2.03100960	0.03100960
4	3.06250000	0.06250000	2.01556443	0.01556443
5	3.03125000	0.03125000	2.00779730	0.00779730
6	3.01562500	0.01562500	2.00390244	0.00390244
7	3.00781250	0.00781250	2.00195217	0.00195217
8	3.00390625	0.00390625	2.00097632	0.00097632
9	3.00195312	0.00195312	2.00048822	0.00048822
10	3.00097656	0.00097656	2.00024412	0.00024412
11	3.00048828	0.00048828	2.00012206	0.00012206
12	3.00024414	0.00024414	2.00006103	0.00006103
13	3.00012207	0.00012207	2.00003051	0.00003051
14	3.00006103	0.00006103	2.00001525	0.00001525
15	3.00003051	0.00003051	2.00000762	0.00000762
16	3.00001525	0.00001525	2.00000381	0.00000381
17	3.00000762	0.00000762	2.00000190	0.00000190
18	3.00000381	0.00000381	2.00000095	0.00000095

DELTA= 0.0000038146 FOR EPSILON= 0.0000010000

USING EPSILON-DELTA DEFINITION TO DETERMINE THE
LIMIT AS X APPROACHES 3.00000
FOR THE FUNCTION F(X)=SQRT(X+1)
THE GUESS OF THE LIMIT IS 3.00000 WITH EPSILON=0.01000

APPROACHING FROM BELOW

I	X	|X-A|	F(X)	|F(X)-3.00000|
1	2.50000000	0.50000000	1.87082869	1.12917130
2	2.75000000	0.25000000	1.93649167	1.06350832
3	2.87500000	0.12500000	1.96850196	1.03149803
4	2.93750000	0.06250000	1.98431348	1.01568651
5	2.96875000	0.03125000	1.99217218	1.00782781
6	2.98437500	0.01562500	1.99608992	1.00391007
7	2.99218750	0.00781250	1.99804592	1.00195407
8	2.99609375	0.00390625	1.99902319	1.00097680
9	2.99804687	0.00195312	1.99951165	1.00048834
10	2.99902343	0.00097656	1.99975584	1.00024415
11	2.99951171	0.00048828	1.99987792	1.00012207
12	2.99975585	0.00024414	1.99993896	1.00006103
13	2.99987792	0.00012207	1.99996948	1.00003051
14	2.99993896	0.00006103	1.99998474	1.00001525
15	2.99996948	0.00003051	1.99999237	1.00000762
16	2.99998474	0.00001525	1.99999618	1.00000381
17	2.99999237	0.00000762	1.99999809	1.00000190
18	2.99999618	0.00000381	1.99999904	1.00000095
19	2.99999809	0.00000190	1.99999952	1.00000047
20	2.99999904	0.00000095	1.99999976	1.00000023
21	2.99999952	0.00000047	1.99999988	1.00000011
22	2.99999976	0.00000023	1.99999994	1.00000005
23	2.99999988	0.00000011	1.99999997	1.00000002
24	2.99999994	0.00000005	1.99999998	1.00000001
25	2.99999997	0.00000002	1.99999999	1.00000000

DID NOT CONVERGE WITHIN 25 ITERATIONS

D. Below is a partial explanation for EXAMPLE 2.

Examine the output. For the case when E = .01, the actual error is smaller than E after only 5 computations from both above and below, hence a D = .03125 is determined. Eighteen computations were required to find a D = .0000038146 when E = .000001. Observe in the last case a different limit of L = 3 was used. Hence, no matter how small a D was taken, the actual error could never be smaller than 1.0, so a value of L = 3 was obviously incorrect. In fact, any value for L other than 2 should provide non-convergence.

5. STUDENT EXERCISES:

For the following functions, write a program which will determine a correct D when given several values for E.

a.	$f(x) = \frac{\tan x}{x}$	a=0	L=1
b.			
b.	$f(x) = 100x^3$	a=1	L=100
c.	$f(x) = \frac{x^2-7x+12}{x-3}$	a=3	L=-1
d.	$f(x) = \sqrt{x}$	a=4 a=2	L=2 L=1.4
e.	$f(x) = 1/x^2$	a=0 a=1/2	L=1,000,000 L=3.9
f.	$f(x) = (1+x)^{1/x}$	a=0	L=2.718281828
g.	$f(x) = \frac{\sin x + \tan x}{x}$	a=0	L=2
h.	$f(x) = 3x^2 + 2x^2 + x + 1$	a=1 a=1	L=7 L=6.9

REFERENCES

1. American Association for the Advancement of Science. Guidelines and Standards for the Education of Secondary School Teachers of Science and Mathematics. Washington, D. C.: AAAS Miscellaneous Publication 71-9, (undated).

2. Andrews, Dr. George H. Information in a letter to Dr. William P. Love of UNC-G from Dr. George H. Andrews of Oberlin College, (February, 1972).

3. Bitter, Gary. "Computer-Oriented Calculus," Journal of Educational Data Processing, VII, 3 (1970), 193-5.

4. Carpenter, Walter J. "The Elementary Calculus Course at North Georgia College," (undated), (Mimeographed).

5. Center for Research in College Instruction of Science and Mathematics. Computer - Calculus. Tallahassee, Florida: Center for Research in College Instruction of Science and Mathematics, Newsletter Number 2, (February, 1971).

6. Committee on the Undergraduate Program in Mathematics. A General Curriculum in Mathematics for Colleges. Berkley, California: CUPM, (January, 1965).

7. Committee on the Undergraduate Program in Mathematics. Calculus with Computers. Berkley, California: CUPM Newsletter Number 4, (August, 1969).

8. Committee on the Undergraduate Program in Mathematics. Recommendations for an Undergraduate Program in Computational Mathematics. Berkley, California: CUPM, (May, 1971).

9. Committee on the Undergraduate Program in Mathematics. Recommendations on Course Content for the Training of Teachers of Mathematics. Berkley, California: CUPM, (August, 1971).

10. Davis, Philip J. "Computer Calculus." Brown University, (December, 1967), (Mimeographed).

11. Davis, Philip J. and Charles Strauss. "Computer Calculus - First Year Report." Brown University, (May, 1969), (Mimeographed).

12. Diaz, Dr. Octavio. Information in a letter to author from Dr. Octavio Diaz of North Carolina A&T State University, (February, 1972).

13. Duren, W. L. "A General Curriculum in Mathematics for Colleges," The American Mathematical Monthly, LXXII (October, 1965), 825-830.

14. Fuller, Dr. William R. Information in a letter to author from Dr. William R. Fuller of Purdue University, (March, 1972).

15. Garfunkel, Soloman. "A Laboratory and Computer Based Approach to Calculus," The American Mathematical Monthly, LXXIX (March, 1972) 282-90.

16. Goldstine, H. H. and Bernard Friedman. "What can be done about Teaching the Applications for Mathematics in Colleges and Universities," The American Mathematical Monthly, LXXII (February, 1965), 113-7.

17. Hethcote, H. W. and A. J. Schaeffer. "A Computer Laboratory Course for Calculus and Linear Algebra," The American Mathematical Monthly, LXXIX (March, 1972), 290-3.

18. Iltis, Dr. Richard. Information in a letter to author from Dr. Richard Iltis of UNC-Chapel Hill, (May, 1972).

19. Miles, E. P. " 'Impact of Computer Applications on Undergraduate Mathematics Instruction' Explored at Florida State - ACM - SIAM Symposuim," Communications of the Association of Computing Machinery, (May, 1966), 388-9.

20. Miles, E. P. "Calculus and the Computer, The CRICISAM Calculus Project - Past, Present and Portent." The American Mathematical Monthly, LXXVII (March, 1971), 284-91.

21. National Science Foundation, Computers in Undergraduate Education: Mathematics, Physics, Statistics, and Chemistry. Edited by Dr. Thomas M. Gallie. College Park, Maryland: Science Teaching Center, University of Maryland, (December, 1967).

22. Porter, E. J. "A Computer Assisted Approach to Integral Calculus via Jordan Content," The American Mathematical Monthly, LXXVII (November, 1970), 999-1001.

23. Smith, Dr. David A. "Calculus - with - Computer Experiment," Educational Studies in Mathematics, 3 (1970), 1-11.

24. Smith, Dr. David A. Information in a letter to Dr. William P. Love of UNC-G from Dr. David A. Smith of Duke University, (January, 1972).

25. Weinstein, Dr. Stanley E. Information in a letter to Dr. William P. Love of UNC-G from Dr. Stanley E. Weinstein of the University of Utah, (March, 1972).

26. Wicht, M. C. Information in a letter to Dr. William P. Love of UNC-G from M. C. Wicht, Chairman of Department of Mathematics, of North Georgia College, (April, 1972).

27. Williams, Dr. Horace E. Information in a letter to Dr. William P. Love of UNC-G from Dr. Horace E. Williams of Vanderbilt University, (February, 1972).