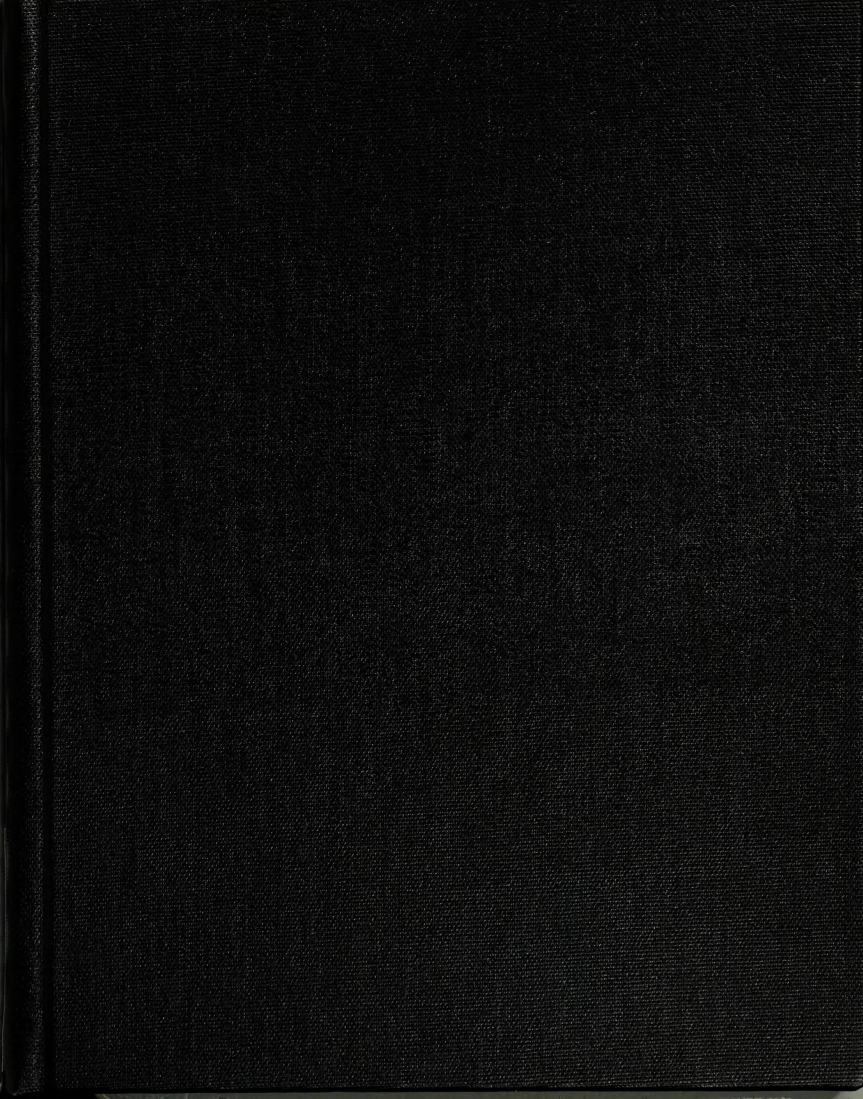






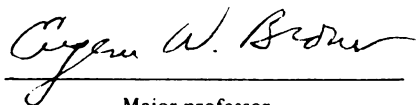
This is to certify that the
thesis entitled
Development of a Data Smoothing Software Package
with Application to Smoothing Technique Comparison

presented by

Daniel J. Wilson

has been accepted towards fulfillment
of the requirements for

M.A. degree in Physical Educ.


Major professor

Date November 10, 1969

0-7639

MSU is an Affirmative Action/Equal Opportunity Institution



PLACE IN RETURN BOX to remove this checkout from your record.
TO AVOID FINES return on or before date due.

DATE DUE	DATE DUE	DATE DUE
_____	_____	_____
_____	_____	_____
_____	_____	_____
_____	_____	_____
_____	_____	_____
_____	_____	_____
_____	_____	_____

MSU Is An Affirmative Action/Equal Opportunity Institution

c:\circ\data\due.pm3-p.1

DEVELOPMENT OF A DATA SMOOTHING COMPUTER SOFTWARE PACKAGE
WITH APPLICATION TO SMOOTHING TECHNIQUE COMPARISON

By

Daniel James Wilson

A THESIS

Submitted to

Michigan State University

in partial fulfillment of the requirements

for the degree of

MASTER OF ARTS

School of Health Education,

Counseling Psychology and Human Performance

1989



6050505

ABSTRACT

DEVELOPMENT OF A DATA SMOOTHING COMPUTER SOFTWARE PACKAGE WITH APPLICATION TO SMOOTHING TECHNIQUE COMPARISON

By

Daniel James Wilson

The purpose of this investigation was to create a flexible computer software package to assist the researcher of human motion in selecting and implementing appropriate data smoothing techniques. This was accomplished by including a tutorial in the package to instruct the researcher about the proper smoothing routine for each major classification of human motion.

Validation of the smoothing routines was accomplished by calculating the mean square error associated with differences in data values between raw and smooth numeric data. All smoothing routines written for this investigation were found to be acceptable at a pre-determined level of ninety-five percent mean square agreement.

The smoothing routines were compared and quantitatively ranked based upon their mean square error. The routines which consistently produced the best results, in descending order were: (1) natural cubic spline function, (2) Chebyshev polynomial, (3) Butterworth filter, (4) least squares polynomial, and (5) fourier series. It was noted that the natural cubic spline function produced no error.

DEDICATION

TO MY PARENTS

ACKNOWLEDGMENTS

While attempting to write this acknowledgment, I find the task of identifying all those individuals who have offered their support overwhelming. So, for those not specifically mentioned here, I offer my deepest appreciation if they should find themselves reading this manuscript.

The following individuals must be acknowledged both for their invaluable assistance in the completion of this project, and for demonstrating that friendship and caring are always the best teachers.

I must begin by expressing my sincere appreciation to the man who has guided my graduate career, my committee chair, advisor, and friend, Dr. Eugene W. Brown. As always, I must thank my long time mentor and friend, Dr. Douglas E. Hanson, who continually provides the role model which guides so many toward achieving their goals. I offer a special thank-you to my committee member and friend, Dr. Tyler Haynes, who shows that a love of teaching comes from a love of students. Also, I feel I must express my gratitude to Dr. Vern Seefeldt, Dr. John Haubenstricker and Dr. Rebecca Henry for providing me with the research assistantships which allowed me to pursue my graduate education. Finally, thank-you to Mr. Kaveh Abani for his assistance in the interpretation of the mathematical theory which builds the foundation of this thesis.

TABLE OF CONTENTS

Chapter	Page
LIST OF TABLES.....	viii
LIST OF FIGURES.....	ix
I. INTRODUCTION.....	1
Statement of the Problem	2
Purpose of the Study	3
Need for the Study.....	4
Delimitations of the Study	5
Limitations of the Study	5
Definitions of Terms.....	6
II. REVIEW OF RELATED LITERATURE.....	8
An Assessment of Selected Data Smoothing Techniques.....	8
Least squares approximation	8
Cubic spline functions	9
Fourier series	10
Chebyshev polynomials	10
Butterworth filter	11
Classification System for Motion Studies.....	12
The Past and Present of Computer Software and	
Future Considerations for Development	15
III. DEVELOPMENTAL METHODOLOGY AND EXPERIMENTAL PROCEDURES.....	17
Criterion for Inclusion.....	17
Structure of the Program	19
Developmental Procedures	20
Least squares approximation	20
Spline functions	22

	Chebyshev polynomials	23
	Fourier series	23
	Butterworth filter	24
	Validation of the Program.....	24
	Comparison of Techniques	25
	Use of the Computer Program	26
	Summary.....	26
IV.	RESULTS AND DISCUSSION.....	27
	Data Smoothing Information Questionnaire.....	27
	Comparison of Smoothing Methods.....	30
	Validation.....	30
	Comparison of techniques	31
	Using the software package	35
	Summary	35
V.	CONCLUSIONS	40
	Summary.....	40
	Findings	40
	Implementation	41
	Recommendations for Further Research and Development...	42
	REFERENCES.....	43
	GENERAL REFERENCES.....	45
	APPENDIX A Data Smoothing Questionnaire	47
	APPENDIX B Numeric Data	50
	APPENDIX C Data Smoothing Software Package	56

LIST OF TABLES

Table		Page
1	Literature Review Frequencies by Motion Classification	14
2	Summary of Data Smoothing Information Questionnaire	28
3	Mean Square Difference for Exact Numeric Data	32
4	Data Smoothing Routine Ranking by Motion Classification in Descending Order	33

LIST OF FIGURES

Figure		Page
1	The Trigonometric Nature of Repeating or Oscillatory Motion	12
2	Digital Filter Black-Box Design	13
3	Block Diagram of Motion Classifications	15
4	Flow Chart of Developmental Procedures in Creating a Software Package	21
5	Main Menu of Data Smoothing Software Package	36
6	Tutorial Menu Screen	36
7	Tutorial Screen from Least Squares Approximation	37
8	Main Screen for Inputting Data from Disk	37
9	Entering Drive and Filename for Data Retrieval from Disk	38
10	Main Menu for Data Smoothing Techniques	38
11	Smoothed Data Curves from Various Data Smoothing Techniques	39

CHAPTER I

INTRODUCTION

The biomechanical analysis of human motion is a complicated and time consuming task. Our predecessors, utilizing qualitative observation techniques, realized the complexity of analyzing human motion and sought to develop quantitative techniques (Bates, 1973; Hoffman and Warsham, 1971; and Ward, 1971). These quantitative methods were developed to help solve the problem of reducing human movement to a speed which could be effectively analyzed. Human movement often occurs far too rapidly to be observed and accurately analyzed with the naked eye. This is especially true in sports skills in which the goal of performance is to produce maximal velocity, acceleration, and/or force.

With the advent of the high speed motion picture camera, technology gave the researcher a new tool to assist in capturing movement. The development of this electronic "eye" to assist researchers derives its origin from the early sequential still photographs taken by Muybridge in order to make general statements as to the nature of human and animal motion. The plates presented in his book, Animals in Motion (c1899), include brief descriptions of body segment reactions to various every-day tasks. From this beginning, the high speed motion picture camera has evolved into an invaluable tool for the collection of film data used in biomechanical research.

With the development of the high speed motion picture camera, a powerful new device for the investigator of human motion, also came new problems in the analysis and interpretation of data. Two major complications faced the investigator.

The first problem came in the form of overwhelming amounts of data which had to be organized and interpreted. Recorded images collected by means of high speed motion picture photography needed to be converted into cartesian coordinate values. The advent of electronic digitizers, instruments which could electronically convert points located on projected film images into coordinate data, effectively bridged the gap between the acquisition of data and the subsequent transformation of that data to a usable electronic form. From film taken at one hundred frames per second, a typical rate used by researchers who analyze sports skills and physical activities, three-thousand or more body segment coordinate pairs could be generated in the analysis of just ten seconds of filming. The investigation of movement through high speed motion picture techniques was made practical by the invention of the digital computer. The ability of the digital computer to store, retrieve, display, and analyze vast amounts of data made it an invaluable tool in biomechanical research. The impact of the computer on data processing in general is best stated by Blakesley:



CHAPTER I

INTRODUCTION

The biomechanical analysis of human motion is a complicated and time consuming task. Our predecessors, utilizing qualitative observation techniques, realized the complexity of analyzing human motion and sought to develop quantitative techniques (Bates, 1973; Hoffman and Warshaw, 1971; and Ward, 1971). These quantitative methods were developed to help solve the problem of reducing human movement to a speed which could be effectively analyzed. Human movement often occurs far too rapidly to be observed and accurately analyzed with the naked eye. This is especially true in sports skills in which the goal of performance is to produce maximal velocity, acceleration, and/or force.

With the advent of the high speed motion picture camera, technology gave the researcher a new tool to assist in capturing movement. The development of this electronic "eye" to assist researchers derives its origin from the early sequential still photographs taken by Muybridge in order to make general statements as to the nature of human and animal motion. The plates presented in his book, Animals in Motion (c1899), include brief descriptions of body segment reactions to various every-day tasks. From this beginning, the high speed motion picture camera has evolved into an invaluable tool for the collection of film data used in biomechanical research.

With the development of the high speed motion picture camera, a powerful new device for the investigator of human motion, also came new problems in the analysis and interpretation of data. Two major complications faced the investigator.

The first problem came in the form of overwhelming amounts of data which had to be organized and interpreted. Recorded images collected by means of high speed motion picture photography needed to be converted into cartesian coordinate values. The advent of electronic digitizers, instruments which could electronically convert points located on projected film images into coordinate data, effectively bridged the gap between the acquisition of data and the subsequent transformation of that data to a usable electronic form. From film taken at one hundred frames per second, a typical rate used by researchers who analyze sports skills and physical activities, three-thousand or more body segment coordinate pairs could be generated in the analysis of just ten seconds of filming. The investigation of movement through high speed motion picture techniques was made practical by the invention of the digital computer. The ability of the digital computer to store, retrieve, display, and analyze vast amounts of data made it an invaluable tool in biomechanical research. The impact of the computer on data processing in general is best stated by Blakesley:



"In the broadest and simplest concepts, computers have their greatest value in extending man's mental capacities in two interrelated areas. Computers are useful where: (1) significant time-consuming, repetitious, and voluminous operations of data and facts exist; and (2) where not so simple, but significantly more complex calculations and logical determinations are to be made." (1967, p. 185).

The second point made by Blakesley applies to the analysis of high speed motion picture data. Kinematic and kinetic calculations of body segment parameters are complex. Minimizing the possibility of human error by using the capabilities of the digital computer to perform mathematical calculations further decreases the burden placed on the researcher.

The second problem arising from cinematographical analysis of human movement is related to the transfer of information into usable form. In order to obtain usable data from film, the images must be transformed into a series of coordinates by use of an electronic digitizer. This process introduces a "noise" component into the numerical data values obtained. Noise can be defined as: "Error present in data collected that is unrelated to the process being studied" (Rodgers and Cavanagh, 1984). Error is introduced into the data by a variety of sources. These include the inherent vibration of high speed motion picture cameras and noise produced through human error (e.g., subjective selection of reference points). In the process of electronically digitizing points on a projected image of the body, a certain amount of subjective judgement is needed in order to locate body landmarks. Consistently choosing the exact location of a selected body landmark is a demanding task for even the most experienced and persistent investigator. The noise created by these potential sources of errors, while perhaps not evident within the raw positional data, may become magnified when they are applied to high-order derivatives, characteristic of certain data smoothing techniques. If researchers are to extract meaningful kinematic and kinetic analyses from cinematographic data, appropriate data smoothing methods which minimize inherent errors associated with derived values is critical.

Statement of the Problem

The problem of this investigation was to create and validate a versatile general purpose computer program for data smoothing that could also be used to compare various smoothing techniques. The program was designed to assist researchers in minimizing the influence of "noise" associated with the cinematographical analysis of human movement by guiding them in analyzing the output from various data smoothing techniques.

The process of carrying out this study was divided into two parts: (1) the design and development of a computer program that could be used to smooth numeric data, and (2) the validation of the program through comparisons of various smoothing techniques applied to biomechanical data.

The first part of the plan was carried out through the development of a general purpose computer software package on an IBM AT personal computer in the BASIC programming language. BASIC was selected because of the universal familiarity among researchers and ease of learning for those not trained in computer programming. The package breaks down the smoothing process into component parts. The parts of the package include: (1) a tutorial to assist the researcher in selecting a proper data smoothing technique; (2) an input/output routine allowing for varying internal/external storage formats to assure maximal flexibility in data entry; (3) a menu for selection of different smoothing techniques and; (4) kinematic descriptions of data including graphic displays of position, velocity, and acceleration. The package is designed in this fashion because of the critical need for selecting the appropriate smoothing technique. Flexibility of input and output of data are maintained in view of this program being part of a larger proposed software package for the complete kinematic and kinetic analysis of biomechanical information. The input/output routines link these individual packages together and varying internal storage formats may occur through different programmers. It is a function of this package to allow for versatility of format.

The second part of this plan was validation. Validation of the data smoothing routines was accomplished by comparison of the mean percent error between the generated numeric data values and the resultant smoothed data. Zernicke, Caldwell, and Roberts (1977) utilized a similar method in determining the mean percent error when comparing force platform data of typical human gait. In their study, a 95.25 percent mean agreement was reached based upon these comparisons. For the purposes of the current investigation, a ninety-five percent mean agreement has been established to determine acceptability of the smoothing routines.

Purpose of the Study

The purpose of this study was to create a flexible computer software package to assist the researcher of human motion in selecting and implementing appropriate data smoothing techniques. This was accomplished by including a tutorial in the package to instruct the researcher about the proper smoothing method to select for each classification of motion. All routines in the package are menu-driven to guide the investigator in the use of each program. This will aid researchers, coaches, and

educators in making more precise kinematic descriptions of movement from cartesian coordinate data generated from high speed film. The program will serve as the second phase of a three part process (data acquisition, data smoothing, and data analysis) in biomechanical investigation utilizing high speed motion picture photography.

Need for the Study

The smoothing of biomechanical data, as a research topic, takes its origins from the mathematical subject of numerical analysis. Since biomechanics is a relatively new field of study, the amount of investigation specifically related to data smoothing is limited to relatively few studies. Among the researchers who have attempted to develop computer programs for data smoothing, the approaches have been diverse, creating problems in generalizing results from one program to another. This lack of continuity between software packages comes in many forms. The input/output format is the first problem that arises in these programs. Since each investigator has his/her own style of computer programming, the corresponding methods of input and output are unique to that author. This can create problems in comparisons of output generated by different programs. With a more universal approach to the design of these procedures, the opportunity for comparison of analyses of various computer programs is created. The actual kinematic and kinetic output from these programs is a second problem. Each researcher develops specific routines needed in the interpretation of his/her own data. Therefore, the ability to compare and contrast results may not be available since these calculations may not have been performed. The problem being addressed most prominently within this investigation is the selection of the proper smoothing technique for the data being analyzed. This step is often taken for granted because researchers develop smoothing routines specific to a type of data being collected. If researchers are to make the information gained through biomechanical investigation available to coaches and educators in the field, they must make the tools suitable for their use. A selection of smoothing routines, together with a tutorial to aid users in their selection, is a logical solution to this problem.

The need for the study, then, becomes the design and development of a computer program for data smoothing which can be used by researchers, coaches, and educators. It is to be designed to allow the user to analyze the results of the cinematographical data. Comparisons become critical in demonstrating the selection of appropriate techniques.

Delimitations of the Study

The following delimitations refer to the development of the computer program:

1. The program is written in the BASIC computer language specifically for an IBM or IBM compatible microcomputer.
2. Kinematic calculations performed within the program are for motion in one plane at a time.
3. Availability of memory size specific to a microcomputer restrict the number of data points allowable.
4. Graphic comparisons of only two curves at one time are allowable, due to memory space availability on some microcomputers.
5. Classifications of motion given in this study are not those typically found in physical activity, but are laboratory conditions developed to allow for testing. The data generated for each classification contains no error, thus facilitating direct comparison.

Limitations of the Study

Human motion occurs in a three dimensional space. Using cinematographical techniques, it is possible to study motion either from a two or three dimensional perspective. Because of the nature of photographic techniques, three dimensional analysis of movement is a complicated process. However, through the use of proper analytical tools, three dimensional investigation should be utilized when appropriate. The problem which arises in high speed cinematography is that a single camera can only record within a plane of motion (x,y coordinate system). It is through the use of multiple cameras that a three dimensional view of the action being investigated can be accomplished (x, y, z coordinate system). The single plane of action captured by each individual camera in a study employing multiple cameras can be converted into three dimensional coordinates by the use of a direct linear transformation routine (Shapiro, 1978). Although this process will not be used in this study, a detailed discussion of this procedure can be found in Walton (1981).

Definition of Terms

Terms defined in this section are primarily based on those given by Rodgers and Cavanagh (1984.)

Acceleration. The rate of change of velocity with respect to time, mathematically, the second time derivative of displacement and the first time derivative of velocity. This term applies to both linear (straight line), and angular (curved path) motions.

BASIC. Beginner's All-Purpose Symbolic Instructional Code.

Derivatives. Quantities obtained by the process of differentiating a given curve or function. The most commonly used derivatives are of displacement (x) where first derivatives dx/dt = velocity, and second derivative d^2x/dt^2 = acceleration.

Differentiation. A technique of calculus for finding the rate of change of a quantity.

Digitizer. A device that is capable of acquiring planar coordinates in numerical form. In biomechanics, the most frequent use of a digitizer is to convert the location of body markers on the projected image of a film into numbers that can be processed by a computer.

Displacement. The change in the position (linear or angular) of a body.

Force. A vector quantity that describes the action of one body on another.

Integration. A technique of calculus for determining the area enclosed between a curve and the x -axis. This can result in either a single value or a new curve derived from the varying function. Integration can be performed either graphically, mathematically, or numerically. It is the inverse process of differentiation.

Kinematics. The description of motion.

Kinetics. The study of the forces that cause motion.

Noise. Error present in data collected that is unrelated to the process being studied. Some noise is almost always present in data collected in biomechanics and most other fields (e.g., vibration).

Smoothing. A class of techniques for reducing the effects of noise.

Software. Sets of instructions called programs that cause a computer to execute certain operations.

Vector. A quantity that has both direction and magnitude.

Velocity. A measure of a body's motion (linear or angular) in a given direction.

Because velocity has both magnitude and direction, it is a vector quantity that can be positive, negative, or zero.

CHAPTER II

REVIEW OF RELATED LITERATURE

This chapter has been structured to present a detailed review of smoothing methods being applied in biomechanics research that utilizes high speed cinematography. A three part approach has been selected to give a logical sequence to the presentation of this information. The first section will give an assessment of each technique to be incorporated in the computer software package, including a brief description of the method and the main considerations for its use. Part two includes a flow diagram of motion classifications and a frequency table based on a review of research articles in biomechanics which incorporate the use of specific data smoothing routines. The third part is a discussion of the past and present state of data smoothing software and future considerations for computer application development.

An Assessment of Selected Data Smoothing Techniques

Several investigations have addressed problems associated with data smoothing. Generally, these studies have focused on a particular smoothing method. The development of a general purpose computer software package incorporates several of these routines. Therefore, each data smoothing technique selected will be reviewed. A brief description of each method will be given, followed by a discussion of its relative merits and considerations for utilizing the procedure.

Least-squares approximation

The least-squares approach to curve fitting involves developing m equations (the normal equations) which minimize the sum, S , of the squares of the deviations of the equation:

$$S = \sum [y_i - p(t_i)]^2, \quad (2.1)$$

where $\sum$ is the summation of i from 1 to n (n = number of coordinate data pairs). The m equations can be solved simultaneously to yield the coefficients for a least-squared fit of a polynomial of degree m to a set of empirical data (Wood, 1982). The popularity of local least-squares approximation techniques has resulted from their simplicity of

application, and the convenience of this procedure in determining linear approximations (Burden and Faires, 1985). With this technique, smoothed data values and corresponding derivatives can be directly obtained from experimental data. Also, if the nature of the data is known or can be assumed to have its mean distributed in a linear manner, the values obtained from a linear least-squares routine are unbiased estimates for the equation that describes the line of best fit through the mean (Dodes, 1978). Decisions made in applying the least-squares approximation technique includes: (1) the degree (m) of the approximating polynomial, (2) the number of data points (n) to include in the smoothing and differentiation formula, and (3) the number of repeat applications of the smoothing algorithm prior to the use of the differentiation formula (Wood, 1982). The degree of the approximating polynomial (m) should be less than four according to Hershey et al. (1971). However, larger degree polynomials may be necessary when greater error is present within the data. Selection of the number of data points (n) and repetitions are at the discretion of the investigator. It should be kept in mind, however, that over smoothing is possible with a large n (number of data points) or number of repetitions.

Cubic spline functions

Mathematically, a spline function is a group of cubic polynomials arranged so that adjacent curves join each other with continuous first and second derivatives (Dodes, 1978). These polynomials are "pieced" together at points in time (t) called "knots" (x_j ; $j = 1, 2, \dots, n$). The resulting cubic spline function consists of $n - 1$ cubic polynomials of the form:

$$g(t) = p_j(t) = a_j + b_j(t) + c_j(t)^2 + d_j(t)^3, \quad (2.2)$$

on the interval $x_{j-1} \leq t < x_j$. This "piecewise" nature of the function allows it to adapt quickly to changes in curvature.

A "spline" is actually a pliable strip of wood or rubber which was used by draftsmen in patterning curves. The mathematical form was made popular as an approximation procedure during the 1960's, when it was shown that a spline function was the smoothest of all functions for fitting (n) data within specified limits (Greville, 1969).

One unique characteristic of the cubic spline is that it involves four constants; (a , b , c , and d), thus insuring that there are sufficient continuously differentiable first and second derivatives (Burden and Faires, 1985). If all constants within equation (2.2) were zero, taking the second derivative would eliminate c and d . This means that velocity and acceleration values can be obtained, because they are the first and second

derivatives of the fitted equation respectively. Other data smoothing techniques may not have four constants. Therefore, if velocity and acceleration values are critical, the cubic spline offers assurance of the needed orders for analysis.

Fourier series

For a series of (n) data points with equally spaced time values, there exist (n) orthogonal functions over these data points which are either a sine or cosine of a multiple of t (Pizer, 1975). These functions make up what is known as the fourier series. The general form of the equation is:

$$a_j + \sum [a_j \sin(j 2\pi t/T) + b_j \cos(j 2\pi t/T)], \quad (2.3)$$

where $\sum$ is the summation of j from 1 to n. The first sine and cosine terms from equation (2.3) represent functions describing one cycle (2π radians) in time (T seconds), and subsequent terms represent functions whose frequency are a multiple of this base frequency. Functions are referred to as "orthogonal" if their first definite integral on the range (a,b) is equal to zero (Anton, 1980).

One of the purposes of using a data smoothing method is to "fit" an equation to data obtained from the digitizing process. If the result should take the form of various ordinary and partial differential equations, the fourier series is an extremely powerful tool in describing the solution (Burden and Faires, 1985). This is true because of the trigonometric nature of a fourier series which fits the approximating polynomial closely. Since partial differential equations require considerable skill to compute (Anton, 1980), the fourier series offers a time-saving method which eliminates costly mistakes by the researcher.

Chebyshev polynomials.

By definition, the set of Chebyshev polynomials $[T_n]$ are:

$$T_n(x) = \cos[n \arccos x], \quad (2.4)$$

for x, an element of $[-1,1]$, and $n = 0,1,2,\dots$ (Burden and Faires, 1985). Introducing the substitution $\theta = \arccos x$, it can be shown that the following Chebyshev polynomials are derived

$$\begin{aligned}T_2(x) &= 2x T_1(x) - T_0(x) = 2x^2 - 1, \\T_3(x) &= 2x T_2(x) - T_1(x) = 4x^3 - 3x, \\T_4(x) &= 2x T_3(x) - T_2(x) = 8x^4 - 8x^2 + 1,\end{aligned}$$

and so on. With the aid of the equation $T_n(x) = \cos(n \theta)$ from the previous substitution, these equations reduce to

$$T_{n+1}(x) = 2x T_n(x) - T_{n-1}(x), \quad (2.5)$$

for each $n = 1, 2, \dots$ and x an element of $[-1, 1]$. This form of the Chebyshev polynomials is used to approximate a "line of best fit".

The Chebyshev method utilizes the trigonometric cosine to follow harmonic or repeating forms of motion (Dodes, 1978). Harmonic motions repeat in a wave-like pattern which closely approximate the graph of the trigonometric sine and cosine (See Figure 1). This type of movement is common in biomechanical research with such motions as running or cycling.

The main asset in using the Chebyshev polynomial is in its ability to remain "stable" for higher-order equations (Dodes, 1978). A stable condition, simply stated, means that a small error made at any step in a calculation does not cause a geometrical summation of the error as the procedure is repeated (Pizer, 1983). Since harmonic motions are cyclic in nature, these motions will be smoothed repetitively introducing the opportunity for small errors to be compounded. The Chebyshev technique allows the researcher to be confident of minimal errors when working with vast amounts of data.

Butterworth filter.

The Butterworth filter is an electronic process to remove "noise" from data. The basic design of a digital filter is quite simple. A digital filter by design is: (1) some form of electronic input, (2) a mathematical process to remove noise, and (3) the resulting output (See Figure 2). The digital filter which will be used in this investigation will be a low-pass Butterworth filter, so called because the operation it performs involves passing only the low-frequency components of a signal. A low-pass filter has been selected based upon the results of previous investigations (Pezzack, Norman and Winter, 1977; and Winter 1979) which concluded that application of a low-pass digital filter followed by a first order finite difference routine produced the best approximation of accelerometer data taken from various classifications of motion.

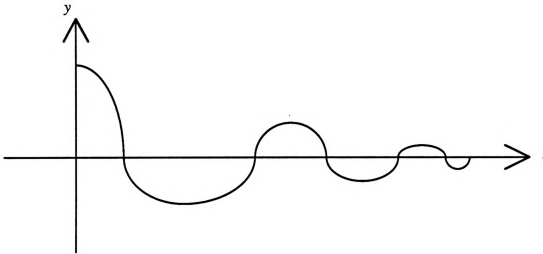


Figure 1. The trigonometric nature of repeating or oscillatory motion.

A digital filter is an effective tool in data smoothing because of its ability to represent oscillating or periodic types of motion effectively (Williams, 1986). Oscillatory motion repeats itself along the same path. This type of harmonic movement, while present in biomechanical research, is not reported as frequently as periodic motion which does not repeat the same path exactly. The main use of the Butterworth digital filter is for periodic types of movement. Since repetitive motion is so prevalent in human locomotion, the Butterworth filter is one of the most frequently utilized data smoothing methods (See Table 1).

Classification System for Motion Studies

In order to properly select a data smoothing technique, it is essential to develop an understanding of the nature of motion being investigated. One such system is presented in the flow chart of Figure 3.

Human motion can be defined as: "Change of position in space and time through force applications" (Barham, 1978). The three general classifications used in describing



Figure 2. Digital filter black-box design.

human movement are: (1) linear, (2) curvilinear or angular, and (3) harmonic. Linear motions are those actions which occur in a straight line. Curvilinear or angular motions follow a curved or arc-like path. This may include projectile motions or the flight paths of implements. Harmonic movements have a repetitive nature in which the directional path is cyclic.

Further subclassifications of curvilinear motion include circular and parabolic movements. Circular movements remain a constant distance from a fixed point. The constant distance from a fixed point is, of course, the radius of a circle. Motion that follows a path which is always an equal distance from a fixed point and a fixed line is parabolic (Barham, 1978). An example of this motion is the pathway followed by a projectile in flight in the earth's atmosphere when it is moving at an oblique angle to the horizontal and the effect of air resistance is ignored.

Periodic motion is any movement that repeats itself. This is distinguished from oscillatory motion by the fact that the latter repeats itself over the same path. An example of periodic movement is the path taken by the foot when running; the motion repeats in an elliptical form, but never exactly over the same course. If the path taken by a particular point of a pendulum were to be traced, it would exemplify oscillatory motion, where the exact arc is retraced on each successive swing.

Table 1 gives a detailed review of the classification system developed in Figure 3 as applied to human motion studies. The far left column of Table 1 includes the types of motion under investigation. Notably absent from these descriptions are the terms circular, parabolic, periodic, and oscillatory. Each of these subdivisions are considered to be part of the description preceding them in the flowchart shown in Figure 3. The labels "constant" and "varying" have been added because human motion rarely occurs as a constant. In the second column in Table 1, the motion classifications are subdivided into "velocity" and "acceleration". They will be considered separately since studies of

MOTION CLASSIFICATION		DATA SMOOTHING TECHNIQUES				
TYPES	SUBCLASSIFICATION	LEAST SQUARE	SPLINE FUNCTION	FOURIER SERIES	CHEBYSHEV POLY.	BUTTERW. FILTERING
CONSTANT LINEAR	VELOCITY	n=1	n=2		n=3	n=2
	ACCELERATION	n=1	n=2		n=3	n=2
VARYING LINEAR	VELOCITY	n=4	n=7	n=1	n=5	n=4
	ACCELERATION	n=4	n=7		n=5	n=5
CONSTANT NONLINEAR	VELOCITY	n=6	n=2	n=4		n=4
	ACCELERATION					
VARYING NONLINEAR	VELOCITY	n=9	n=11	n=3	n=1	n=16
	ACCELERATION	n=7	n=9	n=5	n=1	n=16
CONSTANT HARMONIC	VELOCITY	n=2	n=2	n=3		n=3
	ACCELERATION					
VARYING HARMONIC	VELOCITY	n=8	n=7			n=6
	ACCELERATION	n=8	n=7			n=5

Table 1. Literature review frequencies by motion classification.

human movement are often concerned with only one of these subclassifications.

The headings for the frequency cells in Table 1 are the data smoothing techniques included in the computer software package developed for this study. Numeric values in the cells of Table 1 represent a frequency of appearance of the smoothing techniques by type and subclassification of motion found in a review of biomechanics studies in the: (1) International Journal of Sport Biomechanics (1985-1988), (2) Journal of Biomechanics (1984-1987), and (3) Research Quarterly for Exercise and Sport (1977-1986).

Referring to Table 1, the frequency cells indicate the number of articles reviewed which specifically mentioned the use of the smoothing technique heading each column. Main considerations brought out by this table include the types of motion most frequently under investigation by each technique and the distribution of frequencies throughout the entire table.

Varying nonlinear or angular motion was the most common classification of movement encountered. Because these studies usually involved some sort of electronic signal processing, the Butterworth filter contained the highest cell numbers.

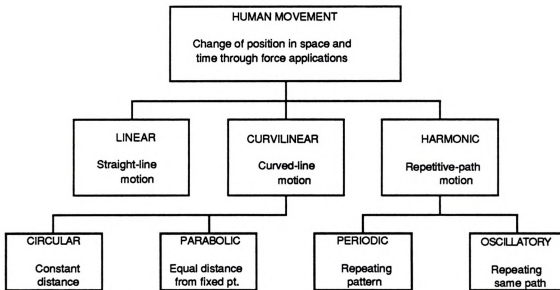


Figure 3. Block diagram of motion classifications.

The relative distribution of cell frequencies is the most important characteristic of Table 1. No specific pattern of smoothing method related to motion classification is evident. Since one of the purposes of this investigation was to develop guidelines as to the utilization of data smoothing techniques, the fact that routines seem to be selected at random supports the need for this study.

The Past and Present of Computer Software and Future Considerations for Development

In an overview of software available, Bates (1973) discussed the characteristics of the programs being used. He was trying to solve the problem of software flexibility by developing a program which could be used by anyone interested in the analysis of human motion. The programs in use were highly individualized to the needs of investigators within a particular institution. His premise, at that time, was that researchers must take a more generalized approach to software development. Programs available such as ANGLCALC (Hoffman and Warsham, 1971) were very efficient at

generating the values desired by individual investigators, but lacked the versatility and power to meet the diverse needs of researchers within human motion studies.

Researchers were making progress during the early nineteen-seventies in simulating movement patterns. The program MOTION (Baumgartner, 1971) was designed to give a representation of segment movement, including kinematic descriptions of the limbs. This would lead researchers into new fields such as computer graphics and simulation of movement.

Today's software for the analysis of human motion has adapted itself to directions being taken in biomechanics. Modeling of movement and the subsequent graphic display give the biomechanical investigator powerful tools in the analysis of motion. Availability of packaged software offers the researcher the technology to save vast amounts of time in data analysis and interpretation. For those who can afford the price of these packages, it is well worth the investment in eliminating time consuming data management and software development.

The future of biomechanical software is in user-friendly packages that allow for the complete analysis of movement from data acquisition to data analysis. Through a shared user environment utilizing multiple CRT's hooked to a mainframe, users can collaborate on research and standardize results. Programs must be written to include clear instructions, so that investigative tools are made accessible to coaches and educators in the field. The development of this package takes this approach in creating a user-friendly, menu-driven program to assist in selecting and implementing appropriate smoothing methods.

CHAPTER III

DEVELOPMENTAL METHODOLOGY AND EXPERIMENTAL PROCEDURES

The purpose of this investigation was to design, construct, and validate a general purpose data smoothing computer package to be used in the comparison of selected data smoothing techniques. The design of this program is based on current need as determined from: (1) a review of literature that assessed and compared smoothing methods for application in biomechanics and (2) a review of current research that reported the application of specific smoothing routines for biomechanical analysis. Based upon the results of a questionnaire sent to researchers in the area of biomechanical analysis of human movement (see Appendix A), subsequent verification and revisions of the software package have been implemented.

The first section of this chapter will consist of a discussion of the criterion used in selecting smoothing techniques for inclusion within the software package. Section two will detail the structure of the software package. Section three will be a discussion of the developmental procedures undertaken, including considerations for each smoothing routine. Section four presents the checks followed in verifying the fact that the package carries out the functions for which it was written. Finally, section five will be a discussion of the procedure followed in comparing the data smoothing techniques.

Criterion for Inclusion

In selecting the data smoothing methods to be included in the computer software package, several factors had to be considered. First, the purposes of fitting equations to data will be reviewed to develop rationale for smoothing data. According to Daniel and Wood (1971), the main considerations for fitting equations to data are: (1) to summarize a mass of data in order to obtain interpolation formulae or calibration curves, and (2) to confirm or refute a theoretical relation; to compare several sets of data in terms of the constants in their representing equations; to aid in the choice of a theoretical model. Second, variables that determine a good method of fitting should be listed to provide guidelines for choosing the smoothing routines. From Daniel and Wood (1971), a good fitting technique should include the following:

- (1) use all relevant data in estimating each constant,
- (2) maintain reasonable economy in the number of constants required

[Equations with large numbers of constants require excessive calculations to determine the constants involved.],

- (3) provide some estimate of the uncontrolled error in y [For example, in the equation; $y = mx + b$, y is the dependent variable.],
- (4) provide some indication of the random error in each constant estimated [Too great an error may indicate another method of smoothing would be more appropriate.],
- (5) make it possible to find regions of systematic deviations ("function bias") from the equation if any such exist [Such deviations could indicate a line of best fit which is not truly representative of the data being given.],
- (6) show whether the conclusions are unduly sensitive to the results of a small number of runs, perhaps even of one run [Converging on an answer too quickly may introduce error due to sudden changes in each numeric value.], and
- (7) give some idea of how well the final equation can be expected to predict the response-both in the overall sense and at important sets of conditions inside the region covered by the data.

Based upon these guidelines, a list of data smoothing techniques was prepared while conducting the review of literature for this study. From those listed, five routines were selected as being most frequently used by researchers (see Table 1). The utility of these methods is supported by the fact that they are applicable to the classifications of motion given in Figure 3 (Pezzack, Norman, and Winter, 1977; Wood, 1982). These techniques are: (1) least-squares approximation, (2) Chebyshev polynomials, (3) fourier series, (4) spline functions, and (5) Butterworth filter.

One additional method that deserves mention is the finite difference technique. This routine, used to calculate velocity and acceleration values is not included as a smoothing technique because it does not generate an equation which can be used to describe the smoothing of raw data. The method can be used following the smoothing process to calculate kinematic values.

An additional criterion used for inclusion within the program is the results of the questionnaire concerning existing software. Information gained from this questionnaire was used to support or refute the selections made on the basis of literature reviewed. Each method previously mentioned has been specifically investigated to determine whether or not biomechanical investigators deem it worth inclusion in data smoothing

software packages. By surveying the experts within the field of biomechanics of human motion, a better understanding of what is needed can be developed. The actual content of the program is also determined in this manner. Therefore, subroutines included in this program are the result of a thorough investigation, synthesizing the stated needs of researchers of human motion.

Structure of the Program

The first goal in the development of the program was to assure flexibility. The program is the second part of a three phase process in the analysis of human motion. This process consists of: (1) data acquisition, (2) data smoothing, and (3) data analysis. Due to memory availability in many microcomputers, data smoothing algorithms were developed as subroutines. A unique feature of this program is memory tracking. Each subroutine will erase memory allocated by the previous subroutine in order to maximize the availability of storage size.

Subroutines selected for inclusion within the package are:

- (1) a tutorial to assist the user in selecting the proper smoothing technique and instructions for running the program,
- (2) an input routine to allow for data to be entered either from key-board or diskette,
- (3) a data display for immediate viewing on a monitor,
- (4) a menu of smoothing routines to allow for proper techniques to be used according to the nature of the data,
- (5) a printer option for hardcopy of data, and
- (6) a graphics routine allowing for plotting of curves from any chosen data set (position, velocity, and acceleration values).

Developmental Procedures

The first step in developing the computer program was to decide which subroutines were to be included. These routines have been enumerated in Table 1. The next step was to interpret the mathematical theory of each smoothing technique into a usable algorithm that served as the basis for writing the code (See Figure 4). When each algorithm had been designed, the routines were written and tied together through a menu-driven control routine to form the program. A short description of the steps taken is provided. Each discussion will include: (1) a mathematical definition of the method, (2) steps in developing an appropriate algorithm, and (3) considerations taken in incorporating each routine as part of the overall package.

Least-squares approximation

The method of least-squares is one of the most universal techniques used for data smoothing. According to Daniel and Wood, the least-squares method says: "Find the values of the constants in the chosen equation that minimize the sum of the squared deviations of the observed values from those predicted by the equation." (1971, p. 6). In developing an equation to "fit" the data acquired by digitizing, the researcher strives to choose the equation which best describes the coordinates. The least-squares method accomplishes this task by considering the equation in a "piece-wise" fashion. It represents the true curve by a series of segments between points. When minimizing the sum of the squared deviations, this mathematically reduces these segments to the desired curve. In this way, the statistical variance is used to insure that the result is an unbiased estimate of the true curve. An unwritten assumption in utilizing this and other smoothing methods is that the data are "good". By this, it is meant that no wild points or outliers are contained within the data.

In developing an effective algorithm for the least-squares technique the first task is to seek constants a and b which minimize the least-squares equation:

$$\sum [y_i - (ax_i + b)]^2, \quad (3.1)$$

where $\sum$ is the summation of i from 1 to n . In order to solve for these constants, a system of two equations must be developed which effectively solve for this process. Once found, these equations are easily solved to give a single equation in terms of y . The corresponding x values are the raw data points.

Problems to be considered before linking the subroutine to the overall package are:

(1) determining which variables must be transferred from the main program and (2)

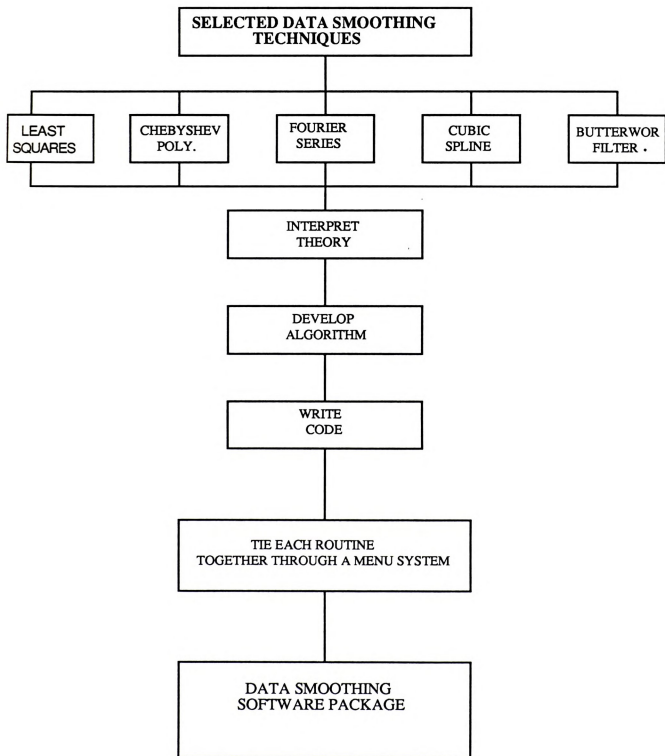


Figure 4. Flow chart of developmental procedures in creating a software package.

what information must be saved besides the smoothed data values for later review of the results. Raw data values, time coordinates, and indicator variables are transferred to the least-squares subroutine. Following the smoothing process, resulting data and the newly calculated indicator values are to be returned.

Spline functions

Spline functions are a recent mathematical tool developed in the 1960's. However, their development is well established in engineering. A spline is actually a flexible strip or ruler that is constrained to pass through a given set of points. When it is constrained in this way, the spline assumes a shape that minimizes its stored strain energy. Mathematically, a spline function is a connected group of cubic polynomials arranged so that adjacent curves join each other with continuous first and second derivatives. This concept is most easily understood by listing conditions in which a function is discontinuous. From Anton (1980), a function $y = f(x)$ at a point $x=c$ is discontinuous if:

- (1) the function, f , is undefined at c ; or
- (2) the limit, $\lim f(x)$, as x approaches c does not exist; or
- (3) the function, f , is defined at c and the limit, $\lim f(x)$, as x approaches c exists, but the value of the function at c and the value of the limit at c are different.

Cubic spline functions are one example of the general class of continuous functions. In order to construct a cubic spline, it is necessary to specify the coefficients that uniquely described each cubic polynomial between given data points. Selecting an appropriate spline for a given smoothing problem depends on the nature of the data being manipulated, and on what type of output is desired. In selecting the least-squares polynomial it was necessary to select the order of the equation. With higher-order polynomials it is often better to use a spline curve. Again, a higher-order polynomial is simply one whose highest power (highest exponential value within the equation) is greater than a certain value as determined by the investigator.

Allocation of internal storage for all smoothing routines is strictly on a demand basis. Arrays will be appropriately dimensioned according to the size of the data set being utilized. Once the smoothed values have been calculated, non-essential indicator variables will be erased. Upon subsequent runs of the smoothing routines, previously calculated data will be overwritten.

Chebyshev polynomials

The n TH Chebyshev polynomial is defined by: $T(x) = \cos(n \arccos x)$, where x is on the interval -1 to 1 (Dodes, 1978, p. 361). Essentially, the Chebyshev method is a trigonometric technique for fitting an equation to a given set of data. Because of the trigonometric nature of the method, repeating periodic and oscillatory types of harmonic motion may be represented quite well by this technique.

A unique characteristic of this method is that it remains "stable" for higher-order polynomials, and converges more rapidly on a solution than other techniques. An improper integral is said to "converge" if the limit exists (Anton, 1980, p. 539). A criterion imposed upon an algorithm whenever possible is that small changes in the initial data produce correspondingly small changes in the final results. A routine that satisfies this property is called "stable" (Burden and Faires, 1986).

A second feature of this technique is that it has many possible orders which may be selected to best fit the given data. For cyclic types of movement it has been shown that little signal exists beyond the seventh harmonic (Wood, 1980). Therefore, in order to maximize flexibility within the Chebyshev routine, a seventh-order polynomial was constructed. Most types of human activity fall within the range of the seventh harmonic. It has been suggested, however, that children's gait may contain more complex components (Foley, Quanbury, and Steinke, 1979).

Fourier series

"Given N data points with equally spaced data arguments, it can be shown that there exist N functions orthogonal over these data arguments which are either sine or cosine of a multiple of t " (Pizer, 1975, p. 393). These functions are collectively known as the Fourier series. Their general equation is:

$$a_j + \sum [a_j \sin(j \cdot 2\pi t / T) + b_j \cos(j \cdot 2\pi t / T)], \quad (3.2)$$

where $\sum$ is the summation of j from 1 to n . "Orthogonality" refers to the first definite integral on the range (a,b) equaling zero (Anton, 1980).

One of the most powerful of the numerical methods for its ability to approximate a curve, the Fourier series is also difficult to calculate and lengthy to process. The constraint placed on this routine is that the data points must be equally spaced in time. If the data points (t,y) are considered, with t representing the time coefficient, it is obvious that cinematographical data is appropriate for this technique; motor driven cameras provide a uniform time standard (t) that can be used to generate the Fourier coefficients.

Zarrugh and Radcliffe (1979) have suggested that at least twenty harmonics might be necessary to adequately describe the motion of the foot. However, for cyclic motions it has been generally shown that little signal exists beyond the seventh harmonic (Wood, 1980). Because of this wide range necessary to study human motion, a general fourier equation was developed applicable to the first through seventh harmonic. For comparison purposes, only the seventh harmonic will be used because the activity under investigation can be temporally controlled.

Butterworth filter

The Butterworth filter is a digital smoothing process designed to extract "noise" from data collected by electronic instrumentation. The theoretical design of a digital filtering system is given in Figure 1. According to Williams "The Butterworth filter is so popular because its passband and stopband are without ripples" (1986, p. 332). What the author means is that this filter represents the path of a curve without oscillating around the curve to a great extent. The passband and stopband refer to the waves used to represent the data curve. This can best be understood by means of an example. A typical sine function oscillates around the x-axis, with each peak the same distance from the ordinate. If each end of the curve could be stretched, the waves would be flattened at the peaks and brought closer to the x-axis. This new oscillatory curve would give a better representation of a straight line, with a correspondingly smaller passband and stopband.

Two design problems must be considered when writing a Butterworth filter program: (1) the order of the filter, and (2) the filter's cutoff frequency. For the purposes of this investigation, a low-pass filter was utilized as it was found to give the best approximation to typical human movements (Winter, 1979). The cutoff frequency is the sensitivity with which the filter can represent the desired curve. Low-pass filters contain good recursive properties, increasing their wave sensitivity (Pezzack, Norman, and Winter, 1977).

Validation of the Program

In order to validate whether or not this computer software package carries out the functions for which it is designed, resultant smoothed data values were compared against the corresponding raw data through the means square error. Data calculated from mathematical formulas contain a known error term of zero at each coordinate pair. Therefore, the summation of the differences between the smoothed values and exact values at their respective locations will yield a composite error term. This error term,

when divided by the number of coordinate pairs, results in the mean numeric error.

Zernicke, Caldwell, and Roberts (1977) utilized a similar method in determining mean percent error. They achieved 95.25 percent mean agreement in comparing force platform data of typical human gait. For the purposes of this investigation, a five percent or less difference between generated data and smoothed values was considered acceptable.

Comparison of Techniques

The purpose of this investigation was to design, construct, and validate the use of a computer program which could subsequently be used to make comparisons of selected data smoothing techniques.

Numeric data was generated from equations which was subsequently used to make the comparisons. This process was to contrast each of the classifications of motion enumerated in Table 1 using the respective smoothing techniques that had been selected for inclusion within the software package. Numeric data was generated from mathematical equations representative of the classifications of motion under investigation (See Appendix B). These values, that contain no error terms, were mathematically processed utilizing each smoothing routine. From the results of this procedure a ranking of smoothing techniques by classification of motion was constructed. The table will be an invaluable tool for researchers, coaches, and educators in helping them make informed decisions as to the appropriate smoothing method based upon the type of movement under investigation. This ranking system is an ideal situation in that no error is present within the raw data coordinates. Therefore, any error detected in the resulting smoothed values can be assumed to be the result of the mathematical routines. This particular approach for comparison was selected because of the problem of generating specific types of linear movement classifications to cinematographically record. Errors which are the result of photographic methods, the digitizing process, and subjective judgement of the researcher are unique to the investigation in which they are used. Therefore, in order to give further validity to the results of this study, the comparison process involved the use of exact numeric data.

Use of the Computer Program

The question of selection of smoothing routines is of increasing interest to researchers, engineers, and mathematicians. Selection of the appropriate method is critical in the analysis of data and has been the subject of numerous studies. This

investigation was developed to create a valuable tool for researchers, coaches, and educators, as well as, make a critical study of the techniques used in the program.

The main use of the computer software package is for investigators of human motion interested in eliminating "outliers" from their numeric data in order to gain a better picture of what the motion under investigation truly looks like. This would encompass a broad range of professionals from the pure research scientist interested in extending man's understanding of locomotion, to the clinician interested in quantifying abnormalities in the gait of patients. It is the purpose of this investigation to develop a tool which may be valuable to all who are interested in the investigation of motion.

Summary

The purpose of this study was to design, create, and validate a computer program which could be used to compare various data smoothing methods. The design of the program is based on a review of current literature, and the results of a questionnaire on software construction. Smoothing routines included are: (1) least-squares approximation, (2) Chebyshev polynomials, (3) cubic spline functions, (4) Butterworth digital filter, and (5) fourier series. The development of the program was on an IBM PC, making it compatible with other microcomputers. Each routine was validated with generated data which was known not to contain any error term. To carry out the comparison of the smoothing methods, data generated from the smoothing process was contrasted to the original data by the use of the mean numeric error. From these comparisons, a table ranking each technique according to the classification of motion being studied was constructed.

CHAPTER IV

RESULTS AND DISCUSSION

This chapter has been organized to present a detailed review of the pertinent findings resulting from the data smoothing software package developed for this investigation. Each of the five data smoothing routines enumerated in Table 1 were used to smooth data which was generated from equations written to represent the classifications of motion diagrammed in Figure 3. Subsequent numeric data values were quantitatively ranked according to their mean numeric error, which is a measure of the deviation from the original data. Part one of this chapter will present the results of a questionnaire sent to professionals within the field of cinematographical analysis of human motion to assess the present state of data smoothing software. Part two is a discussion of the comparison process, contrasting the relative merits of the five data smoothing methods as applied to the major classifications of human motion. Part three is a description of the use of the smoothing package in assisting researchers. Part four is a summary of the results.

Data Smoothing Information Questionnaire

A data smoothing information questionnaire (see Appendix A) was sent to professionals in the field of high-speed cinematographical analysis of human motion in order to assess the present state of data smoothing software used in research, and to determine which peripheral support may be deemed valuable in further enhancing the capabilities of these routines. A summary of the results of this questionnaire is presented in Table 2.

The majority of researchers polled (86%) responded positively to the existence of data smoothing software at their College/University. This would support the fact that computer software is considered a valuable tool in assisting researchers with the analysis of cinematographical data. For those who responded negatively (14%), an available data smoothing software package may prove invaluable in saving time and labor in the duplication of effort.

Fifty percent of researchers questioned indicated that multiple smoothing techniques were available with the software they are currently using. This value would seem to indicate that much of the time a single routine is applied to motion data regardless of the nature of the motion under investigation.

Questionnaire Items	-----		Yes	No			
Data smoothing software available at your college?			86* (12)**	14 (2)			
Multiple smoothing techniques available?			50 (7)	50 (5)			
Manual/tutorial to assist users?			44 (4)	55 (5)			
User view data on monitor?			83 (10)	17 (2)			
Computer printouts available?			100 (12)				
Graphic displays be plotted?			92 (11)	8 (1)			

	FD	LS	CS	FS	BF	CP	QS
Data smoothing techniques supported?	12 (4)	12 (4)	24 (8)	15 (5)	21 (7)	09 (3)	09 (3)

	Key		Floppy	Main	Digitizer	Harddisk	
Input Devices?	27 (8)		27 (8)	17 (5)	20 (6)	10 (3)	

	Raw	Velocity	Accel.	Compar.	Force		
Displays available?	29 (10)	31 (11)	29 (10)	06 (2)	06 (2)		

	Yes		No	Not		Certain	
Current software meets needs?	45 (5)		45 (5)	5 (1)			

Table 2 Summary of Data Smoothing Information Questionnaire.

**Numbers in parentheses represent frequency of responses.

* Numeric values without parentheses are percentages.

The third question addressed the problem of proper manuals/tutorials to assist the researcher in the implementation of the smoothing routines. Fifty-five percent responded that no manual or tutorial existed to aid them in the utilization of their software. Whether this manual were to be in published form, or incorporated within the software, for those researchers unfamiliar with the various smoothing methods this could create great difficulty in the implementation of their routines.

Questions four through six can be effectively considered as a group of items related to the common peripheral devices utilized in computer usage. Specifically, these items ask about the existence of the output devices which may be selected in order to view the resulting data generated from the smoothing routine utilized. In each case, the majority responded affirmatively. Monitors (83%), printouts (100%), and graphics displays (92%) are used by most researchers in order to display the resultant data from their routines. Based upon these findings, each of the output devices selected for this software package can be deemed warranted.

Column headings for question seven are two letter abbreviations for the smoothing methods in Table 1 (e.g. FD=FINITE DIFFERENCE). Two additional methods have been added to this table based upon their frequency of response. CP (cubic polynomial) and QS (quintic spline) are data smoothing methods similar in nature to the cubic spline function which have been utilized by nine percent of the researchers polled in each case. Although these routines have not been incorporated within the software package, they can be considered for inclusion upon subsequent revisions of the program.

Among the routines which were included in the software package, the cubic spline function was the most popular (24%) of the smoothing methods. This is followed by the Butterworth filtering (21%), fourier series (15%), and the finite difference and least squares routines with 12% each. Since the researchers polled in this questionnaire were encouraged to list any smoothing method which was not specifically mentioned within the questionnaire, the top five routines are supported by these results.

Twenty-seven percent responded positively to the use of a keyboard or floppy disk as an input device. The remaining devices, in order, include a digitizer (20%), mainframe computer (17%), and hard disk (10%). Each of these input devices are supported either directly by the software package or indirectly by an intermediate device.

In order to determine what types of graphics displays would be considered valuable, the researchers were asked to list all curves which could be plotted by their software. Velocity curves were the most popular choice (31%), followed by the raw data and acceleration curves (29%), and force and comparison curves (6%). Most commercially available software packages have multiple graphics routines that may be selected.

The last question on the questionnaire asked whether current software in general is

considered to meet the needs of researchers, coaches, and physical educators within the field of high-speed cinematographical analysis of human motion. Forty-five percent responded positively, 45 percent negatively, and 5 percent uncertain. Although no majority responded either "yes" or "no" to the question, the fact that 45 percent responded negatively indicates that further advancement of software would certainly be welcomed by those dissatisfied with current data smoothing packages either written to meet the specific needs of the researcher, or provided commercially.

Perhaps the most valuable information gathered by this questionnaire was the general comments provided by the researchers. In the words of one prominent educator: "I would like to see a program which offers a selection of data smoothing techniques, along with guidelines (written &/or on-line) to assist in selecting the proper smoothing technique." Similar comments related to the difficulty in locating and operating appropriate smoothing methods were to be found in the majority of the questionnaires. With this additional information to draw from, the following section will present a detailed review of both the validation of the individual smoothing routines and the comparisons made in contrasting the relative merits of each algorithm.

Comparison of Smoothing Methods

In order to establish a basis upon which the smoothing routines written for this investigation could be compared, data was generated by pre-written equations to represent each of the classifications of motion diagrammed in Figure 3 (See Appendix B). The first step in this process was to validate the usefulness of each routine by comparing the generated data values to the corresponding smoothed data values. This was accomplished by calculating the mean square error. A five percent mean square error or less was established as acceptable. The second step was to contrast the generated mean square error values in order to assess the relative merits of each data smoothing routine as applied to the generated data.

Validation

In order to assess the validity of each smoothing routine data was generated by pre-written equations representing each of the classifications of motion diagrammed in Figure 3 (See Appendix B). The resulting data was subsequently smoothed utilizing each of the data smoothing methods. Software was then written to calculate the mean square error based upon the equation:

$$[\sum(x-x_1)^2/n],$$

where x is the raw data values, x_1 is the smoothed data values and n is the number of data values. The mean square error was calculated for each of the classifications of motion diagrammed in Figure 3. A five percent mean square error was established acceptable as outlined by Zernicke, Caldwell, and Roberts (1977). It should be noted that the mean square error will be greater for larger data values and so the percent error gives a better value for comparison.

Each of the smoothing routines were found to be acceptable within the limits established (See Table 3). However, every routine did not meet the requirements for each classification of motion. This was to be expected as the individual smoothing routines adapt themselves to varying types of data. Table 3 lists the mean square error for each (x,y) coordinate data list found in Appendix B.

The natural cubic spline function was found to be a special case during the coding phase of this investigation. Because of the nature of the natural cubic spline, the selection of a predetermined power for the polynomial function is not possible. Rather, the natural cubic spline adapts itself to fit through each data coordinate pair creating a function whose power is equal to the number of coordinate pairs. For this reason, the mean square error associated with the natural cubic spline will always be equal to zero as denoted in Table 3. Although this makes validation of the natural cubic spline unnecessary, it creates difficulty in the comparison of the routine as will be discussed later.

Comparison of techniques

In order to compare the data smoothing routines written for this investigation the mean square error was again used. Since each routine was utilized to smooth the same data sets, the resulting mean square error values could be directly contrasted in order to quantitatively rank the routines. The natural cubic spline function was not directly compared to the other routines because of a mean square error of zero. However, it will be included in the discussion.

The results of contrasting the data smoothing routines are presented in Table 4. Motion classifications as diagrammed in Figure 3 are listed in the first row. Smoothing routines appear in descending order below each classification based upon their mean square difference values. The natural cubic spline function does not appear in the table due to a constant mean square error of zero.

The Chebyshev polynomial produced the lowest mean square error for the constant linear data, varying linear data, varying nonlinear data, constant harmonic data, and

MOTION CLASSIFICATION		DATA SMOOTHING TECHNIQUES			
TYPES	LEAST SQUARE	SPLINE** FUNCTION	FOURIER SERIES	CHEBYSHEV POLY.	BUTTERW. FILTERING
CONSTANT LINEAR	x=1.08E 07 y=1.08E 07	* x=0 * y=0	* x=2.314E -02 y=6.239E -02	* x=7.386E -05 y=1.991E -04	x=3.29E 16 y=6.03E 16
VARYING LINEAR	x=5863.681 y=5865.12	* x=0 * y=0	x=233089.8 y=233192.7	x=743.7868 y=744.1159	x=918 y=918
CONSTANT NONLINEAR	* x=12.65 y=2.51E 08	* x=0 y=0	x=399.6757 y=3.169E 09	* x=1.275359 y=1.0112E 07	* x=0 y=60979.7
VARYING NONLINEAR	* x=.496 y=35912.65	* x=0 * y=0	* x=17.88722 y=490854.8	* x=5.707E -02 y=1566.317	* x=1.6E 03 y=22.783
CONSTANT HARMONIC	* x=23.537 y=.512	* x=0 * y=0	* x=925.9224 y=.1180212	* x=2.954617 y=3.766E -04	* x=0 y=4.058
VARYING HARMONIC	* x=53.541 y=.191	* x=0 * y=0	* x=1970.269 y=.1100592	* x=6.287112 y=3.512E -04	* x=16.32 y=2.519

Table 3. Mean Square Difference for Exact Numeric data.

* Denotes a mean square difference less than five percent.

** No error produced by natural cubic spline.

MOTION CLASSIFICATION*

CONSTANT LINEAR	VARYING LINEAR	CONSTANT NONLINEAR	VARYING NONLINEAR	CONSTANT HARMONIC	VARYING HARMONIC
CHEBYSH POLY.	CHEBYSH POLY.	BUTTERW FILTER	CHEBYSH POLY.	CHEBYSH POLY.	CHEBYSH POLY.
FOURIER SERIES	BUTTERW FILTER	CHEBYSH POLY.	LEAST SQUARE	BUTTERW FILTER	BUTTERW FILTER
LEAST SQUARE	FOURIER SERIES	LEAST SQUARE	FOURIER SERIES	LEAST SQUARE	LEAST SQUARE
BUTTERW FILTER	LEAST SQUARE	FOURIER SERIES	BUTTERW FILTER	FOURIER SERIES	FOURIER SERIES

Table 4. Data Smoothing Routine Ranking by Motion Classification in Descending Order.

*Rankings are based upon mean square difference.

varying harmonic data. The natural cubic spline function produced the only consistently lower mean square difference values. All routines used a seventh order polynomial to smooth the data where applicable.

The Butterworth filtering routine, which is a low-pass filter, produced the next best results. This routine must be considered slightly biased because of the nature of the technique. Unlike the other smoothing techniques used in this investigation the Butterworth filter does not produce a polynomial through a given discrete data set. Rather, the data is scaled according to certain criteria set down by the user. For this investigation, high frequency (outliers) data points were filtered out of the raw data set in order to produce a smoother curve through the resulting data.

The least-squares routine, which was concluded the most popular of the smoothing techniques in the review of literature, produced the next most consistent results. This routine produced consistently good data curves for all classifications of data. Although this technique is certainly acceptable for most types of human motion, the popularity of this routine may need to be reexamined based upon the results of this investigation.

The fourier series was found to consistently produce the greatest mean square error. Although these results may seem to lead to the conclusion that this is the poorest of the routines selected, it should be noted that the fourier series did produce acceptable results four out of six motion classifications.

The rankings produced by comparing the mean square error of each routine should be considered with caution. The nature of smoothing routines is to produce a certain amount of error even under exact mathematical calculation. Through the interpretation of these routines and the subsequent coding for computer use more error is inevitably introduced. Each computer represents data in a slightly different manner. Investigators code their programs according to their own expertise. The error introduced by these factors is random. Therefore, the best the prudent researcher can do is to minimize this random error and compare the results to those previously reported.

The quantitative ranking system presented is not meant to be all-inclusive. The tables presented here are the result of comparisons at a generally accepted power for smoothing polynomials. This was done both to add validity to the routines developed for this investigation and to help provide information concerning the proper smoothing technique. The results of this study should not be extrapolated to all smoothing packages.

It was noted in Table 3 that the natural cubic spline function developed for this study produced no error. This routine is designed to fit a polynomial through each data coordinate pair. The result is a function which passes through all points including outliers. This should be considered before using the cubic spline routine in order to avoid outliers in the final smoothed data.

Using the Software Package

The purpose of this investigation was to develop a computer software package which could assist the researcher of human motion in choosing and implementing appropriate smoothing techniques. A tutorial was written as part of the software package to fulfill the goal of assisting in the selection of the appropriate technique. As an example, consider the constant nonlinear (angular) data found in Appendix B.

The tutorial is useful in selecting the appropriate data smoothing technique where no a priori knowledge of the data or technique is available. The main menu shown in figure 5 is the first screen given to the user upon booting the package. Choose selection one, the tutorial in order to find information concerning the use of the smoothing routines and peripheral support routines. Figure 6 shows the main tutorial selection menu. Each subroutine, including the smoothing techniques have complete explanations of its use and applicability.

Figure 7 is an example of a tutorial screen for least squares approximation. Similar tutorials are provided for each smoothing routine. Once the appropriate technique has been selected, the data can be input through the data input routine. Figure 8 shows the main selection menu for inputting data. If the data is stored on disk, figure 9 shows the screen for entering the appropriate drive and filename for retrieval.

In order to smooth the data, a selection of techniques is offered as shown in figure 10. After smoothing, the data may be plotted in order to evaluate the results. Figure 11 shows a graph of the nonlinear data and subsequent smoothed data generated by each of the routines.

Summary

This chapter presented the pertinent findings resulting from the data smoothing software package developed for this investigation. Validation of the smoothing routines was accomplished by calculating the mean square error produced by smoothing numeric data. All smoothing routines written for this study were found to be acceptable at a predetermined level of ninety-five percent mean square agreement. In contrasting the smoothing techniques it was found that in the routines which consistently produced the best results, in descending order were: (1) natural cubic spline function, (2) Chebyshev polynomial, (3) Butterworth filter, (4) least squares polynomial, and (5) fourier series. It was noted, however, that the natural cubic spline produced no error and must be considered seperately.

DATA SMOOTHING PROGRAM
SCHOOL OF HEALTH EDUCATION, COUNSELING
PSYCHOLOGY & HUMAN PERFORMANCE
MICHIGAN STATE UNIVERSITY

TUTORIAL	1
INPUT DATA.....	2
SAVE DATA TO DISK.....	3
VIEW DATA AT TERMINAL.....	4
SMOOTH DATA	5
SEND DATA TO PRINTER	6
GRAPH DATA	7
EXIT PROGRAM	8

Figure 5. Main menu of data smoothing software package.

DATA SMOOTHING PROGRAM
TUTORIAL

PLEASE SELECT ONE OF THE FOLLOWING:

WHAT IS DATA SMOOTHING.....	1
HOW DO I INPUT DATA	2
HOW DO I SAVE DATA.....	3
HOW DO I VIEW DATA.....	4
HOW DO I PRINT DATA	5
HOW DO I GRAPH DATA	6
CHEBYSHEV POLYNOMIALS	7
LEAST SQUARES APPROXIMATION	8
FOURIER SERIES	9
CUBIC SPLINE FUNCTION.....	10
BUTTERWORTH FILTERING	11
RETURN TO MAIN MENU	12

Figure 6. Tutorial menu screen.

LEAST SQUARES APPROXIMATION

Least squares approximation is based upon the ideal that every value has a certain deviation from its true value (an error value), the least squares technique takes into account the deviations of all data points & attempts to minimize the error. This method is highly recommended for biomechanical data as it is very powerful at giving a smooth curve representation of a set of data points.

PRESS ANY KEY TO RETURN TO MENU

Figure 7. Tutorial screen from least squares approximation.

INPUTTING DATA FROM DISK

PLEASE SELECT ONE OF THE FOLLOWING:

- | | |
|---|---|
| INPUT DATA SMOOTHING DATA FROM DISK..... | 1 |
| INPUT DATA ACQUISITION DATA & FORMAT..... | 2 |
| RETURN TO MAIN MENU..... | 3 |

YOUR SELECTION PLEASE:

Figure 8. Main screen for inputting data from disk.

YOUR FILE NAME & DISK DRIVE ID
ARE ENTERED IN THE FOLLOWING FORM:
FOR DISK DRIVE: 'B' & FILENAME: 'SAVE'
B:SAVE

FILENAME PLEASE:

Figure 9. Entering drive and filename for data retrieval from disk

DATA SMOOTHING TECHNIQUES

PLEASE SELECT ONE OF THE FOLLOWING:

CHEBYSHEV POLYNOMIALS.....	1
LEAST SQUARES APPROXIMATION	2
FOURIER SERIES	3
CUBIC SPLINE FUNCTION	4
BUTTERWORTH FILTERING	5
CALCULATE VELOCITY OF RAW DATA.....	6
CALCULATE ACCELERATION OF RAW DATA.....	7
RETURN TO MAIN MENU.....	8

Figure 10. Main menu for data smoothing techniques.

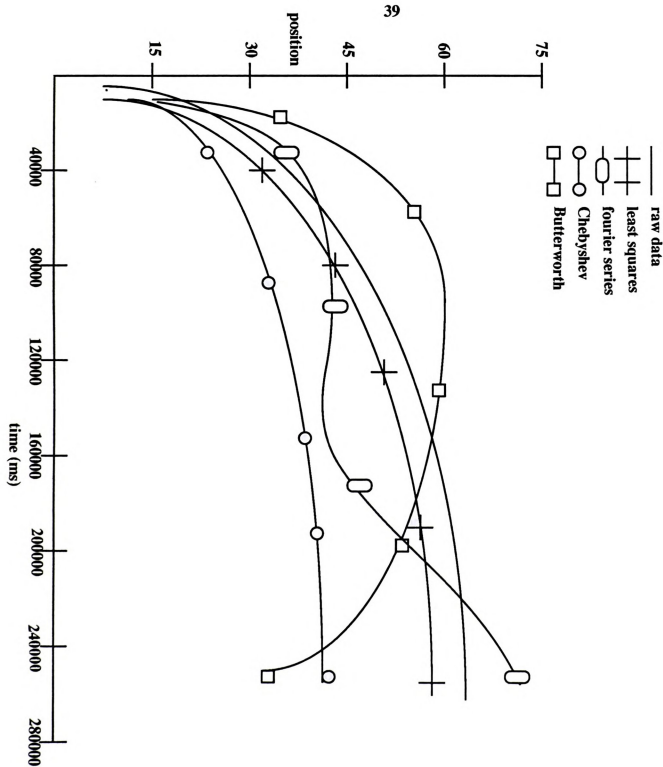


Figure 11. Smoothed data curves from various data smoothing techniques.

CHAPTER V

CONCLUSIONS

Summary

The purpose of this investigation was to create a flexible computer software package to assist the researcher of human motion in selecting and implementing appropriate data smoothing techniques. This was accomplished by including a tutorial in the package to instruct the researcher about the proper smoothing method to select for each major classification of human motion. The package created for this study is the second part of a three-phase process in the analysis of human motion through high speed cinematographical techniques. This process consists of: (1) data acquisition, (2) data smoothing, and (3) data analysis.

The development of the computer software package took place over a two-year period. The package was written on a IBM AT microcomputer and can be used with all IBM and IBM compatible brand microcomputers equipped with color graphics capability. The following is a list of the main characteristics of the package:

- (1) a tutorial to assist the user in selecting the proper smoothing technique, and instructions for running the program,
- (2) an input routine to allow for data to be either entered from keyboard or diskette,
- (3) a data display for immediate viewing on a monitor,
- (4) a selection of smoothing routines to allow for proper techniques to be used according to the nature of the data,
- (5) a printer option for hardcopy of data, and
- (6) a graphics routine allowing for plotting of curves from any chosen data set (position, velocity, and acceleration values).

Findings

The findings of this investigation were the result of two sources. First, a data smoothing information questionnaire was sent to professionals in the field of high-speed cinematographical analysis of human motion in order to assess the present state of data smoothing software used in research, and to determine which peripheral support may be deemed valuable in further enhancing the capabilities of this package. Second, comparisons of raw data values to smoothed values were made by the mean square error. The mean square error was then used to quantitatively rank each smoothing routine according to the

major classifications of human motion.

The results of the questionnaire have been utilized in the design of the data smoothing package. Responses to direct questions regarding the capabilities of existing software gave insight into the needs of researchers of human movement. The responses indicated that although the majority polled (86%) had some kind of data smoothing software available, the capabilities of the software were diverse. The responses also indicated that 45% of those questioned were dissatisfied with the current state of data smoothing software. The results of the questionnaire will also be used in the recommendations for further research and development.

The first part of the comparison process was to validate the effectiveness of the smoothing routines. This was accomplished by determining the mean square error for each routine. Every routine was found to be acceptable within the limits set of ninety-five percent mean square agreement. It was noted that for the natural cubic spline function no error term was generated.

The second part of the comparison process was to compare each routine according to the major classifications of human motion. The routines were quantitatively ranked according to their mean square error. The natural cubic spline function was not directly compared to the others. The routines which consistently produced the best results, in descending order were: (1) natural cubic spline function, (2) Chebyshev polynomial, (3) Butterworth filter, (4) least squares polynomial, and (5) fourier series.

Implementation

The data smoothing software package developed for this investigation was designed to be used in conjunction with existing software at Michigan State University. However, one of the purposes of this study was to make the software as flexible and user-friendly as possible. For this reason the software may be utilized by anyone interested in the reduction of numeric data. The peripheral support developed within the package is designed to allow for input of data from various sources.

The areas of research interested in the use of this package range from the analysis of human gait to the reduction of accelerometer data from impact testing. It is the hope of the author that the software developed for this investigation will meet the needs of a broad array of researchers where the reduction of numeric data to usable form is necessary.

Recommendations for Further Research and Development

It is the nature of software to be continuously updated and improved. As research advances our understanding in all fields of science, the tools we use for research must also be improved. Based upon the recommendations of professionals in the field of high speed cinematographical analysis of human motion, the following recommendations are offered.

- (1) The data smoothing software package should be incorporated into a package for the complete analysis of high speed film, including: (a) data acquisition, (b) data smoothing, and (c) data analysis.
- (2) The graphics capabilities of the package should be expanded to include comparisons of various body segments.
- (3) The data smoothing routines written for this package should be evaluated on polynomials of varying degree.

REFERENCES

REFERENCES

- Anton, A., Calculus with analytic geometry, John Wiley and Sons, Inc., 1980.
- Barham, J.N., Mechanical Kinesiology. The C. V. Mosby Company, Saint Louis, 1978.
- Bates B.T., "The Development of a Computer Program with Application to a Film Analysis: The Mechanics of Female Runners.", Ph.D. Dissertation, Indiana University., 1973.
- Baumgartner, T., "Using the Computer and the Computer Program MOTION.", in Selected Topics on Biomechanics: Proceedings of the C.I.C. Symposium on Biomechanics. (ed. by J.M. Cooper) Chicago: Athletic Institute, 191-194., 1971.
- Blakesley, J.F. "Administration -- Integrated Records and Proceedings", Computers and Education, edited by Ralph N. Gerard, McGraw Hill Book Co., St. Louis , 1967, 307pp.
- Burden, R.L., and Faires, J.D., Numerical Analysis, Prindle, Weber & Schmidt, Boston, 1985 (3rd edition), 676pp.
- Daniel, C., and Wood, F.S., Fitting Equations to Data. John Wiley and Sons, Inc., 1971.
- Dodes, I.A., Numerical Analysis for Computer Science. Elsevier-Holland, Inc., New York., 1978.
- Foley, C.D., Quanbury, A.O., and Steinke, T., "Kinematics of Normal Child Locomotion - A Stationary Study Based on T.V. Data.", J. Biomech. 12: 1-6, 1979.
- Greville, T. Theory and Applications of Spline Functions. New York, Academy Press, 1969.
- Hoffman, S.J., and Warsham, K., "ANGLCALC: A Computer Program for Descriptive Analysis of Movement Used During Repeated Performances of Skills.", in Kinesiology Review Washington, D.C.: American Association for Health, Physical Education and Recreation, 55-59., 1971.
- Muybridge, E., Animals in Motion. Chapman & Hall, London. (c1899).

- Pezzack, J.C., Norman, R.W., and Winter, D.A., "An Assessment of Derivative Determining Techniques Used for Motion Analysis. *Journal of Biomechanics*, 10: 377-382, 1977.
- Pizer, S.M., Numerical Computing and Mathematical Analysis. Science Research Associates, Inc., 1975.
- Pizer, S.M., and Wallace, V.C., To Compute Numerically. Concepts and Strategies. Little, Brown and Company., 1983.
- Rodgers, M.M., and Cavanagh, P.R., "Glossary of Biomechanical Terms, Concepts, and Units." *Physical Therapy*. 64(12), December, 1984.
- Shapiro, R., "Direct Linear Transformation Method for Three-Dimensional Cinematography." *Research Quarterly*. 49(2): 197-205, May, 1978.
- Walton, J.S. "Close-range cine-photogrammetry: a generalized technique for quantifying gross human motion." *Ph.D. Dissertation*. University Park, PA: The Pennsylvania State University, May, 1981.
- Ward, R., "An Investigation into the Use of Computer Integration of Kinematics, Kinetics, and Cinematography Data in Motion Analysis.", *Ph.D. Dissertation*, Indiana University., 1971.
- Williams, C.S., Designing Digital Filters, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1986.
- Winter, D.A., Biomechanics of Human Movement. New York: John Wiley and Sons, 1979.
- Wood, G.A., "Data Smoothing Differentiation Procedures in Biomechanics.", *Exercise and Sport Science Reviews*, Volume 10, 1982.
- Zarrugh, M.Y., and Radcliffe, C.W., "Computer Generation of Human Gait Kinematics.", *J. Biomech.*, 12: 613-621, 1979.
- Zernicke, R.F., Caldwell, G., and Roberts, E.M., "Fitting Biomechanical Data with Cubic Spline Functions.", *Research Quarterly*, 47(1): 9-19, March., 1976.

GENERAL REFERENCES

GENERAL REFERENCES

- Belkin, G.S. Twelve-Hour Basic for the IBM PC Compatibles. McGraw-Hill Book Company, 1986.
- Bernstein, N. The Co-ordination and Regulation of Movement. New York: Pergamon Press, 1967.
- Cuthbert, D., and Wood, F.S. Fitting Equations to Data. John Wiley & Sons, Inc., 1971.
- Douglas, B. Numerical Basic. Howard W. Sams & Co., Inc., Indianapolis, Indiana, 1973.
- Forsythe, G.E. "Generation and the Use of Orthogonal Polynomials for Data Fitting with a Digital Computer." *Journal of the Society for Industrial Applications of Mathematics*. 5:74-88, 1957.
- Garrett, R.E., Widule, C., and Garrett, G.E. "Computer-Aided Analysis of Human Motion." in *Kinesiology Review* Washington D.C.: American Association for Health, Physical Education and Recreation, 55-59, 1971.
- Gustavson, F.G., and Sackson, M.V. "problem Solving and BASIC. A Modular Approach." Science Research Associates, Inc., Chicago, 1986.
- Hamming, R.W. Digital Filters. Prentice-Hall Inc., 1977.
- Henrici, P. Essentials of Numerical Analysis with Pocket Calculator Demonstrations. John Wiley & Sons, Inc., 1982.
- Jackson, K.M. "Fitting of Mathematical Functions to Biomechanical Data." *IEEE Transactions on Biomedical Engineering*. 26(2): 122-124, February, 1979.
- James M.L., Smith, G.M., and Wolford, J.C. Applied Numerical Methods for Digital Computation with FORTRAN. International Textbook Company, 1967.
- James, S.L., and Brubaker, C.E. "Biomechanical and Neuromuscular Aspects of Running." In: *Exercise and Sport Sciences Reviews*, J.A. Wilmore (Ed.), New York, Academic Press, 1973.
- Johnston, R.L. Numerical Methods A Software Approach. John Wiley & Sons, Inc., 1982.
- Lees, A. "An Optimized Film Analysis Method Based on Finite Difference Techniques." *Journal of Human Movement Studies*. 6: 165-180, 1980.

- McCormick, J.M., and Salvadori, M.G. Numerical Methods in FORTRAN. Prentice-Hall Inc., Englewood Cliffs, New Jersey, 1964.
- McLaughlin, T.M., Dillman, C.J., and Lardner, T.J. "Biomechanical Analysis with Cubic Spline Functions." *Research Quarterly for Exercise and Sport*. 48: 569-582, 1977.
- Patrick, S.B., Widule, C.B., and Hillberry, B.M. "Digital Filtering of Cinematographical Data.", in *Biomechanics: Proceedings of the C.I.C. Symposium on Biomechanics*. (ed. by J.M. Cooper, and B. Haven) The Indiana State Board of Helath. 202-213, 1980.
- Petak, K.L. "Acquisition and Reduction of Biomechanical Data by Mini-Computer.", in *Selected Topics in Biomechanics: Proceedings of the C.I.C. Symposium on Biomechanics* (ed. by J.M. Cooper) Chicago: Athletic Institute. 201-205, 1971.
- Shoup, T.E. Numerical Methods for the Personal Computer. Prentice-Hall Inc., Englewood Cliffs, New Jersey, 1983.
- Widule, C.J., and Gossard, D.C. "Data Modeling Techniques in Cinematographic Biomechanical Analysis." In: *Biomechanics III*, E. Jökl (Ed.) Basel: S. Karger, 1973.
- Winter, D.A., Sidwall, H.G., and Hobson, D.A. "Measurement and Reduction of Noise in Kinematics of Locomotion." *Journal of Biomechanics*. 7: 157-159, 1974.
- Wold, S. "Spline Functions in Data Analysis." *Technometrics*. 16: 1-11, 1974.

APPENDIX A

DATA SMOOTHING INFORMATION QUESTIONNAIRE

DIRECTIONS

Please respond to each item by circling the letter preceding the appropriate response. The questionnaire is designed to gather information on up to four individual programs. Please list each program separately and answer the corresponding questions.

Your assistance is greatly appreciated

- 1.) Do you currently have data smoothing software at your College/ University? (If NO, go to question # 11.)

a. Yes
b. No

- 2.) Please state the name(s) of your program(s).
(SPACE IS PROVIDED FOR FOUR INDIVIDUAL PROGRAMS)

(I) _____ (II) _____ (III) _____ (IV) _____

- 3.) Referring to the program(s) listed above; are multiple smoothing techniques available to the user? (i.e. more than ONE)
(If NO, go to question #5.)

(I) a. Yes (II) a. Yes (III) a. Yes (IV) a. Yes
b. No b. No b. No b. No

- 4.) Is there a manual or tutorial to assist users with the program(s) listed above in selecting the proper smoothing technique?

(I) a. Yes (II) a. Yes (III) a. Yes (IV) a. Yes
b. No b. No b. No b. No

- 5.) Which data smoothing technique(s) can be supported by the program(s)?
(Place ALL appropriate answers in the spaces provided.)

e.g. If program III has a finite difference technique and a butterworth filter, place (a) and (e) in the spaces following program III.

a. Finite Difference Technique	
b. Least Squares Approximation	PROGRAM(S)
c. Cubic Spline Function	
d. Fourier Series	(I) _____
e. Butterworth Filtering	(II) _____
f. Chebyshev Polynomials	(III) _____
Other (Please List)	(IV) _____
g. _____	
h. _____	
i. _____	

- 6.) Which input device(s) can be selected with the program(s)?
(Place ALL appropriate answers in the spaces provided.)

a. Keyboard	
b. Floppy Disk	PROGRAM(S)
c. Mainframe	
d. Digitizer	(I) _____
Other (Please List)	(II) _____
e. _____	(III) _____
f. _____	(IV) _____
g. _____	

- 7.) Can the user view data on a computer monitor with the program(s)?

(I) a. Yes	(II) a. Yes	(III) a. Yes	(IV) a. Yes
b. No	b. No	b. No	b. No

- 8.) Are computer printouts of data available with the program(s)?

(I) a. Yes	(II) a. Yes	(III) a. Yes	(IV) a. Yes
b. No	b. No	b. No	b. No

- 9.) Can graphic displays of data be plotted by the program(s)?
(If NO, go to question #11.)

(I) a. Yes	(II) a. Yes	(III) a. Yes	(IV) a. Yes
b. No	b. No	b. No	b. No

10.) Which displays can the program(s) plot?

(Place ALL appropriate answers in the spaces provided.)

	PROGRAM(S)
a. Raw Data Curve	
b. Velocity Curve	
c. Acceleration Curve	(I) _____
d. Comparison Curves of Techniques	
Other (Please List)	(II) _____
e. _____	
f. _____	(III) _____
g. _____	(IV) _____

11.) Do you feel that current data smoothing packages in general are sufficient to meet the needs of researchers, coaches, and educators?

- a. Yes
- b. No
- c. Not Certain

PLEASE USE THE REST OF THIS QUESTIONNAIRE TO DISCUSS ANY PROBLEMS, OPTIONS, OR TRENDS YOU FEEL MAY BE VALUABLE IN DESIGNING A DATA SMOOTHING PACKAGE.

YOUR COMMENTS ARE GREATLY APPRECIATED!

APPENDIX B

CONSTANT LINEAR DATA

.01	.23	
.02	.24	
.03	.25	
.04	.26	
.05	.27	
.06	.28	
.07	.29	
.08	.3	
8.999999E-02		.31
9.999999E-02		.32
.11	.33	
.12	.34	
.13	.35	
.14	.36	
.15	.37	
.16	.38	
.17	.39	
.18	.4	
.19	.41	
.2	.42	
.21	.43	
.22	.4400001	
.23	.4500001	
.2400001	.4600001	
.25	.4700001	
.26	.48	
.27	.49	
.28	.5	
.29	.51	
.3	.52	
.31	.53	
.32	.54	
.33	.55	
.34	.56	
.3499999	.57	
.3599999	.58	
.3699999	.59	
.3799999	.5999999	
.3899999	.6099999	
.3999999	.6199999	
.4099999	.6299999	
.4199999	.6399999	
.4299999	.6499999	
.4399999	.6599999	
.4499999	.6699999	
.4599999	.6799999	
.4699999	.6899999	
.4799999	.6999999	
.4899999	.7099999	
.4999999	.7199999	
.5099999	.7299999	
.5199999	.7399999	
.5299999	.7499999	
.5399999	.7599999	

VARYING LINEAR DATA

100	100.25
115	115.25
160	160.25
175	175.25
220	220.25
235	235.25
280	280.25
295	295.25
340	340.25
355	355.25
400	400.25
415	415.25
460	460.25
475	475.25
520	520.25
535	535.25
580	580.25
595	595.25
640	640.25
655	655.25
700	700.25
715	715.25
760	760.25
775	775.25
820	820.25
835	835.25
880	880.25
895	895.25
940	940.25
955	955.25
1000	1000.25
1015	1015.25
1060	1060.25
1075	1075.25
1120	1120.25
1135	1135.25
1180	1180.25
1195	1195.25
1240	1240.25
1255	1255.25
1300	1300.25
1315	1315.25
1360	1360.25
1375	1375.25
1420	1420.25
1435	1435.25
1480	1480.25
1495	1495.25
1540	1540.25
1555	1555.25
1600	1600.25
1615	1615.25
1660	1660.25
1675	1675.25

CONSTANT NONLINEAR (ANGULAR) DATA

12	1728
13	2197
14	2744
15	3375
16	4096
17	4913
18	5832
19	6859
20	8000
21	9261
22	10648
23	12167
24	13824
25	15625
26	17576
27	19683
28	21952
29	24389
30	27000
31	29791
32	32768
33	35937
34	39304
35	42875
36	46656
37	50653
38	54872
39	59319
40	64000
41	68921
42	74068
43	79507
44	85184
45	91125
46	97336
47	103823
48	110592
49	117649
50	125000
51	132651
52	140608
53	148877
54	157464
55	166375
56	175616
57	185193
58	195112
59	205379
60	216000
61	226981
62	238328
63	250047
64	262144

VARYING NONLINEAR (ANGULAR) DATA

.1	9.999999E-04
-.2	-8.000003E-03
-.5	-.125
-.8	-.512
-1.1	-1.331
-1.4	-2.744001
-1.7	-4.913
-2	-8
-2.1	-9.260998
-2.4	-13.824
-2.7	-19.483
-3	-26.99999
-3.3	-35.93699
-3.6	-46.65599
-3.9	-59.31899
-4.2	-74.08798
-4.5	-91.125
-4.8	-110.592
-5.100001	-132.651
-5.400001	-157.4641
-5.700001	-185.1931
-6.000001	-216.0001
-6.300001	-250.0471
-6.600002	-287.4962
-6.900002	-328.5092
-7.200002	-373.2483
-7.500002	-421.8754
-7.800002	-474.5524
-8.100003	-531.4415
-8.400002	-592.7045
-8.700002	-659.5026
-9.000003	-729.0008
-9.300003	-804.3578
-9.600003	-884.7369
-9.900003	-970.3001
-10.2	-1061.209
-10.5	-1157.626
-10.8	-1259.713
-11.1	-1367.633
-11.40001	-1481.546
-11.70001	-1601.615
-12.00001	-1728.002
-12.30001	-1860.869
-12.60001	-2000.378
-12.90001	-2146.692
-13.20001	-2299.971
-13.50001	-2460.378
-13.80001	-2628.075
-14.10001	-2803.225
-14.40001	-2985.988
-14.70001	-3176.527
-15.00001	-3375.005
-15.30001	-3581.582
-15.60001	-3796.421

CONSTANT HARMONIC DATA

2	.9092975
4	-.7568026
6	-.2794155
8	.9893582
10	-.5440212
12	-.536573
14	.9906074
16	-.2879034
18	-.7309873
20	.9129452
22	-8.351309E-03
24	-.9055784
26	.7625385
28	.2709058
30	-.9880317
32	.8514267
34	.5290827
36	-.9917789
38	.2963686
40	.7451132
42	-.5165216
44	1.770193E-02
46	.9017884
48	-.7682547
50	-.2623749
52	.9866276
54	-.5587891
56	-.521551
58	.9928726
60	-.3048107
62	-.7391807
64	.9200261
66	-2.655115E-02
68	-.8979278
70	.7738907
72	.2538224
74	-.9851463
76	.5661076
78	.5159785
80	-.9928887
82	.3132288
84	.7331904
86	-.9234585
88	.0353983
90	.8929966
92	-.7794661
94	-.245252
96	.9835878
98	-.5733819
100	-.5063657
102	.9948269
104	-.3216224
106	-.7271425
108	.9268186

VARYING HARMONIC DATA

1	-.4161459
2	-.9899925
7	-.1455
8	-.9111303
13	.1367372
14	-.7596879
19	.4080621
20	-.5477293
25	.6469194
26	-.2921388
31	.8942234
32	-1.327675E-02
37	.956737
38	.266443
43	.9995494
44	.525322
49	.9649461
50	.7421543
55	.8532202
56	.8998669
61	.6735071
62	.9858966
67	.4401431
68	.9932904
73	.1717173
74	.9217512
79	-.1103872
80	.776686
85	-.3836985
86	.5697504
91	-.6264445
92	.3174287
97	-.8192883
98	3.982098E-02
103	-.9436681
104	-.2409591
109	-.9990209
110	-.5025443
115	-.9715922
116	-.7240973
121	-.8667671
122	-.8879689
127	-.6928958
128	-.9611056
133	-.4638289
134	-.9964879
139	-.1978136
140	-.9317224
145	8.395944E-02
146	-.7931345
151	.3590443
152	-.5913697
157	-.6055279
158	-.3424948

APPENDIX C

```

100 DIM X1(500),Y1(500),NP(100),PH$(100),TV(100) :
DP = 1
200 DIM BF1(1),BF2(1),CS1(1),CS2(1),CP1(1),CP2(1),FS1(1),FS2(1),LS1(1),LS2(1)
300 REM:
400 REM:
1000 REM: ALL OTHER PROGRAMS BRANCH TO THIS POINT.
1100 REM:
1200 REM:
1300 KEY OFF
3800 REM:
3900 CLS
4000 LOCATE 1,15
4100 PRINT "*****"
4200 LOCATE 2,15
4300 PRINT "*"
4400 LOCATE 3,15
4500 PRINT "*" : COLOR 0,7 : LOCATE 3,30 : PRINT "DATA" : COLOR 7,0 : LOCATE 3,3
5 : COLOR 0,7 : PRINT "SMOOTHING" : COLOR 7,0 : LOCATE 3,45 : COLOR 0,7 : PRINT
"PROGRAM" : COLOR 7,0 : LOCATE 3,67 : PRINT "*"
4600 LOCATE 4,15
4700 PRINT "*"
4800 LOCATE 5,15
4900 PRINT "*" : COLOR 0,7 : LOCATE 5,22 : PRINT "SCHOOL OF HEALTH EDUCATION, CO
UNSELING" : COLOR 7,0 : LOCATE 5,67 : PRINT "*"
5000 LOCATE 6,15
5100 PRINT "*"
5200 LOCATE 7,15
5300 PRINT "*" : COLOR 0,7 : LOCATE 7,26 : PRINT "PSYCHOLOGY & HUMAN PERFORMANCE
" : COLOR 7,0 : LOCATE 7,67 : PRINT "*"
5400 LOCATE 8,15
5500 PRINT TAB(15) "*" TAB(67) "*"
5600 LOCATE 9,15
5700 PRINT "*" : COLOR 0,7 : LOCATE 9,29 : PRINT "MICHIGAN STATE UNIVERSITY" : COLOR 7,0 :
LOCATE 9,67 : PRINT "*"
5800 LOCATE 10,15
5900 PRINT "*"
6000 LOCATE 11,15
6100 PRINT "*****"
6200 LOCATE 13,19
6300 PRINT "TUTORIAL..... 1"
6400 LOCATE 14,19
6500 PRINT "INPUT DATA..... 2"
6600 LOCATE 15,19
6700 PRINT "SAVE DATA TO DISK..... 3"
6800 LOCATE 16,19
6900 PRINT "VIEW DATA AT TERMINAL..... 4"

```

```

7000 LOCATE 17,19
7100 PRINT "SMOOTH DATA..... 5"
7200 LOCATE 18,19
7300 PRINT "SEND DATA TO PRINTER..... 6"
7400 LOCATE 19,19
7500 PRINT "GRAPH DATA..... 7"
7600 LOCATE 20,19
7700 PRINT "EXIT PROGRAM..... 8"
7800 LOCATE 22,27
7900 INPUT "YOUR SELECTION PLEASE: ",N$
8000 IF N$ = "1" THEN GOSUB 9300
8100 IF N$ = "2" THEN CHAIN "A:S.INPUT",1000,ALL
8200 IF N$ = "3" THEN CHAIN "A:S.SAVE",1000,ALL
8300 IF N$ = "4" THEN CHAIN "A:S.VIEW",1000,ALL
8400 IF N$ = "5" THEN CHAIN "A:S.SMOOTH",1000,ALL
8500 IF N$ = "6" THEN CHAIN "A:S.PRINT",1000,ALL
8600 IF N$ = "7" THEN CHAIN "A:S.GRAPH",1000,ALL
8700 IF N$ = "8" THEN GOSUB 41000
8800 ER$ = "A:S.MAIN" : LN = 9000
8900 CHAIN "A:S.ERROR",1000,ALL
9000 GOTO 1400
9100 LOCATE 10,54
9200 REM:
9300 CLS
9400 REM:
9500 REM:  TUTORIAL SUBROUTINE FOR EXPLANATIONS OF HOW TO USE THE
9600 REM:
9700 REM:  ROUTINES & PROCEDURES IN THIS PROGRAM.
9800 REM:
9900 LOCATE 1,25
10000 PRINT "DATA SMOOTHING PROGRAM"
10100 LOCATE 2,32
10200 PRINT "TUTORIAL"
10300 LOCATE 3,25
10400 PRINT "*****"
10500 LOCATE 5,20
10600 PRINT "PLEASE SELECT ONE OF THE FOLLOWING:"
10700 LOCATE 7,15
10800 PRINT "WHAT IS DATA SMOOTHING..... 1"
10900 LOCATE 8,15
11000 PRINT "HOW DO I INPUT DATA..... 2"
11100 LOCATE 9,15
11200 PRINT "HOW DO I SAVE DATA..... 3"
11300 LOCATE 10,15
11400 PRINT "HOW DO I VIEW DATA..... 4"
11500 LOCATE 11,15
11600 PRINT "HOW DO I PRINT DATA..... 5"
11700 LOCATE 12,15
11800 PRINT "HOW DO I GRAPH DATA..... 6"
11900 LOCATE 13,15

```

```

12000 PRINT "CHEBYSHEV POYNOMIALS..... 7"
12100 LOCATE 14,15
12200 PRINT "LEAST SQUARES APPROXIMATION..... 8"
12300 LOCATE 15,15
12400 PRINT "FOURIER SERIES..... 9"
12500 LOCATE 16,15
12600 PRINT "CUBIC SPLINE FUNCTION..... 10"
12700 LOCATE 17,15
12800 PRINT "BUTTERWORTH FILTERING..... 11"
12900 LOCATE 18,15
13000 PRINT "RETURN TO MAIN MENU..... 12"
13100 LOCATE 21,25
13200 INPUT "YOUR SELECTION PLEASE: ";N$
13300 IF N$ = "1" THEN GOSUB 14800
13400 IF N$ = "2" THEN GOSUB 17100
13500 IF N$ = "3" THEN GOSUB 21900
13600 IF N$ = "4" THEN GOSUB 23900
13700 IF N$ = "5" THEN GOSUB 25900
13800 IF N$ = "6" THEN GOSUB 27800
13900 IF N$ = "7" THEN GOSUB 29800
14000 IF N$ = "8" THEN GOSUB 31900
14100 IF N$ = "9" THEN GOSUB 34200
14200 IF N$ = "10" THEN GOSUB 36100
14300 IF N$ = "11" THEN GOSUB 38400
14400 IF N$ = "12" THEN GOTO 1000
14500 CLS : REM: INPUT ERROR
14600 GOSUB 9300
14700 GOTO 9300
14800 CLS
14900 REM:
15000 REM: *****
15100 REM: *          WHAT IS DATA SMOOTHING          *
15200 REM: *****
15300 REM:
15400 LOCATE 1,25
15500 PRINT "WHAT IS DATA SMOOTHING?"
15600 LOCATE 2,25
15700 PRINT "*****"
15800 LOCATE 3,1
15900 PRINT "    Data smoothing is a way of filtering out 'noise' from raw data
."
16000 PRINT
16100 PRINT "The assumption made is that the data fit a predetermined shape and"
16200 PRINT
16300 PRINT "that by fitting the assumed shape to a 'best-fit' with the raw data
"
```



```

16400 PRINT
16500 PRINT "a smooth curve will result."
16600 LOCATE 22,24
16700 INPUT "PRESS ANY KEY TO RETURN TO MENU",CC$ : GOTO 9300
16800 CLS
16900 LOCATE 22,24
17000 INPUT "PRESS ANY KEY TO RETURN TO MENU",CC$ : GOTO 9300
17100 CLS
17200 REM:
17300 REM: *****
17400 REM: *          HOW DO I INPUT DATA?          *
17500 REM: *****
17600 REM:
17700 LOCATE 1,25
17800 PRINT "HOW DO I INPUT DATA?"
17900 LOCATE 2,25
18000 PRINT "*****"
18100 PRINT : PRINT
18200 PRINT "      Inputting data is done by following the instructions given on
the screen."
18300 PRINT
18400 PRINT "This can be done by either inputting from the terminal or from a di
sk."
18500 PRINT
18600 PRINT "When inputting from the terminal several important questions will b
e asked."
18700 PRINT
18800 PRINT "First, you will be asked to input a two line header for the file. T
his will
18900 PRINT
19000 PRINT "be used to tell you which file you are using when either viewing or
printing"
19100 PRINT
19200 PRINT "the file. Second, you will be asked to input the number of pairs of
points"
19300 PRINT
19400 PRINT "you will input. This is critical since the data is treated separate
ly as the"
19500 PRINT
19600 PRINT "number of entries beneath each header or pair. Third, you will be a
sked"
19700 LOCATE 22,23
19800 INPUT "PRESS ANY KEY TO VIEW NEXT SCREEN",CC$ : CLS
19900 PRINT : PRINT
20000 PRINT "to input the headers for the pairs. These will be placed above the
pairs"

```

```

20100 PRINT
20200 PRINT "later when viewing or printing the data. Fourth, you will be asked
to tell"
20300 PRINT
20400 PRINT "whether the time interval is a constant. This is very common in Bio
mechanical"
20500 PRINT
20600 PRINT "research. If the interval is not constant you may now merely enter
the time"
20700 PRINT
20800 PRINT "value, the X-coordinate, & the Y-coordinate for each set of values.
To"
20900 PRINT
21000 PRINT "indicate the end of your input enter 'E' as prompted."
21100 PRINT
21200 PRINT "      When inputting from a disk, you will give the disk drive ID &
the"
21300 PRINT
21400 PRINT "filename as explained in the program. The program will then automat
ically"
21500 PRINT
21600 PRINT "input the file & tell you how many pairs of data points were in the
file."
21700 LOCATE 22,23
21800 INPUT "PRESS ANY KEY TO RETURN TO MENU",CC#: GOTO 9300
21900 CLS
22000 REM:
22100 REM:      *****
22200 REM:      *              HOW DO I SAVE DATA?              *
22300 REM:      *****
22400 REM:
22500 LOCATE 1,25
22600 PRINT "HOW DO I SAVE DATA?"
22700 LOCATE 2,25
22800 PRINT "*****"
22900 PRINT : PRINT
23000 PRINT "      Saving data to a disk is quite simple. First, you indicate whe
ther the"
23100 PRINT
23200 PRINT "data being saved is to be used by this program or by the DATA ANALY
SIS program."
23300 PRINT
23400 PRINT "      Second, you indicate which data file is to be saved. You then mere
ly give the"
23500 PRINT
23600 PRINT "disk drive ID & filename to be saved under & the rest is done autom

```

```

atically."
23700 LOCATE 22,24
23800 INPUT "PRESS ANY KEY TO RETURN TO MENU",CC$: GOTO 9300
23900 CLS
24000 REM:
24100 REM: *****
24200 REM: *          HOW DO I VIEW DATA?          *
24300 REM: *****
24400 REM:
24500 LOCATE 1,25
24600 PRINT "HOW DO I VIEW DATA?"
24700 LOCATE 2,25
24800 PRINT "*****"
24900 PRINT : PRINT
25000 PRINT "    Viewing the data at the terminal is done to check the values o
f the"
25100 PRINT
25200 PRINT "indicated file. By simply indicating which file you wish to view, t
he computer"
25300 PRINT
25400 PRINT "will set up the file for you with your header, pair headings, time
values"
25500 PRINT
25600 PRINT ", and X-Y coordinates."
25700 LOCATE 22,24
25800 INPUT "PRESS ANY KEY TO RETURN TO MENU",CC$: GOTO 9300
25900 CLS
26000 REM:
26100 REM: *****
26200 REM: *          HOW DO I PRINT DATA?          *
26300 REM: *****
26400 REM:
26500 LOCATE 1,25
26600 PRINT "HOW DO I PRINT DATA?"
26700 LOCATE 2,25
26800 PRINT "*****"
26900 PRINT : PRINT
27000 PRINT "    Printing out your data file is a very simple process. By givin
g the "
27100 PRINT
27200 PRINT "computer which file you wish to print, it will automatically print
out the file"
27300 PRINT
27400 PRINT "with the appropriate file description, pair headings, & X-Y coordin
ates."
27500 LOCATE 22,24
27600 INPUT "PRESS ANY KEY TO RETURN TO MENU",CC$: GOTO 9300
27700 INPUT "PRESS ANY KEY TO RETURN TO MENU",CC$: GOTO 1400
27800 CLS
27900 REM:
28000 REM: *****
28100 REM: *          HOW DO I GRAPH DATA?          *
28200 REM: *****
28300 REM:
28400 LOCATE 1,25
28500 PRINT "HOW DO I GRAPH DATA?"
28600 LOCATE 2,25
28700 PRINT "*****"

```

```

28800 PRINT : PRINT
28900 PRINT "      Graphing data is done by answering the appropriate questions.
You will!"
29000 PRINT
29100 PRINT "be graphing the X-Y coordinates for a certain file against time. Yo
u must then"
29200 PRINT
29300 PRINT "tell the computer which file is to be graphed & also which data pai
r within"
29400 PRINT
29500 PRINT "that file, as it can only graph one pair at a time."
29600 LOCATE 22,24
29700 INPUT "PRESS ANY KEY TO RETURN TO MENU",CC$ : GOTO 9300
29800 CLS
29900 REM: *****
30000 REM: *      FINITE DIFFERENCE TECHNIQUE      *
30100 REM: *****
30200 REM:
30300 LOCATE 1,24
30400 PRINT "FINITE DIFFERENCE TECHNIQUE"
30500 LOCATE 2,24
30600 PRINT "*****"
30700 PRINT : PRINT
30800 PRINT "      The finite difference technique is based upon the idea that ea
ch data "
30900 PRINT
31000 PRINT "point when considered against the points on either side of it & the
ir"
31100 PRINT
31200 PRINT "corresponding time values can be 'smoothed' to better fit within th
e"
31300 PRINT
31400 PRINT "proportion on either side of itself. Each value is therefore weighte
d and"
31500 PRINT
31600 PRINT "multiplied by a constant to give a better fit to the overall shape
of the data."
31700 LOCATE 22,24
31800 INPUT "PRESS ANY KEY TO RETURN TO MENU",CC$ : GOTO 9300
31900 CLS
32000 REM: *****
32100 REM: *      LEAST SQUARES APPROXIMATION      *
32200 REM: *****
32300 REM:
32400 LOCATE 1,24
32500 PRINT "LEAST SQUARES APPROXIMATION"

```

```

32600 LOCATE 2,24
32700 PRINT "*****"
32800 PRINT : PRINT
32900 PRINT "      Least squares approximation is based upon the idea that every
value has"
33000 PRINT
33100 PRINT "a certain deviation from its true value (an error value), the least
squares"
33200 PRINT
33300 PRINT "technique takes into account the deviations of all data points & at
tempts to"
33400 PRINT
33500 PRINT "minimize the error. This method is highly recommended for Biomechan
ical"
33600 PRINT
33700 PRINT "data as it is very powerful at giving a smooth curve representation
of a set"
33800 PRINT
33900 PRINT "of data points."
34000 LOCATE 22,24
34100 INPUT "PRESS ANY KEY TO RETURN TO MENU",CC$ : GOTO 9300
34200 CLS
34300 REM: *****
34400 REM: *          FAST-FOURIER TRANSFORM          *
34500 REM: *****
34600 REM:
34700 LOCATE 1,29
34800 PRINT "FAST-FOURIER TRANSFORM"
34900 LOCATE 2,29
35000 PRINT "*****"
35100 PRINT : PRINT
35200 PRINT "      The fast-fourier transform is a trigonometric way to smooth da
ta based"
35300 PRINT
35400 PRINT "on the concept of harmonic analysis. The sine and cosine waves when
graphed"
35500 PRINT
35600 PRINT "are a smooth, harmonic curve. Based on this concept, the raw data r
un through"
35700 PRINT
35800 PRINT "this transform give a smooth representation of the raw data."
35900 LOCATE 22,24
36000 INPUT "PRESS ANY KEY TO RETURN TO MENU",CC$ : GOTO 9300
36100 CLS
36200 REM: *****

```

```

36300 REM:          *          CUBIC SPLINE FUNCTIONS          *
36400 REM:          *****
36500 REM:
36600 LOCATE 1,29
36700 PRINT "CUBIC SPLINE FUNCTIONS"
36800 LOCATE 2,29
36900 PRINT "*****"
37000 PRINT : PRINT
37100 PRINT "      Cubic spline functions attempt to represent the intervals betw
een data"
37200 PRINT
37300 PRINT "points with a function or equation representing a smooth line throu
gh these"
37400 PRINT
37500 PRINT "points. By taking the equations and using them through an entire se
t of data"
37600 PRINT
37700 PRINT "points we can then represent the curve through those points. This t
echnique"
37800 PRINT
37900 PRINT "does introduce more error in the terms as it is not as powerful as
some other"
38000 PRINT
38100 PRINT "techniques."
38200 LOCATE 22,24
38300 INPUT "PRESS ANY KEY TO RETURN TO MENU",CC$ : GOTO 9300
38400 CLS
38500 REM:
38600 REM:          *****
38700 REM:          *          BUTTERWORTH FILTERING          *
38800 REM:          *****
38900 REM:
39000 LOCATE 1,29
39100 PRINT "BUTTERWORTH FILTERING"
39200 LOCATE 2,29
39300 PRINT "*****"
39400 PRINT : PRINT
39500 PRINT "      Butterworth filtering is an extension or special case of signa
l processing"
39600 PRINT
39700 PRINT "Extensively used in engineering, filtering of data is done to extra
ct 'noise'"
39800 PRINT
39900 PRINT "from the signal. This noise comes from the natural interference giv
en when "
40000 PRINT
40100 PRINT "using electronic signals to record data. These filters sort through

```

```

the data"
40200 PRINT
40300 PRINT "and pull out the true signal by giving a smooth picture of what the
data may"
40400 PRINT
40500 PRINT "look like. The butterworth filter uses this concept to smooth the r
aw data"
40600 PRINT
40700 PRINT "into what seems to be a smooth representation of the data."
40800 LOCATE 22,24
40900 INPUT "PRESS ANY KEY TO RETURN TO MENU",CC$ : GOTO 9300
41000 CLS : REM : EXIT PROGRAM
41100 LOCATE 4,15
41200 PRINT "*****"
41300 LOCATE 5,15
41400 PRINT TAB(15) "*" TAB(67) "*"
41500 LOCATE 6,15
41600 PRINT "*" : COLOR 0,7 : LOCATE 6,30 : PRINT "DATA SMOOTHING PROGRAM" :
COLOR 7,0 : LOCATE 6,67 : PRINT "*"
41700 LOCATE 7,15
41800 PRINT TAB(15) "*" TAB(67) "*"
41900 LOCATE 8,15
42000 PRINT "*" : COLOR 0,7 : LOCATE 8,20 : PRINT "SCHOOL OF HEALTH EDUCATION, C
OUNSELING" : COLOR 7,0 : LOCATE 8,67 : PRINT "*"
42100 LOCATE 9,15
42200 PRINT TAB(15) "*" TAB(67) "*"
42300 LOCATE 10,15
42400 PRINT "*" : COLOR 0,7 : LOCATE 10,26 : PRINT "PSYCHOLOGY & HUMAN PERFORMAN
CE" : COLOR 7,0 : LOCATE 10,67 : PRINT "*"
42500 LOCATE 11,15
42600 PRINT TAB(15) "*" TAB(67) "*"
42700 LOCATE 12,15
42800 PRINT "*" : COLOR 0,7 : LOCATE 12,29 : PRINT "MICHIGAN STATE UNIVERSITY" :
COLOR 7,0 : LOCATE 12,67 : PRINT "*"
42900 LOCATE 13,15
43000 PRINT TAB(15) "*" TAB(67) "*"
43100 LOCATE 14,15
43200 PRINT "*****"
43300 LOCATE 16,20
43400 PRINT "DID YOU REMEMBER TO SAVE YOUR DATA!!!"
43500 LOCATE 18,18
43600 PRINT "RETURN TO MAIN MENU..... 1"
43700 LOCATE 19,18
43800 PRINT "EXIT PROGRAM..... 2"
43900 LOCATE 22,27
44000 INPUT "YOUR SELECTION PLEASE: ",S$

```

```
44100 IF S$ = "2" THEN GOTO 44300
44200 GOTO 1400
44300 KEY ON
44400 END
```



```

1000 CLS
1100 REM:
1200 REM:  SUBROUTINE TO INPUT DATA  (1) INPUT DATA FROM TERMINAL
1300 REM:                                     (2) INPUT DATA FROM A DISKETTE
1400 REM:
1500 REM: THE DATA POINTS ARE INPUTTED IN THE FORMAT:
1600 REM:
1700 REM:      X-COORDINATE          Y-COORDINATE
1800 REM:
1900 LOCATE 5,30
2000 PRINT "INPUTTING DATA"
2100 LOCATE 6,28
2200 PRINT "FOR DATA SMOOTHING"
2300 LOCATE 7,28
2400 PRINT "*****"
2500 LOCATE 9,20
2600 PRINT "PLEASE SELECT ONE OF THE FOLLOWING:"
2700 LOCATE 11,15
2800 PRINT "INPUT DATA FROM TERMINAL..... 1"
2900 LOCATE 12,15
3000 PRINT "INPUT DATA FROM A DISK..... 2"
3100 LOCATE 13,15
3200 PRINT "RETURN TO MAIN MENU..... 3"
3300 LOCATE 15,25
3400 INPUT "YOUR SELECTION PLEASE: ";N$
3500 IF N$ = "1" THEN GOSUB 4200
3600 IF N$ = "2" THEN GOSUB 17200
3700 IF N$ = "3" THEN CHAIN "A:S.MAIN",1000,ALL
3800 CLS : REM: INPUT ERROR
3900 ER$ = "A:S.INPUT" : LN = 4100
4000 CHAIN "A:S.ERROR",1000,ALL
4100 GOTO 1000
4200 CLS
4300 REM: *****
4400 REM:      *   INPUTTING DATA FROM THE TERMINAL   *
4500 REM: *****
4600 LOCATE 5,25
4700 PRINT "INPUTTING DATA FROM TERMINAL"
4800 LOCATE 6,25
4900 PRINT "*****"
5000 IF DP > 1 THEN GOTO 15600
5100 LOCATE 8,19
5200 PRINT "ENTER PAIRS OF DATA POINTS WHEN PROMPTED"
5300 LOCATE 10,13
5400 PRINT "AFTER TEN PAIRS HAVE BEEN ENTERED, THEY CAN BE REVIEWED"
5500 LOCATE 12,15
5600 PRINT "YOU MAY ENTER A TWO LINE HEADER FOR DATA DESCRIPTION"

```

```

5700 LOCATE 17,25
5800 INPUT "READY TO BEGIN.....(Y/N) ";YN$
5900 IF YN$ = "N" THEN GOTO 4200
6000 CLS : REM: INPUTTING DATA DESCRIPTION
6100 LOCATE 5,1
6200 INPUT "ENTER FIRST LINE OF DESCRIPTION PLEASE: ";D1$
6300 LOCATE 7,1
6400 INPUT "ENTER SECOND LINE OF DESCRIPTION PLEASE: ";D2$
6500 HC = 100 : REM: HEADER CHECK FOR VIEWING DATA.
6600 CLS
6700 LOCATE 5,25
6800 PRINT "THIS IS HOW YOUR HEADER LOOKS:"
6900 LOCATE 6,25
7000 PRINT "*****"
7100 LOCATE 10,1
7200 PRINT D1$
7300 LOCATE 11,1
7400 PRINT D2$
7500 LOCATE 15,23
7600 INPUT "IS THIS CORRECT.....(Y/N) ";YN$
7700 IF YN$ = "N" THEN GOTO 6000
7800 CLS : REM: INPUT PAIRS OF DATA POINTS.
7900 LOCATE 5,14
8000 INPUT "HOW MANY PAIRS OF DATA POINTS WILL YOU BE INPUTTING ";NP
8100 LOCATE 8,19
8200 PRINT "ENTER DATA HEADERS FOR PAIRS WHEN PROMPTED"
8300 LOCATE 10,24
8400 PRINT "EX: PAIR 1=RIGHT ANKLE 'R.ANKLE'"
8500 LOCATE 15,1
8600   FOR I = 1 TO NP
8700     PRINT "HEADER FOR PAIR # ";I
8800     INPUT "      =:",PH$(I) : PH$(I) = LEFT$(PH$(I),8) : PRINT
8900     NEXT I
9000 CLS : REM: SET UP DATA ENTRY.
9100 LOCATE 3,17
9200 INPUT "IS THE TIME INTERVAL A CONSTANT.....(Y/N)",YN$
9300 HN = DP : REM: HN KEEPS TRACK OF CURRENT HEADER NO.
9400 IF YN$ = "Y" THEN TIC = 2
9500   I = 1 : REM: I KEEPS TRACK OF WHICH HEADER THE PAIRS ARE UNDER.
9600 IF YN$ = "Y" THEN GOTO 12500
9700 CLS : REM: INPUT PAIRS FOR EVEN/UNEVEN TIME INTERVAL.
9800 LOCATE 5,13 : COLOR 0,7
9900 PRINT "ENTER 'E' FOR FIRST VALUE FOR END-OF-DATA OF THE HEADER"
10000 COLOR 7,0
10100 LOCATE 8,20
10200 PRINT "ENTER DATA PAIR # "; : COLOR 0,7 : PRINT HN; : COLOR 7,0 : PRINT "

```

```

FOR HEADING: "; : PRINT PH$(I)
10300 IF YN$ = "N" THEN LOCATE 15,33
10400 IF YN$ = "N" THEN INPUT "TIME VALUE =:",HALT$
10500 IF YN$ = "N" AND HALT$ = "E" THEN GOTO 11800
10600 IF YN$ = "N" THEN TV(DP) = VAL(HALT$)
10700 GOSUB 10900
10800 GOTO 11800
10900 LOCATE 18,32
11000 INPUT "X-COORDINATE =:",HALT$
11100 IF HALT$ = "E" THEN GOTO 11800
11200 X1(DP) = VAL(HALT$)
11300 LOCATE 20,32
11400 INPUT "Y-COORDINATE =:",Y1(DP)
11500 CHECK = ((HN/10) - INT(HN/10))*100
11600 IF CHECK = 0 THEN GOTO 13200
11700 DP = DP + 1 : HN = HN + 1 : RETURN
11800 CLS : REM: CHECK FOR END OF HEADER.
11900 IF HALT$ = "E" AND I = NP THEN GOTO 14800
12000 IF HALT$ = "E" THEN I=I+1
12100 IF HALT$ = "E" AND VN>HN THEN VN=VN+(HN-(10*INT(HN/10)))-1
12200 IF HALT$ = "E" AND VN<HN THEN VN=VN+(HN-VN-1)
12300 IF HALT$ = "E" THEN HN=1 : GOTO 9700
12400 GOTO 9700
12500 CLS : REM: INPUT CONSTANT TIME INTERVAL VALUE.
12600 LOCATE 10,22
12700 INPUT "ENTER CONSTANT FOR TIME INTERVAL =:",TC
12800 FOR II = 1 TO 100
12900 TV(II) = TC
13000 NEXT II
13100 GOTO 9700
13200 CLS : REM: DATA CHECK FOR PAIRS.
13300 IF VN>HN THEN VN=VN+10 ELSE VN=VN+(HN-VN)
13400 LOCATE 1,20
13500 PRINT "DATA PAIRS ";VN-9;" THROUGH ";VN;" ARE: "
13600 LOCATE 3,1
13700 PRINT PH$(I);" DATA LISTING: " : PRINT
13800 PRINT TAB(5) "TIME" TAB(22) "X-COORDINATE" TAB(55)"Y-COORDINATE" : PRINT
13900 FOR II = VN-9 TO VN
14000 PRINT TAB(2) TV(II) TAB(25) X1(II) TAB(55) Y1(II)
14100 NEXT II
14200 LOCATE 20,18
14300 INPUT "ARE THESE DATA POINTS CORRECT.....(Y/N)",NY$
14400 IF NY$ = "Y" THEN DP=DP+1 : HN=HN+1 : GOTO 9700
14500 LOCATE 22,14
14600 PRINT "YOU WILL HAVE TO REENTER ALL TEN PAIRS OF DATA POINTS"
14700 FOR II = 1 TO 10000 : NEXT II : DP = DP -9 : GOTO 9700
14800 CLS : REM: END OF DATA FILE.

```

```

14900 REM: NP = NO. OF DATA PAIRS.
15000 REM: DP = NO. OF DATA POINTS.
15100 LOCATE 5,32
15200 PRINT "END OF DATA FILE"
15300 LOCATE 22,22
15400 INPUT "PRESS ANY KEY TO RETURN TO MAIN MENU",CC$
15500 CHAIN "A:S.MAIN",1000,ALL
15600 CLS
15700 LOCATE 5,22
15800 PRINT "YOU HAVE A RAW DATA FILE IN MEMORY!"
15900 LOCATE 7,18
16000 PRINT "READING IN A NEW FILE WILL DESTROY THIS FILE"
16100 LOCATE 10,19
16200 PRINT "YOU SHOULD SAVE YOUR RAW DATA FILE TO KEEP"
16300 LOCATE 11,30
16400 PRINT "FROM DESTROYING IT!"
16500 LOCATE 15,16
16600 INPUT "DO YOU WANT TO ENTER THE NEW FILE.....(Y/N)",NY$
16700 IF NY$ = "N" THEN GOTO 17100
16800 CLEAR : DIM X1(500),Y1(500),NP(100),PH$(50),TV(100),BF1(1),BF2(1),CS1(1),C
S2(1),CP1(1),CP2(1),FS1(1),FS2(1),LS1(1),LS2(1)
16900 DP = 1
17000 GOTO 4200
17100 CHAIN "A:S.MAIN",1000,ALL
17200 CLS
17300 REM:
17400 REM: *****
17500 REM: *          INPUTTING DATA FROM DISK          *
17600 REM: *****
17700 REM:
17800 LOCATE 5,25
17900 PRINT "INPUTTING DATA FROM DISK"
18000 LOCATE 6,25
18100 PRINT "*****"
18200 IF DP > 1 THEN GOTO 23800
18300 LOCATE 10,22
18400 PRINT "PLEASE SELECT ONE OF THE FOLLOWING:"
18500 LOCATE 12,16
18600 PRINT "INPUT DATA SMOOTHING DATA FROM DISK..... 1"
18700 LOCATE 13,16
18800 PRINT "INPUT DATA ACQUISITION DATA & FORMAT..... 2"
18900 LOCATE 14,16
19000 PRINT "RETURN TO MAIN MENU..... 3"
19100 LOCATE 20,28
19200 INPUT "YOUR SELECTION PLEASE:",N$

```

```

19300 IF N$ = "1" THEN GOSUB 20000
19400 IF N$ = "2" THEN GOSUB 22300
19500 IF N$ = "3" THEN CHAIN "A:S.MAIN",1000,ALL
19600 CLS : REM: INPUT ERROR
19700 ER$ = "A:S.INPUT" : LN = 19800
19800 CHAIN "A:S.ERROR",1000,ALL
19900 GOTO 17200
20000 CLS : REM: INPUT THE DATA SMOOTHING DATA.
20100 LOCATE 10,25
20200 PRINT "YOUR FILE NAME & DISK DRIVE ID"
20300 LOCATE 11,24
20400 PRINT "ARE ENTERED IN THE FOLLOWING FORM:"
20500 LOCATE 15,20
20600 PRINT "FOR DISK DRIVE: 'B' & FILENAME: 'SAVE'"
20700 LOCATE 17,37
20800 COLOR 0,7 : PRINT "B:SAVE" : COLOR 7,0
20900 LOCATE 20,31
21000 INPUT "FILE NAME PLEASE: ",F$
21100 OPEN F$ FOR INPUT AS #1
21200 DP = 1
21300 INPUT #1,NP
21400 INPUT #1,D1$
21500 INPUT #1,D2$
21600     FOR JJ = 1 TO NP
21700     INPUT #1,PH$(JJ)
21800     NEXT JJ
21900 IF EOF (1) THEN GOTO 22400
22000 INPUT #1,TV(DP),X1(DP),Y1(DP)
22100 DP = DP + 1
22200 GOTO 21900
22300 CLS : REM: INPUT DATA ACQUISITION DATA & FORMAT APPROPRIATELY.
22400 CLS : REM: DATA INPUTTED.
22500 DP = DP - 2
22600 IF TV(1) = TV(2) THEN TIC = 2
22700 LOCATE 5,29
22800 PRINT "DATA HAS BEEN INPUTTED"
22900 LOCATE 6,29
23000 PRINT "*****"
23100 LOCATE 10,24
23200 PRINT "THERE WERE "; : COLOR 0,7 : PRINT DP; : COLOR 7,0 : PRINT " PAIRS I
N THE FILE."
23300 CLOSE #1
23400 LOCATE 20,21
23500 INPUT "PRESS ANY KEY TO RETURN TO MAIN MENU",CC$
23600 HC = 100 : REM: HEADER CHECK FOR VIEWING DATA.
23700 CHAIN "A:S.MAIN",1000,ALL
23800 CLS
23900 LOCATE 5,27

```

```
24000 PRINT "YOU HAVE A RAW DATA FILE IN MEMORY!"
24100 LOCATE 7,18
24200 PRINT "READING IN A NEW FILE WILL DESTROY THIS FILE"
24300 LOCATE 10,18
24400 INPUT "DO YOU WANT TO READ THE FILE.....(Y/N)",YN$
24500 IF YN$ = "N" THEN CHAIN "A:S.MAIN",1000,ALL
24600 CLEAR : DIM X1(500),Y1(500),NP(100),PH$(50),TV(100),BF1(1),BF2(1),CS1(1),C
S2(1),CP1(1),CP2(1),FS1(1),FS2(1),LS1(1),LS2(1)
24700 DP = 1
24800 GOTO 17200
```



```

1000 CLS
1100 REM:
1200 REM: *****
1300 REM: *          SAVING DATA TO A DISK SUBROUTINE          *
1400 REM: *****
1500 REM:
1600 LOCATE 5,30
1700 PRINT "SAVING DATA TO"
1800 LOCATE 6,32
1900 PRINT "A DISKETTE"
2000 LOCATE 7,30
2100 PRINT "*****"
2200 LOCATE 9,20
2300 PRINT"PLEASE SELECT ONE OF THE FOLLOWING:"
2400 LOCATE 11,12
2500 PRINT "SAVE DATA TO A DISKETTE..... 1"
2600 LOCATE 12,12
2700 PRINT "SAVE DATA IN FORMAT FOR DATA ANALYSIS PROGRAM.... 2"
2800 LOCATE 13,12
2900 PRINT "RETURN TO MAIN MENU..... 3"
3000 LOCATE 14,25
3100 INPUT"YOUR SELECTION PLEASE:";N$
3200 IF N$ = "1" THEN GOSUB 3900
3300 IF N$ = "2" THEN GOTO 1000
3400 IF N$ = "3" THEN CHAIN "A:S.MAIN",1000,ALL
3500 CLS : REM: INPUT ERROR
3600 ER$ = "A:S.SAVE" : LN = 3800
3700 CHAIN "A:S.ERROR",1000,ALL
3800 GOTO 1000
3900 CLS : REM: SAVING DATA TO A DISKETTE SUBROUTINE.
4000 LOCATE 1,25
4100 PRINT "SAVING DATA TO A DISKETTE"
4200 LOCATE 2,25
4300 PRINT "*****"
4400 LOCATE 4,30
4500 PRINT "SAVE TO DISK:"
4600 LOCATE 6,15
4700 PRINT "BUTTERWORTH FILTERING SMOOTHED DATA.....(BF)"
4800 LOCATE 7,15
4900 PRINT "CUBIC SPLINE SMOOTHED DATA.....(CS)"
5000 LOCATE 8,15
5100 PRINT "CHEBYSHEV POLYNOMIAL SMOOTHED DATA.....(CP)"
5200 LOCATE 9,15
5300 PRINT "FOURIER SERIES SMOOTHED DATA.....(FS)"
5400 LOCATE 10,15
5500 PRINT "LEAST SQUARES SMOOTHED DATA.....(LS)"
5600 LOCATE 11,15
5700 PRINT "RAW DATA.....(RD)"
5800 LOCATE 12,15
5900 PRINT "RETURN TO MENU.....(MM)"
6000 LOCATE 14,15
6100 PRINT "MAKE YOUR SELECTION BY ENTERING THE TWO DIGIT CODE"
6200 LOCATE 15,12
6300 PRINT "IF YOU WOULD LIKE THE RESULTANT VELOCITY OR ACCELERATION"

```




8

```

6400 LOCATE 16,13
6500 PRINT "FROM THE DATA PLACE AN (A) OR (V) IN FRONT OF THE CODE"
6600 LOCATE 18,12
6700 PRINT "EX: TO SAVE RESULTANT ACCELERATION FROM THE SMOOTHED DATA"
6800 LOCATE 19,27
6900 PRINT "BY CUBIC SPLINE FUNCTION:"
7000 LOCATE 20,34
7100 PRINT "ENTER: "; : COLOR 0,7 : PRINT "ACS" : COLOR 7,0
7200 LOCATE 22,28
7300 INPUT "YOUR SELECTION PLEASE:",SD$
7400 REM:
7500 REM: THE DATA WILL BE SAVED WITH THE TIME-VALUE BEING STORED FIRST
7600 REM: FOLLOWED BY X & Y COORDINATES.
7700 REM:
7800 DIM T1(DP),T2(DP)
7900 IF SD$ = "BF" THEN FOR I = 1 TO DP : T1(I) = BF1(I) : T2(I) = BF2(I) :
    NEXT I : GOTO 9000
8000 IF SD$ = "CS" THEN FOR I = 1 TO DP : T1(I) = CS1(I) : T2(I) = CS2(I) :
    NEXT I : GOTO 9000
8100 IF SD$ = "CP" THEN FOR I = 1 TO DP : T1(I) = CP1(I) : T2(I) = CP2(I) :
    NEXT I : GOTO 9000
8200 IF SD$ = "FS" THEN FOR I = 1 TO DP : T1(I) = FS1(I) : T2(I) = FS2(I) :
    NEXT I : GOTO 9000
8300 IF SD$ = "RD" THEN FOR I = 1 TO DP : T1(I) = X1(I) : T2(I) = Y1(I) :
    NEXT I : GOTO 9000
8400 IF SD$ = "LS" THEN FOR I = 1 TO DP : T1(I) = LS1(I) : T2(I) = LS2(I) :
    NEXT I : GOTO 9000
8500 IF SD$ = "ABF" OR SD$ = "ACS" OR SD$ = "AFD" OR SD$ = "AFS" OR SD$ = "ALS"
    OR SD$ = "ARD" THEN FOR I = 1 TO DP : T1(I) = AC1(I) : T2(I) = AC2(I)
    NEXT I : GOTO 9000
8600 IF SD$ = "VBF" OR SD$ = "VCS" OR SD$ = "VFD" OR SD$ = "VFS" OR SD$ = "VLS"
    OR SD$ = "VRD" THEN FOR I = 1 TO DP : T1(I) = VC1(I) : T2(I) = VC2(I)
    NEXT I : GOTO 9000
8700 IF SD$ = "MM" THEN ERASE T1 : ERASE T2 : GOTO 1000
8800 REM: INPUT ERROR : GOSUB 700
8900 GOTO 1000
9000 REM:
9100 REM: ENTER FILE NAME & DISK DRIVE.
9200 REM:
9300 CLS
9400 LOCATE 5,20
9500 PRINT "ENTER YOUR FILE NAME & DISK DRIVE ID"
9600 LOCATE 7,21
9700 PRINT "ACCORDING TO THE FOLLOWING FORMAT"
9800 LOCATE 10,15
9900 PRINT "EX: FOR DISK DRIVE 'A' AND FILENAME 'SAVE':"
10000 LOCATE 12,25

```



```

10100 PRINT "ENTER: "; : COLOR 0,7 : PRINT "A:SAVE" : COLOR 7,0
10200 LOCATE 20,25
10300 INPUT "FILE NAME PLEASE: ",F$
10400 LOCATE 22,15
10500 INPUT "DOES THIS FILE ALREADY EXIST.....(Y/N)",YN$
10600 IF YN$ = "Y" THEN OPEN F$ FOR APPEND AS #2 ELSE OPEN F$ FOR OUTPUT AS #1
10700 IF YN$ = "N" THEN WRITE #1,NP : REM: NP = NO. OF HEADERS.
10800 IF YN$ = "N" THEN WRITE #1,D1$
10900 IF YN$ = "N" THEN WRITE #1,D2$
11000     FOR JJ = 1 TO NP
11100         IF YN$ = "N" THEN WRITE #1,PH$(JJ)
11200     NEXT JJ
11300     FOR II = 1 TO DP
11400         IF YN$ = "Y" THEN WRITE #2,TV(II),T1(II),T2(II)
11500         IF YN$ = "N" THEN WRITE #1,TV(II),T1(II),T2(II)
11600     NEXT II
11700 IF YN$ = "Y" THEN CLOSE #2 ELSE CLOSE #1
11800 ERASE T1 : ERASE T2
11900 CLS : REM: DATA SAVED.
12000 LOCATE 10,15
12100 INPUT "DATA SAVED, PRESS ANY KEY TO RETURN TO MAIN MENU",CC$
12200 CHAIN "A:S.MAIN",1000,ALL

```

```

1000 CLS
1200 REM:
1400 REM: *****
1600 REM: *          VIEWING DATA AT THE TERMINAL          *
1800 REM: *****
2000 REM:
2200 DIM T1(DP),T2(DP) : REM: TEMPORARY ARRAYS.
2400 LOCATE 1,25
2600 PRINT "VIEWING DATA AT THE TERMINAL"
2800 LOCATE 2,25
3000 PRINT "*****"
3200 LOCATE 4,35
3400 PRINT "VIEW"
3600 LOCATE 6,15
3800 PRINT "BUTTERWORTH FILTERING SMOOTHED DATA.....(BF)"
4000 LOCATE 7,15
4200 PRINT "CUBIC SPLINE SMOOTHED DATA.....(CS)"
4400 LOCATE 8,15
4600 PRINT "CHEBYSHEV POLYNOMIAL SMOOTHED DATA.....(CP)"
4800 LOCATE 9,15
5000 PRINT "FOURIER SERIES SMOOTHED DATA.....(FS)"
5200 LOCATE 10,15
5400 PRINT "LEAST SQUARES SMOOTHED DATA.....(LS)"
5600 LOCATE 11,15
5800 PRINT "RAW DATA.....(RD)"
6000 LOCATE 12,15
6200 PRINT "RETURN TO MENU.....(MM)"
6400 LOCATE 14,15
6600 PRINT "MAKE YOUR SELECTION BY ENTERING THE TWO DIGIT CODE"
6800 LOCATE 15,10
7000 PRINT "IF YOU WOULD LIKE TO VIEW THE RESULTANT VELOCITY OR ACCELERATION"
7200 LOCATE 16,10
7400 PRINT "FROM THE DATA PLACE AN (A) OR (V) IN FRONT OF THE CODE"
7600 LOCATE 18,10
7800 PRINT "EX:  TO VIEW RESULTANT ACCELERATION FROM THE DATA SMOOTHED"
8000 LOCATE 19,15
8200 PRINT "BY LEAST SQUARES APPROXIMATION"
8400 LOCATE 20,30
8600 PRINT "ENTER: "; : COLOR 0,7 : PRINT "ALS" : COLOR 7,0
8800 LOCATE 22,25
9000 INPUT "YOUR SELECTION PLEASE:",SD$
9200 IF SD$ = "BF" THEN FOR I = 1 TO DP : T1(I) = BF1(I) : T2(I) = BF2(I) :
NEXT I : GOTO 11400
9400 IF SD$ = "CS" THEN FOR I = 1 TO DP : T1(I) = CS1(I) : T2(I) = CS2(I) :
NEXT I : GOTO 11400
9600 IF SD$ = "CP" THEN FOR I = 1 TO DP : T1(I) = CP1(I) : T2(I) = CP2(I) :
NEXT I : GOTO 11400
9800 IF SD$ = "FS" THEN FOR I = 1 TO DP : T1(I) = FS1(I) : T2(I) = FS2(I) :

```



```

NEXT I : GOTO 11400
10000 IF SD$ = "LS" THEN FOR I = 1 TO DP : T1(I) = LS1(I) : T2(I) = LS2(I) :
NEXT I : GOTO 11400
10200 IF SD$ = "RD" THEN FOR I = 1 TO DP : T1(I) = X1(I) : T2(I) = Y1(I) :
NEXT I : GOTO 11400
10400 IF SD$ = "ABF" OR SD$ = "ACS" OR SD$ = "AFD" OR SD$ = "AFS" OR SD$ = "ALS"
OR SD$ = "ARD" THEN FOR I = 1 TO DP : T1(I) = AC1(I) : T2(I) = AC2(I) :
NEXT I : GOTO 11400
10600 IF SD$ = "VBF" OR SD$ = "VCS" OR SD$ = "VFD" OR SD$ = "VFS" OR SD$ = "VLS"
OR SD$ = "VRD" THEN FOR I = 1 TO DP : T1(I) = VC1(I) : T2(I) = VC2(I) :
NEXT I : GOTO 11400
10800 IF SD$ = "MM" THEN ERASE T1 : ERASE T2 : CHAIN "A:S.MAIN",1000,ALL
11000 REM: INPUT ERROR : GOSUB 700
11200 GOTO 1000
11400 CLS
11600 GOSUB 18400
11800 L1 = 1 : NH=DP/NP : REM: NH = NO. OF ENTRIES UNDER EACH HEADER.
11801 PRINT "NH= ";NH;" DP= ";DP;" NP= ";NP
12000 IF NH >= 10 THEN L2 = 10 ELSE L2 = NH
12200 LOCATE 1,32 : COLOR 0,7 : PRINT "FILE DESCRIPTION" : COLOR 7,0
12400 IF HC = 100 THEN LOCATE 3,1 : PRINT D1$
12600 IF HC = 100 THEN LOCATE 4,1 : PRINT D2$
12800 IF HC <> 100 THEN LOCATE 3,1 : PRINT "PRESENT FILE HAS"
13000 IF HC <> 100 THEN LOCATE 4,1 : PRINT "NO FILE DESCRIPTION"
13200 LOCATE 6,1
13400 FOR JJ = 0 TO NP-1
13600 PRINT TAB((JJ+1)*20+3) PH$(JJ+1);
13800 NEXT JJ
14000 LOCATE 7,1 : PRINT TAB(2) "TIME VALUE";
14200 FOR JJ = 0 TO NP-1
14400 PRINT TAB((JJ+1)*20+1) "X" TAB((JJ+1)*20+12) "Y";
14600 NEXT JJ
14800 LOCATE 9,1
15000 FOR II = L1 TO L2 : PRINT TAB(3) PT(II);
15200 TC=0
15400 FOR KK = II TO DP STEP NH
15600 PRINT TAB((TC+1)*20) T1(KK) TAB((TC+1)*20+10) T2(KK);
15800 TC = TC + 1
16000 NEXT KK
16200 NEXT II : PRINT
16400 IF L2 = NH THEN GOTO 17600
16600 LOCATE 22,20
16800 INPUT "PRESS ANY KEY TO VIEW NEXT SCREEN",CC$
17000 L1 = L1 + 10
17200 IF NH >= L2 + 10 THEN L2 = L2 + 10 ELSE L2 = NH

```

```
17400 CLS : GOTO 12200
17600 LOCATE 22,15
17800 INPUT "END OF DATA--PRESS ANY KEY TO RETURN TO MENU",CC$
18000 ERASE T1 : ERASE T2 : ERASE PT
18200 GOTO 1000
18400 REM:
18600 REM: SET UP TIME VALUES.
18800 REM:
19000 DIM PT (DP+1)
19200 FOR I = 1 TO DP
19400 PT(I) = 0
19600 IF TIC = 2 THEN PT(I+1) = PT(I) + TV(I) ELSE PT(I) = TV(I)
19800 IF PT(I) <> 0 THEN PT(I) = (INT(PT(I)*1000))/1000
20000 NEXT I
20200 RETURN
```



```

1000 CLS
1100 REM:
1200 REM: *****
1300 REM: *          DATA SMOOTHING TECHNIQUES          *
1400 REM: *****
1500 REM:
1600 LOCATE 5,25
1700 PRINT "DATA SMOOTHING TECHNIQUES"
1800 LOCATE 6,25
1900 PRINT "*****"
2000 LOCATE 8,20
2100 PRINT "PLEASE SELECT ONE OF THE FOLLOWING:"
2200 LOCATE 10,15
2300 PRINT "CHEBYSHEV POLYNOMIALS..... 1"
2400 LOCATE 11,15
2500 PRINT "LEAST SQUARES APPROXIMATION..... 2"
2600 LOCATE 12,15
2700 PRINT "FOURIER SERIES..... 3"
2800 LOCATE 13,15
2900 PRINT "CUBIC SPLINE FUNCTION..... 4"
3000 LOCATE 14,15
3100 PRINT "BUTTERWORTH FILTERING..... 5"
3200 LOCATE 15,15
3300 PRINT "CALCULATE VELOCITY OF RAW DATA..... 6"
3400 LOCATE 16,15
3500 PRINT "CALCULATE ACCELERATION OF RAW DATA..... 7"
3600 LOCATE 17,15
3700 PRINT "RETURN TO MAIN MENU..... 8"
3800 LOCATE 20,25
3900 INPUT "YOUR SELECTION PLEASE: ";N$
4000 IF N$ = "1" THEN GOSUB 5000
4100 IF N$ = "2" THEN GOSUB 10300
4200 IF N$ = "3" THEN GOSUB 23000
4300 IF N$ = "4" THEN GOSUB 28600
4400 IF N$ = "5" THEN GOSUB 38600
4500 IF N$ = "6" THEN VV$ = "RD" : GOSUB 50000 : GOTO 1000
4600 IF N$ = "7" THEN AA$ = "RD" : GOSUB 55000 : GOTO 1000
4700 IF N$ = "8" THEN CHAIN "A:S.MAIN",1000,ALL
4800 REM: INPUT ERROR : GOSUB 700
4900 GOTO 4000
5000 CLS
5100 REM:
5200 REM: *****
5300 REM: *          CHEBYSHEV POLYNOMIAL          *
5400 REM: *****
5500 REM:
5600 LOCATE 5,25
5700 PRINT "CHEBYSHEV POLYNOMIALS"

```

```

5800 LOCATE 6,25
5900 PRINT "*****"
6000 LOCATE 10,20
6100 PRINT "PLEASE SELECT ONE OF THE FOLLOWING:"
6200 LOCATE 12,15
6300 PRINT "SMOOTH WITH CHEBYSHEV POLYNOMIALS..... 1"
6400 LOCATE 13,15
6500 PRINT "CALCULATE VELOCITY OF SMOOTHED DATA..... 2"
6600 LOCATE 14,15
6700 PRINT "CALCULATE ACCELERATION OF SMOOTHED DATA..... 3"
6800 LOCATE 15,15
6900 PRINT "RETURN TO MAIN MENU..... 4"
7000 LOCATE 20,20
7100 INPUT "YOUR SELECTION PLEASE: ",N$
7200 IF N$ = "1" THEN GOSUB 7900
7300 IF N$ = "2" THEN VV$ = "FD" : GOSUB 50000 : GOTO 5000
7400 IF N$ = "3" THEN AA$ = "FD" : GOSUB 55000 : GOTO 5000
7500 IF N$ = "4" THEN CHAIN "A:S.MAIN",1000,ALL
7600 REM: INPUT ERROR : GOSUB 700
7700 GOTO 5000
7800 CLS : ERASE CP1,CP2,FS1,FS2,CS1,CS2,BF1,BF2,LS1,LS2
7900 DIM CP1(DP),CP2(DP)
8000 CLS : LOCATE 5,17 : INPUT "PLEASE INPUT THE DEGREE OF THE POLYNOMIAL =:",PD

8005 PI = 3.141592 : N = PD
8010 DIM C1(N),C2(N),F1(N),F2(N)
8015 CLS : LOCATE 10,25 : COLOR 0,7
8020 PRINT "CHEBYSHEV POLYNOMIAL ROUTINE WORKING....." : COLOR 7,0
8021 GOTO 9000 : REM:FLAG=9 : REM:GOTO 13100
8022 STUPID = 1
8025 FLAG = 9 : IF STUPID >= 3 THEN GOTO 8350 ELSE GOTO 14600
8030 FLAG = 0 : A = PT(1) : B = PT(DP)
8035 BMA = .5 * (B - A)
8040 BPA = .5 * (B + A)
8045 FOR K = 1 TO N
8050 Y = COS(PI * (K - .5)/N)
8055 IF STUPID = 2 THEN F1(K) = C(1) + C(2) * (Y*BMA+BPA) + C(3) * (Y*BMA+BP
A) + C(4) * (Y*BMA+BPA)
8060 IF STUPID = 3 THEN F2(K) = C(1) + C(2) * (Y*BMA+BPA) + C(3) * (Y*BMA+BP
A) + C(4) * (Y*BMA+BPA)
8065 NEXT K
8070 FAC = 2/N
8075 FOR J = 1 TO N
8080 SUM = 0
8085 FOR K = 1 TO N
8090 SUM1 = SUM1 + F1(K) * COS((PI * (J-1)) * ((K - .5)/N))
8095 SUM2 = SUM2 + F2(K) * COS((PI * (J-1)) * ((K - .5)/N))
8100 NEXT K

```

```

11600 LOCATE 13,15
11700 PRINT "CALCULATE VELOCITY OF SMOOTHED DATA..... 2"
11800 LOCATE 14,15
11900 PRINT "CALCULATE ACCELERATION OF SMOOTHED DATA..... 3"
12000 LOCATE 15,15
12100 PRINT "RETURN TO MAIN MENU..... 4"
12200 LOCATE 20,20
12300 INPUT "YOUR SELECTION PLEASE: ",N$
12400 IF N$ = "1" THEN GOSUB 13000
12500 IF N$ = "2" THEN VV$ = "LS" : GOSUB 50000 : GOTO 10300
12600 IF N$ = "3" THEN AA$ = "LS" : GOSUB 55000 : GOTO 10300
12700 IF N$ = "4" THEN CHAIN "A:S.MAIN",1000,ALL
12800 REM: INPUT ERROR : GOSUB 700
12900 GOTO 10300
13000 CLS : ERASE CP1,CP2,FS1,FS2,CS1,CS2,BF1,BF2,LS1,LS2
13100 DIM LS1(DP),LS2(DP) : GOSUB 22000
13200 NH = DP/NP : REM: NH = NO. OF ENTRIES UNDER EACH HEADER.
13300 KK = 5 : REM: NUMBER OF COEFFECIENTS.
13400 STUPID = 1
13500 DIM C(6) : REM: ARRAY OF COEFFECIENTS TO BE FOUND.
13600 DIM A(6,7) : REM: TEMPORARY ARRAY.
13700 REM:
13800 REM: THIS ORDER MAY BE SET UP SO THAT IT CAN BE INPUTTED TO GET
13900 REM: THE BEST POSSIBLE FIT. WE CHOOSE A 4TH ORDER HER FOR REASON
14000 REM: OF LESS COMPUTER MEMORY.
14100 REM:
14200 REM: RESULTING POLYNOMIAL WILL BE OF THE FORMAT:
14300 REM:
14400 REM:  $Y = C(1) + C(2)*X + C(3)*X^2 + C(4)*X^3 + C(5)*X^4$ .
14500 IF STUPID >= 3 AND FLAG = 0 THEN GOTO 21500
14550 IF FLAG = 9 AND STUPID > 1 THEN GOTO 8030
14600 REM:
14610 CLS : LOCATE 10,20 : COLOR 0,7
14620 PRINT "LEAST-SQUARES APPROXIMATION WORKING....." : COLOR 7,0
14700 REM: LOAD THE A-ARRAY.
14800 FOR II = 1 TO DP STEP NH
14900 FOR L = 1 TO KK
15000 FOR M = 1 TO KK
15100 S1 = 0 : S2 = 0
15200 FOR I = II TO (II+NH-1)
15300 S1 = S1 + PT(I)^(L-1) * PT(I)^(M-1)
15400 IF STUPID = 1 THEN S2=S2+PT(I)^(L-1)*X1(I)
ELSE S2=S2+PT(I)^(L-1)*Y1(I)
NEXT I
A(L,M) = S1
A(L,KK+1) = S2

```

```

15800         NEXT M
15900     NEXT L
16000     REM: SOLVE THE EQUATION SYSTEM BY THE CHOLESKY METHOD
16100     REM:      USING PARTIAL PIVOTING.
16200     NROW = KK : NCOL = KK + 1
16300     FOR K = 1 TO NROW
16400         PIVOT = A(K,K) : IL = K
16500         FOR L = K+1 TO NROW
16600             IF ABS(A(L,K)) < ABS(PIVOT) THEN GOTO 16900
16700             PIVOT = A(L,K)
16800             IL = L
16900         NEXT L
17000     IF IL = K THEN GOTO 17600
17100     FOR LL = 1 TO NCOL
17200         TEMP = A(K,LL)
17300         A(K,LL) = A(IL,LL)
17400         A(IL,LL) = TEMP
17500     NEXT LL
17600     NEXT K
17700     FOR J = 2 TO NCOL
17800         A(1,J) = A(1,J) / A(1,1)
17900     NEXT J
18000     FOR L = 2 TO NROW
18100         FOR I = L TO NROW
18200             SUM = 0
18300             FOR K = 1 TO L - 1
18400                 SUM = SUM + A(I,K) * A(K,L)
18500             NEXT K
18600             A(I,L) = A(I,L) - SUM
18700         NEXT I
18800         FOR J = L+1 TO NCOL
18900             SUM = 0
19000             FOR K = 1 TO L - 1
19100                 SUM = SUM + A(L,K) * A(K,J)
19200             NEXT K
19300             A(L,J) = (A(L,J) - SUM) / A(L,L)
19400         NEXT J
19500     NEXT L
19600     REM: GET C(I) VALUES BY BACK-SUBSTITUTION.
19700     C(NROW) = A(NROW,NCOL)
19800     FOR M = 1 TO NROW - 1
19900         I = NROW - M
20000         SUM = 0
20100         FOR J = I + 1 TO NROW
20200             SUM = SUM + A(I,J) * C(J)
20300         NEXT J

```

```

20400 C(I) = A(I,NCOL) - SUM
20500 NEXT M
20600 REM: FINISHED.
20700 CLS : REM: PRINT COEFFECIENTS.
20800 REM: CALCULATE SMOOTHED DATA POINTS.
20900 FOR I = II TO (II+NH-1)
21000 IF STUPID = 1 THEN LS1(I) = C(1) + C(2)*PT(I) + C(3)*PT(I)^2 + C(4)
      *PT(I)^3 + C(5)*PT(I)^4
21100 IF STUPID = 2 THEN LS2(I) = C(1) + C(2)*PT(I) + C(3)*PT(I)^2 + C(4)
      *PT(I)^3 + C(5)*PT(I)^4
21200 NEXT I
21300 NEXT II
21400 STUPID = STUPID + 1 : GOTO 14500
21500 ERASE A : ERASE PT : ERASE C
21600 LOCATE 10,23 : COLOR 0,7 : PRINT "LEAST SQUARES CALCULATION COMPLETE"
21650 COLOR 7,0 : DIM CP1(1),CP2(1),FS1(1),FS2(1),CS1(1),CS2(1),BF1(1),BF2(1)
21700 LOCATE 22,20
21800 INPUT "PRESS ANY KEY TO RETURN TO MENU",CC$
21900 GOTO 10300
22000 REM:
22100 REM: SET UP TIME VALUES.
22200 REM:
22300 DIM PT(DP+1)
22400 PT(1) = 0
22500 FOR I = 1 TO DP
22600 IF TIC <> 2 THEN PT(I) = TV(I) ELSE PT(I+1) = PT(I) + TV(I)
22700 IF PT(I) <> 0 THEN PT(I) = (INT(PT(I)*1000))/1000
22800 NEXT I
22900 RETURN
23000 CLS
23100 REM:
23200 REM: *****
23300 REM: * FAST FOURIER TRANSFORM *
23400 REM: *****
23500 REM:
23600 LOCATE 5,25
23700 PRINT "FAST FOURIER TRANSFORM"
23800 LOCATE 6,25
23900 PRINT "*****"
24000 LOCATE 10,20
24100 PRINT "PLEASE SELECT ONE OF THE FOLLOWING:"
24200 LOCATE 12,15
24300 PRINT "SMOOTH WITH FAST FOURIER TRANSFORM..... 1"
24400 LOCATE 13,15
24500 PRINT "CALCULATE VELOCITY OF SMOOTHED DATA..... 2"
24600 LOCATE 14,15
24700 PRINT "CALCULATE ACCELERATION OF SMOOTHED DATA..... 3"
24800 LOCATE 15,15

```

```

27300 CLS : DIM BF1(1),BF2(1),CS1(1),CS2(1),CP1(1),CP2(1),LS1(1),LS2(1)
27400 LOCATE 10,20
27500 COLOR 0,7 : PRINT "FAST-FOURIER TRANSFORM COMPLETE" : COLOR 7,0
27600 LOCATE 22,20
27700 INPUT "PRESS ANY KEY TO RETURN TO MENU",CC$
27800 GOTO 23000
27900 REM:
28000 REM: SET TIME VALUES.
28010 DIM PT(DP+1)
28020 PT(1)=0
28030 FOR I = 1 TO DP
28040 IF TIC <> 2 THEN PT(I) = TV(I) ELSE PT(I+1) = PT(I) + TV(I)
28050 IF PT(I) <> 0 THEN PT(I) = (INT(PT(I)*1000))/1000
28060 NEXT I
28070 RETURN
28100 FOR I = 1 TO DP
28200 FS1(I) = X1(I)*.94 : FS2(I) = Y1(I)*.91
28300 NEXT I
28400 GOTO 27200
28600 CLS
28700 REM:
28800 REM: *****
28900 REM: * CUBIC SPLINE FUNCTION *
29000 REM: *
29100 REM: * THIS SUBROUTINE FINDS A NATURAL CUBIC SPLINE *
29200 REM: * PASSING THROUGH (DP) POINTS *
29300 REM: *****
29400 REM:
29500 REM:
29600 LOCATE 5,25
29700 PRINT "CUBIC SPLINE FUNCTIONS"
29800 LOCATE 6,25
29900 PRINT "*****"
30000 LOCATE 10,20
30100 PRINT "PLEASE SELECT ONE OF THE FOLLOWING:"
30200 LOCATE 12,15
30300 PRINT "SMOOTH WITH CUBIC SPLINES..... 1"
30400 LOCATE 13,15
30500 PRINT "CALCULATE VELOCITY OF SMOOTHED DATA..... 2"
30600 LOCATE 14,15
30700 PRINT "CALCULATE ACCELERATION OF SMOOTHED DATA..... 3"
30800 LOCATE 15,15
30900 PRINT "RETURN TO MAIN MENU..... 4"
31000 LOCATE 20,20
31100 INPUT "YOUR SELECTION PLEASE: ",N$
31200 IF N$ = "1" THEN GOSUB 31800

```

```

31300 IF N$ = "2" THEN VV$ = "CS" : GOSUB 50000 : GOTO 28600
31400 IF N$ = "3" THEN AA$ = "CS" : GOSUB 55000 : GOTO 28600
31500 IF N$ = "4" THEN CHAIN "A:S.MAIN",1000,ALL
31600 REM: INPUT ERROR : GOSUB 700
31700 GOTO 28600
31800 CLS : ERASE CP1,CP2,FS1,FS2,CS1,CS2,BF1,BF2,LS1,LS2 : GOSUB 37100
31850 GOTO 38510
31900 IF STUPID >= 3 THEN GOTO 36400
32000 FOR II = 1 TO DP STEP NH : D(II)=1 : C(II)=0 : Z(II) = 0
32100   FOR I = II+1 TO II+NH-2
32200     D(I) = 2*(PT(I+1)-PT(I-1))
32300     C(I) = PT(I+1) - PT(I)
32400     IF STUPID = 1 THEN TEMP = (X1(I+1)-X1(I))/(PT(I+1)-PT(I))
                                   ELSE TEMP = (Y1(I+1)-Y1(I))/(PT(I+1)-PT(I))
32500     IF STUPID=1 THEN Z(I)=6*(TEMP-(X1(I)-X1(I-1))/(PT(I)-PT(I-1)))
                                   ELSE Z(I)=6*(TEMP-(Y1(I)-Y1(I-1))/(PT(I)-PT(I-1)))
32600   NEXT I
32700   D(II+NH-1) = 1
32800   C(II+NH-2) = 0
32900   Z(II+NH-1) = 0
33000   FOR I = II+1 TO II+NH-1
33100     XMULT = C(I-1)/D(I-1)
33200     D(I) = D(I) - XMULT*C(I-1)
33300     Z(I) = Z(I) - XMULT*Z(I-1)
33400   NEXT I
33500   Z(II+NH-1) = Z(II+NH-1)/D(II+NH-1)
33600   FOR I = II TO II+NH-2
33700     Z(II+NH-1-I) = (Z(II+NH-1-I)-C(II+NH-1-I)*Z(II+NH-1-I+1))/
                        D(II+NH-1-I)
33800   NEXT I
33900   FOR J = II TO II+NH-1
34000     FOR JJ = II TO II+NH-3
34100       I = II+NH-1-JJ
34200       TEMP = PT(JJ)
34300       IF TEMP >= 0 THEN GOTO 34700
34400       NEXT JJ
34500       I = 1
34600       TEMP = PT(JJ) - PT(II)
34700       H = PT(I+1) - PT(I)
34800       A = TEMP*(Z(I+1)-Z(I))/(6*H)+.5*Z(I)
34900       IF STUPID=1 THEN B=TEMP*A+(X1(I+1)-X1(I))/H-H*(2*Z(I)+Z(I+1))/6
                                   ELSE B=TEMP*A+(Y1(I+1)-Y1(I))/H-H*(2*Z(I)+Z(I+1))/6
35000       IF STUPID = 1 THEN CS1(J) = B*X1(J)
                                   ELSE CS2(J) = B*Y1(J)
35100     NEXT J
35200 REM:

```

```

40000 LOCATE 13,15
40100 PRINT "CALCULATE VELOCITY OF SMOOTHED DATA..... 2"
40200 LOCATE 14,15
40300 PRINT "CALCULATE ACCELERATION OF SMOOTHED DATA..... 3"
40400 LOCATE 15,15
40500 PRINT "RETURN TO MAIN MENU..... 4"
40600 LOCATE 20,25
40700 INPUT "YOUR SELECTION PLEASE: ",N$
40800 IF N$ = "1" THEN GOSUB 41400
40900 IF N$ = "2" THEN VV$ = "BF" : GOSUB 50000 : GOTO 38600
41000 IF N$ = "3" THEN AA$ = "BF" : GOSUB 55000 : GOTO 38600
41100 IF N$ = "4" THEN CHAIN "A:S.MAIN",1000,ALL
41200 REM: INPUT ERROR : GOSUB 700
41300 GOTO 38600
41400 CLS : ERASE CP1,CP2,CS1,CS2,FS1,FS2,BF1,BF2,LS1,LS2
41500 DIM BF1(DP),BF2(DP) : REM: X-Y COORDINATES FOR SMOOTHED DATA.
41600 DIM T1(DP),T2(DP) : REM: TEMPORARY ARRAYS.
41700 NH = DP/NP : REM: NH = NO. OF ENTRIES UNDER EACH HEADER.
41710 CLS : LOCATE 10,25 : COLOR 0,7
41720 PRINT "BUTTERWORTH FILTER WORKING....." : COLOR 7,0
41800 BUTT1 = 1.41412
41900 BUTT2 = .1
42000 PI = 4 * ATAN(1)
42100 TEMP1 = PI*BUTT2
42200 TEMP2 = SIN(TEMP1) / COS(TEMP1)
42300 TEMP1 = TEMP2 * BUTT1
42400 TEMP2 = TEMP2 * TEMP2
42500 BUTT3 = TEMP1 + TEMP2 + 1
42600 BUTT4 = TEMP2 / BUTT3
42700 BUTT5 = BUTT4 * 2
42800 BUTT6 = 2 * (1 - TEMP2) / BUTT3
42900 BUTT7 = (TEMP1 - TEMP2 - 1) / BUTT3
43000 FOR K = 1 TO DP STEP NH
43100 T1(K) = X1(K) : T1(K+1) = X1(K+1)
43200 T2(K) = Y1(K) : T2(K+1) = Y1(K+1)
43300 NEXT K
43400 FOR MM = 1 TO DP STEP NH
43500 FOR II = MM+2 TO NH+MM-1
43600 T1(II) = BUTT4*X1(II)+BUTT5*X1(II-1)+BUTT4*X1(II-2)+BUTT6*T1(II-1)
         +BUTT7*T1(II-2)
43700 NEXT II
43800 FOR KK = MM TO NH+MM-1
43900 J = (MM+NH-1) + 1 -KK
44000 T2(J) = T1(KK)
44100 NEXT KK
44200 BF1(MM) = T1(MM) : BF1(MM+1) = T1(MM+1)
44300 BF2(MM) = T2(MM) : BF2(MM+1) = T2(MM+1)

```



```

44400     FOR I = MM+2 TO NH-MM-1
44500     BF1(I) = BUTT4*T1(I)+BUTT5*T1(I-1)+BUTT4*T1(I-2)+BUTT6*X1(I-1)+
           BUTT7*X1(I-2)
44600     BF2(I) = BUTT4*T2(I)+BUTT5*T2(I-1)+BUTT4*T2(I-2)+BUTT6*Y1(I-1)+
           BUTT7*Y1(I-2)
44700     NEXT I
44800     NZ = (MM+NH-1)/2
44900     REM: FOR J = 1 TO NZ
45000     REM: JJ = (MM+NH-1)+1 - J
45100     REM: XTEMP = X1(J) : YTEMP = Y1(J)
45200     REM: X1(JJ) = X1(JJ) : Y1(JJ) = Y1(JJ)
45300     REM: X1(JJ) = XTEMP : Y1(JJ) = YTEMP
45400     REM: NEXT J
45500     NEXT MM
45600     CLS
45700     LOCATE 10,18
45800     COLOR 0,7 : PRINT "BUTTERWORTH FILTERING CALCULATIONS COMPLETE" : COLOR 7,
0
45900     LOCATE 22,20
46000     INPUT "PRESS ANY KEY TO RETURN TO MENU",CC$
46100     ERASE T1,T2 : DIM CP1(1),CP2(1),FS1(1),FS2(1),CS1(1),CS2(1),LS1(1),LS2(1)
46200     GOTO 38600
50000     CLS
50100     REM:
50200     REM: *****
50300     REM: * VELOCITY CALCULATION *
50400     REM: *
50500     REM: * THIS SUBROUTINE WILL CALCULATE VELOCITY. IT *
50600     REM: * FINDS THE DERIVATIVES OF A SET OF DATA VALUES *
50700     REM: * BY MEANS OF THE FIRST FORWARD--FIRST CENTRAL *
50800     REM: * --AND FIRST BACKWARD FORMULAS FOR EITHER *
50900     REM: * UNIFORMLY OR NON-UNIFORMLY SPACED VALUES. *
51000     REM: *****
51100     REM:
51200     REM:
51300     REM: WRITE ROUTINE TO USE THE PROPER ARRAY WHEN CALCULATING VELOCITY.
51400     REM:
51500     DIM VC1(DP),VC2(DP)
51600     DIM V1(DP),V2(DP) : REM: TEMPORARY ARRAYS.
51700     REM:
51800     REM: CALCULATE VELOCITY.
51900     REM:
52000     IF VV$ = "BF" THEN FOR I = 1 TO DP : V1(I) = BF1(I) : V2(I) = BF2(I) :
           NEXT I : GOTO 52600
52100     IF VV$ = "CS" THEN FOR I = 1 TO DP : V1(I) = CS1(I) : V2(I) = CS2(I) :
           NEXT I : GOTO 52600
52200     IF VV$ = "CP" THEN FOR I = 1 TO DP : V1(I) = CP1(I) : V2(I) = CP2(I) :

```

```

NEXT I : GOTO 52600
52300 IF VV$ = "FS" THEN FOR I = 1 TO DP : V1(I) = FS1(I) : V2(I) = FS2(I) :
NEXT I : GOTO 52600
52400 IF VV$ = "LS" THEN FOR I = 1 TO DP : V1(I) = LS1(I) : V2(I) = LS2(I) :
NEXT I : GOTO 52600
52500 IF VV$ = "RD" THEN FOR I = 1 TO DP : V1(I) = X1(I) : V2(I) = Y1(I) :
NEXT I : GOTO 52600
52600 VC1(1) = (V1(2) - V1(1)) / TV(1) : REM: FIRST FORWARD-DIFFERENCE (X)
52700 VC2(1) = (V2(2) - V2(1)) / TV(1) : REM: FIRST FORWARD-DIFFERENCE (Y)
52800 FOR I = 2 TO DP - 1 : REM: FIRST CENTRAL-DIFFERENCE ARRAY.
52900 VC1(I) = (V1(I+1) - V1(I-1)) / (2*TV(I))
53000 VC2(I) = (V2(I+1) - V2(I-1)) / (2*TV(I))
53100 NEXT I
53200 VC1(DP) = (V1(DP) - V1(DP-1)) / TV(DP) : REM: FIRST BACKWARD-DIFFERENCE
53300 VC2(DP) = (V2(DP) - V2(DP-1)) / TV(DP) : REM: FIRST BACKWARD-DIFFERENCE
53400 REM: VELOCITY CALCULATION COMPLETE.
53500 ERASE V1 : ERASE V2
53600 CLS
53700 LOCATE 10,20
53800 COLOR 0,7 : PRINT "VELOCITY CALCULATION COMPLETE" : COLOR 7,0
53900 LOCATE 22,20
54000 INPUT "PRESS ANY KEY TO RETURN TO MENU",CC$
54100 RETURN
55000 CLS
55100 REM:
55200 REM: *****
55300 REM: * ACCELERATION CALCULATION *
55400 REM: *****
55500 REM:
55600 DIM AC1(DP),AC2(DP)
55700 DIM A1(DP),A2(DP) : REM: TEMPORARY ARRAYS.
55800 IF AA$ = "BF" THEN FOR I = 1 TO DP : A1(I) = BF1(I) : A2(I) = BF2(I) :
NEXT I : GOTO 56400
55900 IF AA$ = "CS" THEN FOR I = 1 TO DP : A1(I) = CS1(I) : A2(I) = CS2(I) :
NEXT I : GOTO 56400
56000 IF AA$ = "CP" THEN FOR I = 1 TO DP : A1(I) = CP1(I) : A2(I) = CP2(I) :
NEXT I : GOTO 56400
56100 IF AA$ = "FS" THEN FOR I = 1 TO DP : A1(I) = FS1(I) : A2(I) = FS2(I) :
NEXT I : GOTO 56400
56200 IF AA$ = "LS" THEN FOR I = 1 TO DP : A1(I) = LS1(I) : A2(I) = LS2(I) :
NEXT I : GOTO 56400
56300 IF AA$ = "RD" THEN FOR I = 1 TO DP : A1(I) = X1(I) : A2(I) = Y1(I) :
NEXT I : GOTO 56400
56400 AC1(1) = (A1(3) - 2*A1(2) + A1(1)) / TV(1)^2 : REM: FIRST FORWARD-DIFF.
56500 AC2(1) = (A2(3) - 2*A2(2) + A2(1)) / TV(1)^2 : REM: FIRST FORWARD-DIFF.
56600 FOR I = 2 TO DP - 1 : REM: FIRST CENTRAL-DIFFERENCE ARRAY.

```

```

56700      AC1(I) = (A1(I+1) -2*A1(I) + A1(I-1)) / TV(I)^2
56800      AC2(I) = (A2(I+1) -2*A2(I) + A2(I-1)) / TV(I)^2
56900      NEXT I
57000 AC1(DP) = (A1(DP) -2*A1(DP-1) + A1(DP-2)) / TV(DP)^2 : REM: FIRST BACK.
57100 AC2(DP) = (A2(DP) -2*A2(DP-1) + A2(DP-2)) / TV(DP)^2 : REM: FIRST BACK.
57200 REM: ACCELERATION CALCULATION COMPLETE.
57300 ERASE A1 : ERASE A2
57400 CLS : LOCATE 10,20
57500 COLOR 0,7 : PRINT "ACCELERATION CALCULATION COMPLETE" : COLOR 7,0
57600 LOCATE 22,20
57700 INPUT "PRESS ANY KEY TO RETURN TO MENU",CC$
57800 RETURN

```

```

1000 CLS
1200 REM:
1400 REM: *****
1600 REM: *          PRINTING DATA SUBROUTINE          *
1800 REM: *****
2000 REM:
2200 LOCATE 5,30
2400 PRINT "PRINTING DATA"
2600 LOCATE 6,30
2800 PRINT "*****"
3000 LOCATE 10,20
3200 PRINT "PLEASE SELECT ONE OF THE FOLLOWING:"
3400 LOCATE 12,15
3600 PRINT "PRINT DATA..... 1"
3800 LOCATE 13,15
4000 PRINT "RETURN TO MAIN MENU..... 2"
4200 LOCATE 15,25
4400 INPUT "YOUR SELECTION PLEASE: ",N$
4600 IF N$ = "1" THEN GOSUB 5800
4800 IF N$ = "2" THEN CHAIN "A:S.MAIN",1000,ALL
5000 CLS : REM: INPUT ERROR
5200 ER$ = "A:S.PRINT" : LN = 5600
5400 CHAIN "A:S.ERROR",1000,ALL
5600 GOTO 1000
5800 CLS : REM: PRINTING DATA SUBROUTINE.
6000 LOCATE 1,30
6200 PRINT "PRINTING DATA"
6400 LOCATE 2,30
6600 PRINT "*****"
6800 LOCATE 4,35
7000 PRINT "PRINT:"
7200 LOCATE 6,15
7400 PRINT "BUTTERWORTH FILTERING SMOOTHED DATA.....(BF)"
7600 LOCATE 7,15
7800 PRINT "CUBIC SPLINE SMOOTHED DATA.....(CS)"
8000 LOCATE 8,15
8200 PRINT "CHEBYSHEV POLYNOMIAL SMOOTHED DATA.....(CP)"
8400 LOCATE 9,15
8600 PRINT "FOURIER SERIES SMOOTHED DATA.....(FS)"
8800 LOCATE 10,15
9000 PRINT "LEAST SQUARES SMOOTHED DATA.....(LS)"
200 LOCATE 11,15
400 PRINT "RAW DATA.....(RD)"
600 LOCATE 12,15
800 PRINT "RETURN TO MENU.....(MM)"
0000 LOCATE 14,15
0200 PRINT "MAKE YOUR SELECTION BY ENTERING THE TWO DIGIT CODE"
0400 LOCATE 15,12
0600 PRINT "IF YOU WOULD LIKE THE RESULTANT VELOCITY OR ACCELERATION"
0800 LOCATE 16,13
1000 PRINT "FROM THE DATA PLACE AN (A) OR (V) IN FRONT OF THE CODE"
1200 LOCATE 18,12

```

```

1400 PRINT "EX: TO PRINT RESULTANT ACCELERATION FROM THE SMOOTHED DATA"
1600 LOCATE 19,27
1800 PRINT "BY CUBIC SPLINE FUNCTION"
2000 LOCATE 20,34
2200 PRINT "ENTER: "; : COLOR 0,7 : PRINT "ACS" : COLOR 7,0
2400 LOCATE 22,28
2600 INPUT "YOUR SELECTION PLEASE:",SD$
2800 REM:
3000 REM: THE DATA WILL BE PRINTED WITH THE X-COORDINATE BEING SAVED & THE
3200 REM: Y-COORDINATE WILL BE REPLACED WITH THE SMOOTHED VALUE.
3400 REM:
3600 DIM T1(DP),T2(DP) : REM: TEMPORARY ARRAYS.
3800 IF SD$ = "BF" THEN FOR I = 1 TO DP : T1(I) = BF1(I) : T2(I) = BF2(I) :
NEXT I : GOTO 16000
4000 IF SD$ = "CS" THEN FOR I = 1 TO DP : T1(I) = CS1(I) : T2(I) = CS2(I) :
NEXT I : GOTO 16000
4200 IF SD$ = "CP" THEN FOR I = 1 TO DP : T1(I) = CP1(I) : T2(I) = CP2(I) :
NEXT I : GOTO 16000
4400 IF SD$ = "FS" THEN FOR I = 1 TO DP : T1(I) = FS1(I) : T2(I) = FS2(I) :
NEXT I : GOTO 16000
4600 IF SD$ = "LS" THEN FOR I = 1 TO DP : T1(I) = LS1(I) : T2(I) = LS2(I) :
NEXT I : GOTO 16000
4800 IF SD$ = "RD" THEN FOR I = 1 TO DP : T1(I) = X1(I) : T2(I) = Y1(I) :
NEXT I : GOTO 16000
5000 IF SD$ = "ABF" OR SD$ = "ACS" OR SD$ = "AFD" OR SD$ = "AFS" OR SD$ = "ALS"
OR SD$ = "ARD" THEN FOR I = 1 TO DP : T1(I) = AC1(I) : T2(I) = AC2(I) :
NEXT I : GOTO 16000
5200 IF SD$ = "VBF" OR SD$ = "VCS" OR SD$ = "VFD" OR SD$ = "VFS" OR SD$ = "VLS"
OR SD$ = "VRD" THEN FOR I = 1 TO DP : T1(I) = VC1(I) : T2(I) = VC2(I) :
NEXT I : GOTO 16000
5400 IF SD$ = "MM" THEN ERASE T1 : ERASE T2 : GOTO 1000
15600 REM: INPUT ERROR : GOSUB 700
15800 GOTO 1000
16000 REM:
16200 REM: SEND ARRAY TO PRINTER.
16400 REM:
16600 CLS
16800 DIM PT(DP+1) : PT(1) = 0
17000 NH = DP/NP : REM: NH = NO. OF ENTRIES UNDER EACH HEADER.
17200 LOCATE 10,15
17400 COLOR 0,7 : PRINT "REMEMBER TO ALIGN PAPER TO TOP OF A SHEET" : COLOR 7,0
17600 LOCATE 10,58
17800 COLOR 0,7 : PRINT "!!!" : COLOR 7,0
18000 LOCATE 22,20
18200 INPUT "PRESS ANY KEY TO BEGIN PRINTING",CC$
18400 LOCATE 1,32 : LPRINT "FILE DESCRIPTION" : LPRINT
18600 IF HC = 100 THEN LOCATE 3,1 : LPRINT D1$

```

```

18800 IF HC = 100 THEN LOCATE 4,1 : LPRINT D2$
19000 IF HC <> 100 THEN LOCATE 3,1 : LPRINT "PRESENT FILE HAS"
19200 IF HC <> 100 THEN LOCATE 4,1 : LPRINT "NO FILE DESCRIPTION"
19400 LOCATE 6,1
19600     FOR JJ = 0 TO NP - 1
19800         LPRINT TAB((JJ+1)*20+3) PH$(JJ+1);
20000     NEXT JJ
20200 LOCATE 7,1 : LPRINT TAB(2) "TIME VALUE";
20400     FOR JJ = 0 TO NP - 1
20600         LPRINT TAB((JJ+1)*20+1) "X" TAB((JJ+1)*20+12) "Y";
20800     NEXT JJ
21000 LPRINT
21200 LOCATE 9,1
21400     FOR II = 1 TO NH : PT(II+1) = PT(II)+TV(II) : PT(II+1) = (INT(PT(II+1)
)*1000)/1000 : LPRINT TAB(3) PT(II);
21600     TC = 0
21800     FOR KK = II TO DP STEP NH
22000         LPRINT TAB((TC+1)*20) T1(KK) TAB((TC+1)*20+10) T2(KK);
22200         TC = TC + 1
22400     NEXT KK
22600     NEXT II
22800 LPRINT : LPRINT : LPRINT : LPRINT
23000 LPRINT TAB(20); "END OF PRINTOUT*****"
23200 CLS
23400 LOCATE 10,15
23600 INPUT "END OF DATA--PRESS ANY KEY TO RETURN TO MENU",CC$
23800 ERASE T1 : ERASE T2 : ERASE PT
24000 GOTO 1000

```

```

1000 CLS
1100 REM:
1200 REM: *****
1300 REM: *          GRAPHING DATA SUBROUTINE          *
1400 REM: *****
1500 REM:
1600 LOCATE 5,30
1700 PRINT "GRAPHING DATA"
1800 LOCATE 6,30
1900 PRINT "*****"
2000 LOCATE 9,20
2100 PRINT "PLEASE SELECT ONE OF THE FOLLOWING:"
2200 LOCATE 11,15
2300 PRINT "GRAPH DATA..... 1"
2400 LOCATE 12,15
2500 PRINT "SEND GRAPH TO PRINTER..... 2"
2600 LOCATE 13,15
2700 PRINT "RETURN TO MAIN MENU..... 3"
2800 LOCATE 15,25
2900 INPUT "YOUR SELECTION PLEASE: ",N$
3000 IF N$ = "1" THEN GOSUB 3700
3100 IF N$ = "2" THEN GOTO 1000
3200 IF N$ = "3" THEN CHAIN "A:S.MAIN",1000,ALL
3300 CLS : REM: INPUT ERROR
3400 ER$ = "A:S.GRAPH" : LN = 3600
3500 CHAIN "A:S.ERROR",1000,ALL
3600 GOTO 1000
3700 CLS : REM: GRAPHING DATA
3800 LOCATE 1,25
3900 PRINT "GRAPHING DATA SUBROUTINE"
4000 LOCATE 2,25
4100 PRINT "*****"
4200 LOCATE 4,35
4300 PRINT "GRAPH:"
4400 LOCATE 6,15
4500 PRINT "BUTTERWORTH FILTERING SMOOTHED DATA.....(BF)"
4600 LOCATE 7,15
4700 PRINT "CUBIC SPLINE SMOOTHED DATA.....(CS)"
4800 LOCATE 8,15
4900 PRINT "CHEBYSHEV POYNOMIAL DATA.....(CP)"
5000 LOCATE 9,15
5100 PRINT "FOURIER SERIES SMOOTHED DATA.....(FS)"
5200 LOCATE 10,15
5300 PRINT "LEAST SQUARES SMOOTHED DATA.....(LS)"
5400 LOCATE 11,15
5500 PRINT "RAW DATA.....(RD)"
5600 LOCATE 12,15
5700 PRINT "RETURN TO MENU.....(MM)"
5800 LOCATE 14,15
5900 PRINT "MAKE YOUR SELECTION BY ENTERING THE TWO DIGIT CODE"

```

```

6000 LOCATE 15,12
6100 PRINT "IF YOU WOULD LIKE THE RESULTANT VELOCITY OR ACCELERATION"
6200 LOCATE 16,13
6300 PRINT "FROM THE DATA PLACE AN (A) OR (V) IN FRONT OF THE CODE"
6400 LOCATE 18,12
6500 PRINT "EX: TO GRAPH RESULTANT ACCELERATION FROM THE SMOOTHED DATA"
6600 LOCATE 19,27
6700 PRINT "BY CUBIC SPLINE FUNCTION"
6800 LOCATE 20,34
6900 PRINT "ENTER: "; : COLOR 0,7 : PRINT "ACS" : COLOR 7,0
7000 LOCATE 22,25
7100 INPUT "YOUR SELECTION PLEASE:",SD$
7200 REM:
7300 REM: THE DATA WILL BE GRAPHED WITH THE X-COORDINATE BEING SAVED & THE
7400 REM: Y-COORDINATE WILL BE REPLACED WITH THE SMOOTHED VALUES.
7500 REM:
7600 DIM T1(DP),T2(DP) : REM: TEMPORARY ARRAYS.
7700 IF SD$ = "BF" THEN FOR I = 1 TO DP : T1(I) = BF1(I) : T2(I) = BF2(I) :
NEXT I : GOTO 8800
7800 IF SD$ = "CS" THEN FOR I = 1 TO DP : T1(I) = CS1(I) : T2(I) = CS2(I) :
NEXT I : GOTO 8800
7900 IF SD$ = "CP" THEN FOR I = 1 TO DP : T1(I) = CP1(I) : T2(I) = CP2(I) :
NEXT I : GOTO 8800
8000 IF SD$ = "FS" THEN FOR I = 1 TO DP : T1(I) = FS1(I) : T2(I) = FS2(I) :
NEXT I : GOTO 8800
8100 IF SD$ = "LS" THEN FOR I = 1 TO DP : T1(I) = LS1(I) : T2(I) = LS2(I) :
NEXT I : GOTO 8800
8200 IF SD$ = "RD" THEN FOR I = 1 TO DP : T1(I) = X1(I) : T2(I) = Y1(I) :
NEXT I : GOTO 8800
8300 IF SD$ = "ABF" OR SD$ = "ACS" OR SD$ = "AFD" OR SD$ = "AFS" OR SD$ = "ALS"
OR SD$ = "ARD" THEN FOR I = 1 TO DP : T1(I) = AC1(I) : T2(I) = AC2(I) :
NEXT I : GOTO 8800
8400 IF SD$ = "VBF" OR SD$ = "VCS" OR SD$ = "VFD" OR SD$ = "VFS" OR SD$ = "VLS"
OR SD$ = "VRD" THEN FOR I = 1 TO DP : T1(I) = VC1(I) : T2(I) = VC2(I) :
NEXT I : GOTO 8800
8500 IF SD$ = "MM" THEN ERASE T1 : ERASE T2 : GOTO 1000
8600 REM: INPUT ERROR : GOSUB 700
8700 GOTO 1000
8800 REM:
8900 REM: *****
9000 REM: * GRAPHING DATA SUBROUTINE *
9100 REM: *****
9200 REM:
9300 CLS : ERASE BF1,BF2,CS1,CS2,CP1,CP2,FS1,FS2,LS1,LS2
9400 DIM G1(DP),G2(DP),G3(DP),G4(DP)
9500 REM: ARRAYS FOR DETERMINING SCALING.
9600 IF PT(1) >= 0 THEN ERASE PT

```



```

9700 DIM PT(DP+1),TP(DP+1)
9800 LOCATE 5,18
9900 PRINT "WHICH OF THE FOLLOWING DO YOU WISH TO GRAPH:"
10000 PRINT : PRINT : PRINT
10100   FOR I = 1 TO NP
10200     PRINT TAB(5) PH*(I) TAB(20) "..... ";I
10300   NEXT I
10400 PRINT TAB(5) "OTHER" TAB(20) "..... ";NP+1
10500 LOCATE 20,29 : INPUT "YOUR SELECTION PLEASE: ",IS
10600 IF IS < NP+1 THEN GOTO 12100
10700 CLS
10800 LOCATE 5,10
10900 PRINT "YOU MAY GRAPH TWO BODY SEGMENTS AGAINST EACH OTHER AT A TIME"
11000 LOCATE 7,22
11100 PRINT "PLEASE SELECT TWO OF THE FOLLOWING: "
11200 PRINT : PRINT : PRINT
11300   FOR I = 1 TO NP
11400     PRINT TAB(5) PH*(I) TAB(20) "..... ";I
11500   NEXT I
11600   PRINT TAB(5) "RETURN TO MENU" TAB(20) "..... ";NP+1
11700 LOCATE 20,26 : COLOR 0,7 : INPUT "YOUR FIRST SELECTION PLEASE: ",IS :
      COLOR 7,0
11800 IF IS = NP+1 GOTO 15200
11900 LOCATE 22,26 : COLOR 0,7 : INPUT "YOUR SECOND SELECTION PLEASE: ",SI :
      COLOR 7,0
12000 CHECK = 2 : GOSUB 15500 : GOTO 12200
12100 CHECK = 1 : GOSUB 15500
12200 REM: SET UP X-Y COORDINATE AXIS.
12300 SCREEN 1,0,0
12400 LINE (20,150)-(20,0)
12500 LINE (20,150)-(320,150)
12600 REM: GRAPH X-Y COORDINATES -VS- TIME
12700 IF CHECK = 2 THEN GOTO 13700
12800   FOR I = (NH*IS-(NH-1)) TO NH*IS-1
12900     LINE (PT(I),G1(I))-(PT(I+1),G1(I+1)) : REM: X-GRAPH
13000     LINE (PT(I),G2(I))-(PT(I+1),G2(I+1)) : REM: Y-GRAPH
13100   NEXT I
13200 LOCATE 20,11
13300 PRINT "X-GREEN  Y-YELLOW"
13400 LOCATE 22,5
13500 INPUT "PRESS ANY KEY TO RETURN TO MENU",CC#
13600 GOTO 15000
13700 REM: GRAPH TWO SEGMENTS.
13800   FOR I = (NH*IS-(NH-1)) TO NH*IS-1
13900     LINE (PT(I),G1(I))-(PT(I+1),G1(I+1))
14000     LINE (PT(I),G2(I))-(PT(I+1),G2(I+2))
14100   NEXT I
14200   FOR J = (NH*SI-(NH-1)) TO NH*SI-1
14300     LINE (TP(J),G3(J))-(TP(J+1),G3(J+1))
14400     LINE (TP(J),G4(J))-(TP(J+1),G4(J+1))

```

```

14500          NEXT J
14600 LOCATE 20,5
14700 PRINT PH$(IS);"-GREEN  ";PH$(SI);"-YELLOW"
14800 LOCATE 22,5
14900 INPUT "PRESS ANY KEY TO RETURN TO MENU",CC$
15000 SCREEN 0,1,0,0 : COLOR 7,0
15100 WIDTH 80
15200 ERASE G1,G2,G3,G4,T1,T2,PT,TP
15300 DIM BF1(1),BF2(1),CS1(1),CS2(1),CP1(1),CP2(1),FS1(1),FS2(1),LS1(1),LS2(1)
15400 GOTO 1000
15500 REM:
15600 REM: SUBROUTINE TO DETERMINE SCALING.
15700 REM:
15800 NH = DP/NP : REM: NH = NO. OF HEADERS.
15900   FOR I = 1 TO DP
16000     G1(I)=T1(I) : G2(I)=T2(I) : G3(I)=T1(I) : G4(I)=T2(I)
16100   NEXT I
16200   FOR I = (NH*IS-(NH-1)) TO NH*IS-1
16300     FOR J = I+1 TO NH*IS
16400       IF T1(I) >= T1(J) THEN GOTO 16800
16500       TEMP = T1(J)
16600       T1(J) = T1(I)
16700       T1(I) = TEMP
16800     NEXT J
16900   NEXT I
17000   FOR I = (NH*IS-(NH-1)) TO NH*IS
17100     FOR J = I+1 TO NH*IS
17200       IF T2(I) >= T2(J) THEN GOTO 17600
17300       TEMP = T2(J)
17400       T2(J) = T2(I)
17500       T2(I) = TEMP
17600     NEXT J
17700   NEXT I
17800 REM: SET MIN-MAX VALUES.
17900 XMIN = T1(NH*IS)
18000 XMAX = T1(NH*IS-(NH-1)) : X RANGE = XMAX - XMIN
18100 YMIN = T2(NH*IS)
18200 YMAX = T2(NH*IS-(NH-1)) : Y RANGE = YMAX - YMIN
18300 REM: SCALE COORDINATES.
18400 IF XMIN >= 0 THEN 18900
18500   FOR I = (NH*IS-(NH-1)) TO NH*IS
18600     G1(I) = G1(I)+ABS(XMIN)
18700   NEXT I
18800 GOTO 19200
18900   FOR I = (NH*IS-(NH-1)) TO NH*IS
19000     G1(I) = G1(I)-XMIN
19100   NEXT I

```

```

19200 IF YMIN >= 0 THEN GOTO 19700
19300     FOR I = (NH*IS-(NH-1)) TO NH*IS
19400         G2(I) = G2(I)+ABS(YMIN)
19500     NEXT I
19600 GOTO 20000
19700     FOR I = (NH*IS-(NH-1)) TO NH*IS
19800         G2(I) = G2(I)-YMIN
19900     NEXT I
20000 XCON = 150 / (XMAX-XMIN)
20100 YCON = 150 / (YMAX-YMIN)
20200     FOR I = (NH*IS-(NH-1)) TO NH*IS
20300         G1(I) = G1(I)*XCON
20400         G1(I) = 150 - G1(I)
20500         G2(I) = G2(I)*YCON
20600         G2(I) = 150 - G2(I)
20700     NEXT I
20800 IF CHECK = 1 THEN GOTO 25500
20900     FOR I = (NH*SI-(NH-1)) TO NH*SI-1
21000         FOR J = I+1 TO NH*SI
21100             IF T1(I) >= T1(J) THEN GOTO 21500
21200             TEMP=T1(J)
21300             T1(J)=T1(I)
21400             T1(I)=TEMP
21500         NEXT J
21600     NEXT I
21700     FOR I = (NH*SI-(NH-1)) TO NH*SI
21800         FOR J = I+1 TO NH*SI
21900             IF T2(I) >= T2(J) THEN GOTO 22300
22000             TEMP=T2(J)
22100             T2(J)=T2(I)
22200             T2(I)=TEMP
22300         NEXT J
22400     NEXT I
22500 REM: SET MIN-MAX VALUES
22600 XMIN2 = T1(NH*SI)
22700 XMAX2 = T1(NH*SI-(NH-1))
22800 YMIN2 = T2(NH*SI)
22900 YMAX2 = T2(NH*SI-(NH-1))
23000 REM: SCALE COORDINATES.
23100 IF XMIN2 >= 0 THEN GOTO 23600
23200     FOR I = (NH*SI-(NH-1)) TO NH*SI
23300         G3(I) = G3(I) + ABS(XMIN2)
23400     NEXT I
23500 GOTO 23900
23600     FOR I = (NH*SI-(NH-1)) TO NH*SI
23700         G3(I) = G3(I) - XMIN2
23800     NEXT I
23900 IF YMIN2 >= 0 THEN GOTO 24400

```

```

24000     FOR I = (NH*SI-(NH-1)) TO NH*SI
24100     G4(I) = G4(I) + ABS(YMIN2)
24200     NEXT I
24300 GOTO 24700
24400     FOR I = (NH*SI-(NH-1)) TO NH*SI
24500     G4(I) = G4(I) - YMIN2
24600     NEXT I
24700     XCON2 = 150 / (XMAX2-XMIN2)
24800     YCON2 = 150 / (YMAX2-YMIN2)
24900     FOR I = (NH*SI-(NH-1)) TO NH*SI
25000     G3(I) = G3(I)*XCON2
25100     G3(I) = 150 - G3(I)
25200     G4(I) = G4(I)*YCON2
25300     G4(I) = 150 - G4(I)
25400     NEXT I
25500 CLS
25600 LOCATE 5,17
25700 PRINT "GRAPHING IS DONE ONLY FOR CONSTANT TIME VALUES"
25800 LOCATE 15,28
25900 INPUT "CONTINUE.....(Y/N)",YN#
26000 IF YN# = "Y" THEN GOTO 26300
26100 ERASE T1,T2,G1,G2 : DIM BF1(1),BF2(1),CS1(1),CS2(1),FD1(1),FD2(1),FS1(1),
      FS2(1),LS1(1),LS2(1)
26200 GOTO 1000
26300 REM: SET TIME VALUES.
26400     FOR I = (NH*IS-(NH-1)) TO NH*IS
26500     PT(I+1) = PT(I) + TV(I)
26600     NEXT I
26700     TCON = 300/PT(NH*IS)
26800     FOR I = (NH*IS-(NH-1)) TO NH*IS
26900     PT(I) = PT(I)*TCON
27000     PT(I) = PT(I) + 20
27100     NEXT I
27200 IF CHECK = 1 THEN GOTO 28200
27300 TP(NH*SI-(NH-1)) = 0
27400     FOR I = (NH*SI-(NH-1)) TO NH*SI
27500     TP(I+1) = TP(I)+TV(I)
27600     NEXT I
27700     TCON2 = 300/TP(NH*SI)
27800     FOR I = (NH*SI-(NH-1)) TO NH*SI
27900     TP(I) = TP(I)*TCON2
28000     TP(I) = TP(I) + 20
28100     NEXT I
28200 RETURN

```



MICHIGAN STATE UNIV. LIBRARIES



31293007869286