SOUTHWEST FISHERIES CENTER

NATIONAL MARINE FISHERIES SERVICE SOUTHWEST FISHERIES CENTER P.O. BOX 271 LA JOLLA, CA 92038

MAY 1983

DOCUMENTATION FOR COMPUTER PROGRAMS USED TO PRODUCE SIMULATED LENGTH DISTRIBUTIONS FOR SPINNER DOLPHIN POPULATIONS

By

Michael Lichter, Wayne Perryman
and Douglas DeMaster

ADMINISTRATIVE REPORT LJ-83-10

DOCUMENTATION FOR COMPUTER PROGRAMS USED TO PRODUCE SIMULATED LENGTH DISTRIBUTIONS FOR SPINNER DOLPHIN POPULATIONS

Michael Lichter, Wayne Perryman
and Douglas DeMaster
National Marine Fisheries Service
Southwest Fisheries Center
La Jolla, California 92038

May 1983

Administrative Report LJ-83-10

CONTENTS

DOCUMENTATION FOR COMPUTER PROGRAMS USED TO PRODUCE SIMULATED LENGTH DISTRIBUTIONS FOR SPINNER DOLPHIN POPULATIONS

Michael Lichter, Wayne Perryman
and Douglas DeMaster

National Marine Fisheries Service
Southwest Fisheries Center
La Jolla, California 92038

INTRODUCTION

The computer programs described in this document form part of an effort to arrive at an estimate of crude birth rate for dolphin stocks through the use of aerial photography. Existing techniques allow the estimation of object sizes from photographs with great accuracy. These techniques can be applied to aerial photographs of dolphin schools, allowing the generation of corresponding length distributions for the photographed schools. Needed for an estimate of crude birth rate is a good estimate of the relative abundance of animals less than one year old in the population. If the relationship between age and length is known, or a reasonable estimate is available, the length frequency distribution generated from photographs may be used to estimate the crude birth rate.

Within what?

This report addresses the following queries:

1. With what accuracy can animals be aged by length when taking into consideration the overlap in lengths of different age groups?

2. Up to what animal length can length-to-age conversions be considered reliable, given the asymptotic nature of dolphin growth and the variability in growth rates?

3. How do the answers to the above questions differ given variation in parameters such as the periodicity of reproduction, and the range of lengths in each age class?

The final output of these programs is a length frequency distribution. This distribution is calculated by taking the estimated percentage of animals in each age class and multiplying it by the estimated distribution of lengths for that age class. This yields the percentages of animals in particular length classes, which are then summed over all ages to generate the length-frequency distribution.

A paper is being prepared which analyzes the data produced by these programs. The purpose of this document is to present the methods and ideas used in the development of the programs, as well as to describe the function of the programs developed and to serve as a user's guide to them.

PHOTOGRAMMETRY

Length measurements from aerial photographs have been used to estimate reproductive parameters for a number of mammal species. The technique has been only recently applied to small cetaceans because of technical problems associated with photographing these animals from the air. Based upon preliminary analysis of field data, this technique should provide estimates of animal lengths within 5% of actual size. The technique produces data with a consistent negative bias because the movement and orientation of an animal will make its apparent length, when seen from above, less than its actual length.

DOLPHIN GROWTH

Determination of age from length, which is at the core of these programs, depends upon a well-behaved model of animal growth. Perrin, Holts, and Miller (1977), and Perrin, Holts, and Zweifel (1975), found that two-phase versions of Laird's (1969) growth model work well in fitting the growth of eastern spinner dolphins (*Stenella longirostris*), and the spotted dolphin (*Stenella attenuata*).

In these programs the results of Perrin et al. (1976 and 1977) are used to generate the mean length for each age class considered. The two papers examine three areas in length-to-age conversions: juvenile, adult female, and adult male growth. In each paper, Lairds' growth model is fitted to each of the three growth curves. The adult curves match fairly closely, but the model fits the female curve better and so it is used for the adult phase of the length function.

These models were developed using tooth layers for aging, rather than years. Hypotheses estimating the number of dental layers accumulated per year proposed in Perrin et al. (1977), are as follows:

1. 1.5 layers in the first year and 1 per year thereafter.

2. 1.5 layers per year.

3. 1.5 layers per year until puberty (about 9 layers in males and 5-6 layers in females) and 1 per year thereafter

For eastern spinner dolphins, Hypothesis 1. appears to be the best supported of the three.

POPULATION STRUCTURE

The dolphin age structures generated in these programs come from the standard set of Lotka equations:

$$1 = \sum \lambda^{-x} \ell_x m_x$$

$$\frac{1}{\beta} = \sum \lambda^{-x} \ell_x$$

$$c_x = \beta \lambda^{-x} \ell_x$$

where c(x) is the population age structure, β is known as the birth rate per capita,($\lambda=e^r$), r is the instantaneous rate of population change, and $\ell(x)$ is the survivorship to age x. For the purpose of this experiment, the size of the population was assumed to be constant (although the programs allow for increase/decrease), and the $\ell(x)$ function was generated from the mortality function $\mu(x)$ given by Keyfitz (1968):

$$\ell_x = \exp\left[-\int_0^x \mu(t)dt\right]$$

The functions concerned with the population structure modify it in two ways. First, they change the age structure generated by the Lotka equations to simulate the passage of time between reproductive cycles. Current evidence points to the birth of either one or two cohorts of animals per year. This factor is brought into play by adjusting the age structures at each one-tenth year stage in the reproductive cycle. Secondly, these functions modify the appearance of the age structure slightly by assuming that the birth of each cohort is spread out over three time periods with a peak in the center and an arbitrary percentage on either side.

ORGANIZATION

The organization of the calculations carried out by the set of functions relies heavily on the example of Clark (1981), whose statement of the problem fits in well with the goals of this experiment. Clark introduces a matrix X such that X(i,j) is the proportion of animals in age class j whose lengths are in the length interval i. He also defines a column vector PI whose elements PI(j) are the proportion of the total population belonging to age group j. PI is synonomous with the age structure, c(x). Finally, the column vector P is defined such that P(i) gives the portion of the population that falls within length interval i. It follows that, in matrix notation,

$$(X)\ (PI) = P$$

where, again, PI represents the population age distribution, P represents that of length, and X is a matrix whose columns represent the length composition of each age group.

Although the PI vector (population structure) varies given seasonal variations in reproduction and other factors, the length composition within an age group remains relatively constant, meaning that the X matrix for a given animal can be treated as an invariant quantity. This simplifies the job of the programs considerably, because once an X matrix is constructed, modifications can be performed on the c(x) and new P vectors (length distribution) attained without repeating the heavy computation involved in X matrix generation.

PROGRAM STRUCTURE

The operation of the programs described herein follows a two-level hierarchy, with the DOLPHINS workspace supervising the XS and XP workspace. The XS workspace creates data files, the XP workspace processes them, and the DOLPHINS workspace creates files of commands, or command files, which govern the operations of the other two workspaces. The use of command files allows the specification of many runs at one time so that human supervision is not necessary, and programs may run overnight.

The problem of deriving a length distribution divides into three parts, as described above: 1) finding the proportion of each age class in each length group, 2) finding the proportion of the population in each age group, and 3) multiplying the two to arrive at a length distribution. XP performs the first function, and deposits the results in a file. XS performs the second function and then puts the two together to perform the third and to obtain graphical and numerical results. The steps involved in these processes are detailed below. Names in parentheses are of the functions which perform the calculations.

Workspace XS (governed by the function XSET)

1. Create vector of ages (i.e. 1, 2, 3, 4, 5,....) (XSET).

2. Convert ages in years into ages in tooth layers (LAYEARS).

3. Translate ages into their associated mean length (LEN).

4. Assuming a normal distribution, calculate the percentage of each age group at each length group within three standard deviations of the calculated mean for that age; this is the X matrix (actually, the transpose of the X matrix) and is calculated in 5-year intervals (AREA, MAREA, NORM, STAND, XSET).

Workspace XP (Governed by the function LFREQ)

1. Enter CYCLE loop, which iterates once for every unique configuration of the population (once for ten reproductive pulses per year, ten times for one pulse per year - where the smallest unit of time is 0.1

years) (LFREQ).

2. Create age vector corresponding to time of year, i.e., at T=0, age=0, 0.5, 1, 1.5, ... and at T=0.4, age=0.4, 0.9, 1.4, 1.9 ....(LFREQ).

3. Create the Cx from the current age vector (XPROC).

4. Modify Cx to include any spread desired in the age structure or the pulsing of reproduction (XPROC).

5. Matrix multiply X by PI. The result is held in P, and the intermediate result giving the percentages of the total population found in each length group of each age group is stored on disk (XPROC).

The System

These functions were designed on a Z-80 based microcomputer and written in APL/V80, a subset of the APL language marketed by Vanguard Systems Corporation.[1] Much of the effort consumed in the programming went towards learning and working around the drawbacks of the system. The lack of memory space made a simple and elegant design impossible and necessitated dividing up large arrays and storing them on disk, and swapping functions in and out of the XP workspace.

APL/V80 lacks the transpose and inner product operators normally found in APL systems, and has implementation restrictions on others. Those functions not included in the interpreter are found as defined functions, i.e., TRANSPOSE and INPROD. The file-handling functions in this system were also implemented as defined functions. The need to have these functions in a workspace makes the already bad memory space situation even worse.

Transfer of these functions from microcomputer to mainframe usage would allow quite a bit of simplification and would save time. As implemented, a single run of the XP workspace may take as long as several hours. The availability of a more powerful system on a faster machine with more memory space would improve efficiency.

Operation

The purpose of the DOLPHINS workspace is to facilitate the use of these programs for multiple runs without supervision, especially for users with little experience in APL. To run the DOLPHINS workspace on the system in which it was written, the following are necessary: a disk with APL and the necessary workspaces and functions, and a destination disk for the files to be

---

[1]Brand name/disclaimer.

accessed and written. Given that two drives are available, numbered 0 and 1, place APL and functions disk in the first drive, and the files disk in the second. Reset the system. The CP/M operating system will respond with the prompt:

A>

Initialize the APL system by typing "APL <cr>" where "<cr>" denotes a carriage return. This should look like

A> APL <cr>

APL/V80 will prompt for the date with

MMDDYY

with "090381 <cr>" being an appropriate response. APL will then display some log-in information and echo "9-3-81." Afterwards enter the command

)LOAD DOLPHINS <cr>

After a moment, the prompt will appear:

C)REATE X FILE, R)UN CALCULATIONS, S)ET PARAMETERS

Typing an "S" followed by <cr> will initiate Set Mode, and print the prompt

C)HANGE, D)ISPLAY, M)AKE FILE, Q)UIT, W)RITE PARAMS

Entering a "D" <cr> will display the current values of the parameters for XS and XP. The display function will wait for a <cr> before returning to the Set Mode command level. The "M" <cr> command creates a command file for any number of runs of either XS or XP (but not both simultaneously). It will ask

NAME OF COMMAND FILE?

Any 8-character name is an appropriate response. After this is specified, it will prompt

NUMBER OF ENTRIES?

which is the number of separate runs that will be initiated by the command file which has just been named. This given, a file is created. This file, however, will remain empty until a complete set of parameters is written to it. Before writing to the file, if parameter values other than the default values are desired, it is necessary to change them as follows. Type "C" <cr> and wait for the prompt

CHANGE WHAT VARIABLE?

Enter the variable name, and <cr>. DOLPHINS will respond with

ENTER NEW VALUE(S)

Values may be given either as numbers (or letters, when appropriate) or as variable names, when corresponding variables exist. The variable CONVERSION, for instance, may be assigned the values 1 2 1, or it may be assigned the variable MODEL1 merely by typing either of these responses. If the input is to be a vector, enter all values on the same line, each one separated by a space, and the whole thing terminated with a <cr>. When changing the value of the variable NAME, which gives the name of the X file being created or used, the new name must be entered in quotes:

ENTER NEW VALUE(S)   'ZORCH'

When the parameter values are correct for a particular entry, invoke the 'Write params' option to write them to the command file by typing "W" <cr>. Writing more times than specified in the file create mode will produce a DISK FULL error, and kill the program. To start anew, execute the command ')LOAD DOLPHINS' once again.

Typing "Q" <cr> closes the command file, and returns control to the topmost level of the program. At this point, a "C" <cr> will load the XS workspace and create any number of X files, and an "R" will load the XP workspace and process any number of X files in any number of ways. See Appendix II for a sample session.

To edit input, backspace to the point of the error, enter a <line feed>, and then proceed to type in the correct input.

APL WORKSPACES

NAME	FUNCTION
DOLPHINS	Interactive command processing program with prompts for guiding user through use of programs.
XS	Produces X matrix in datafile on disk. This involves the calculating mean lengths for all ages to be considered, calculating a normal distribution about each mean value, and determining what percentage at each age are in which length group given this normal distribution.
XP	Runs calculations to determine the portion of a population in the length specified groups. This involves the construction of a population structure (CX), modification of the CX to reflect periodicity in births, and the matrix multiplication of the PI vector (the internal representation of the CX) against the X matrix. The workspace also performs output functions, including a number of options in graphical and numerical output.

FUNCTIONS IN WORKSPACE XS

LEFT ARG	FN	RIGHT ARG	DESCRIPTION
	AREA	PTS	Given a vector in standard normal form, AREA returns a vector one item larger than the input vector which gives the area under a standard normal curve between the points given in PTS which are less than 3 standard deviations distant from the mean, and between + and - infinity.
	C		A C(ontrol) function which reads in the parameters for a particular run of XS from a command file specified in the 'DOLPHINS' workspace.
	LAYEARS	LAYERS	Converts age in years into age in dental layers, for the function LENS. Uses the golbal variables CONVERSION which are: 1. Initial rate layers/years 2. Cutoff age between rates 3. Final rate layers/year
	LEN	AGE	Uses Laird-Gompertz model to convert animal age in layers into rough length estimates in centimeters. Uses golbal variable COEFF which provides the coefficients for the model.
	MAREA	MAT	Performs the function AREA on each row of matrix mat. In this capacity MAREA forms a portion of the X matrix (transposed); for MAT, each row represents an age class, and area function determines what portion of each row/age class is in each length interval. Here, MAT has been formed by the command 'LENS STAND INCS' which creates a matrix whose rows are the distance in standard deviations between the mean lengths for each row (LENS [I] and the boundary values for each length group (INCS [J].
	NORM	X	Returns for all X the area between X and infinity under a standard normal curve, where X is the number of standard deviations from the mean (which is 0 in standard normal form).
	RECDATA		Initializes an X matrix file with the parameters which produced the file.

FUNCTIONS IN WORKSPACE XS - Continued

LEFT ARG	FN	RIGHT ARG	DESCRIPTION
BYTES	SIZEOF	LIMS	Determines the storage space necessary for a data item, where BYTES is the size of one unit, say a character, in bytes, and LIMS is the size of the data item in units.
MU	STAND	PTS	Converts the rows of a matrix into standard normal form, where MU[I] is the main value for the Ith row, and PTS are the points to be converted (a vector of points). The function performed is: Point = (point - mean) / 2.
	XSET		Constructs an X matrix as defined by Clark (1981) where X (I, J) gives the fraction of age group J at length I. Because of the massive proportions of matrix, XSET constructs it in AGES [2] 2 year pieces (5 is an optimum size). Important variables: a. INCS: INCS are the length intervals-2 where the missing intervals are (0 to INCS [1]) and (INCS [N] to infinity). b. FSZ: The size of the 'X' file to be made. c. Lengths: mean animal lengths given by Laird-Gompertz model from AGE vector (0 .1 .2...). d. CV: Current coefficient of variation among animal lengths.

FUNCTIONS IN WORKSPACE XP

LEFT ARG	FN	RIGHT ARG	DESCRIPTION
	C		A C(ontrol) function which reads in the parameters for a particular run of XS from a command file specified in the 'dolphins' workspace.
	CHARMAT	V	Makes a vector of characters into a matrix with one word per row.
U	GETFNS	F	Copies of list of functions F from the copy library of disk U.
LIMS	HIST	PARTS	Makes histograms. LIMS is as follows: 1. Number of vertical divisions 2. Maximum value Parts as follows: 1. File the number 2. Maximum animal age If LIMS [3] = 1 the histogram will be divided into pieces according to what age class the lengths come from, with differing symbols for each age class. Otherwise, the graph will be solid.
	INFO		Prints out information about the current run.
	LFREQ		Oversees the processing of the X matrix and the population vector. The CYCLE loop simulates the seasonality or lack of it in reproduction.
	PRINTX		Prints out the matrix X (I,J)(fraction of age class J in length interval I) or the X matrix multiplied by the population structure vector (fraction of population in age class J and length interval I).
	READATA		Reads data from X file.
	XPRINT		Prints the length frequencies of the total population, and oversees the printing and formatting of the graph, if there is to be one.

FUNCTIONS IN WORKSPACE XP - Continued

LEFT ARG	FN	RIGHT ARG	DESCRIPTION
	XPROC		Performs the creation of a population structure, including calculation of lotka's equations for beta, lambda, lx, and cx. The population structure CX is modified to conform to the dimension of the X matrix and to reflect the requested seasonality of reproduction and the spread factor for births.

FUNCTIONS IN WORKSPACE DOLPHINS

LEFT ARG	FN	RIGHT ARG	DESCRIPTION
	CREATE		Creates an X file by loading the XS workspace which then copies the passed parameters and creates the file.
	DISPLAY		Displays the current values of the global variables.
	INPN	OUT	Displays OUT and returns numerical unput.
	INPN	OUT	Displays OUT and returns character input.
	LISTV	NAMES	Does the dirty work for (DISPLAY), showing the variables names followed by their values.
	MAKE		Creates a command file.
	QUIT		Closes the command file created, if any.
	RUN		Loads and runs the XP workspace, generating length frequencies and requested output.
	SET		Oversees the 'set mode' in which variables are examined, changed, and written to command files.
	WRITE		Writes parameter values to command file.
	Z		Oversees the 'command mode' from which the user may call either the XS for XP workspaces, or may enter 'set mode' and create a command file.

GLOBAL VARIABLES

NAME	SHAPE	DESCRIPTION
AGES	2	1. The number of years to be put into an X matrix file. By the tenth year spinner lengths are all the same, so this cutoff allows a big savings on time and space. When the maximum age considered is greater than this cutoff point, the program lumps the last LIMS[1]-AGES[1] years together. Note: A value of 11 for ages [13] is ten years by themselves plus an eleventh year which is really the remaining years combined. 2. Number of age groups per year. Should be divisible by 2. Default value is 10.
CLX	3	1. and 2. Coefficients of LX survivorship equation. 3. Survivorship through first year with LX coefficients given.
COEFF	7	Coefficients for two-step Laird-Gompertz growth model. Default values are those for juvenile and female adult spinners.
CONVERSION	3	1. Initial number of growth layers per year. 2. Cut-off point in years between initial and final values. 3. Terminal number of growth layers per year.
DRIVES	3	1. Disk drive which has X matrix files. 2. Disk drive which has the needed external functions. 3. Disk drive for temporary file.
CVS	3	1. Initial coefficient of variation. 2. Final coefficient of variation. 3. Cutoff year between.
LENLIMS	3	1. Minimum animal length checked for. 2. Number of length groups. 3. Size, in cm., of length groups.

GLOBAL VARIABLES - Continued

NAME	SHAPE	DESCRIPTION
LIMS	3	1. Maximum animal age. 2. Number of reproductive pulses per year. 3. Fraction born at pulse. If this fraction is not unity, half of the remaining fraction will be born one time division (1 AGES [2] years) earlier than pulse, the other half will be born one period later. 4. R, the instantaneous rate of population change.
OPT	3	1. No graph = 1. Solid bar graph = 2. Divided bar graph = 3. 2. Output no data = 1. Output X matrix = 2. Output X matrix multiplied by PI vector = 3. 3. Number of years to output of above, starting from year 0.

VARIABLES SPECIFIC TO DOLPHINS WS

NAME	SHAPE	DESCRIPTION
CLX1	3	Coefficients for LX equation giving first year survivorship of .66.
CLX2	3	Same, giving first year survivorship of .77.
CLX3	3	Same, giving first year survivorship of .81.
CLX4	3	Same, giving first year survivorship of .69.
FSPINNER	7	Coefficients for Laird-Gompertz growth model using data for juvenile and female spinners.
MSPINNER	7	Same, using data for juvenile and male adult spinner dolphins.
MODEL1	3	Coefficients for years to layers conversion assuming 1.5 layers for the first year, and 1 per year thereafter.
MODEL2	3	Same, assuming 1 per year until dental occlusion.

DEFAULT VALUES OF VARIABLES

NAME	VALUES
AGES	11 10
CLX	0.8 0.1 0.66
CLX1	0.8 0.1 0.66
CLX2	0.4 0.1 0.77
CLX3	0.4 0.05 0.81
CLX4	0.8 0.05 0.69
COEFF	77 0.663 0.9098 4.11 156.85 0.0546 0.6354
CONVERSION	1.5 1 1
CVS	0.05 0.05 0
DRIVES	0 2 0
FSPINNER	77 0.663 0.9098 4.11 156.85 0.0546 0.6354
LENLIMS	55 35 5
LIMS	30 2 0.5 0
MODEL1	1.5 1 1
MODEL2	1 1 1
MSPINNER	77 0.663 0.9098 4.11 156.85 0.0507 0.3765
NAME	
OPT	3 3 5

LITERATURE CITED

Clark, W.G. 1981. Restricted least-squares estimates of age composition from length composition. Can. J. Fish. Aquat. Sci. 38:297-307.

Eberhardt, L.L., and D.B. 1977. Population dynamics and marine mammal management policies. J. Fish. Res. Board Can. 34:183-190.

Keyfitz, N. 1968. Introduction to the mathematics of populations. Addison-Wesley Publ. Co., Reading, Mass. 450 pp.

Perrin, W.F., D.B. Holts, and R.B. Miller. 1977. Growth and reproduction of the eastern spinner dolphin, a geographical form of *Stenella longirostris* in the eastern tropical Pacific. Fish. Bull. (U.S.) 4:725-750.

Perrin, W.F., J.M. Coe, and J.R. Zweifel. 1976. Growth and reproduction of the eastern spotted porpoise, *Stenella attenuata*, in the offshore eastern tropical Pacific. Fish. Bull. (U.S.) 2:229-269.

APPENDIX I
SAMPLE SESSION WITH DOLPHINS WORKSPACE

Note: Program Output is underlined.
<cr>= carriage return

```
C)REATE X FILE, R)UN CALCULATIONS, S)ET PARAMETERS        S  <cr>

C)HANGE, D)ISPLAY, M)AKE FILE, Q)UIT, W)RITE PARAMS       D  <cr>

CURRENT VALUES OF VARIABLES

AGES:  11 10
CLX:  0.8 0.1 0.66
COEFF:  77 0.663 0.9098 4.11 156.85 0.0546 0.6354
CONVERSION:  1.5 1 1
CVS:  0.05 0.05 0
DRIVES:  1 1 2
FN:
LENLIMS:  55 36 5
LIMS:  30 10 0.5 0
NAME:
OPT:  3 3 5

C)HANGE, D)ISPLAY, M)AKE FILE, Q)UIT, W)RITE PARAMS       C  <cr>

CHANGE WHAT VARIABLE?      NAME  <cr>
INPUT NEW VALUE (S)       'XFIL'  <cr>

C)HANGE, D)ISPLAY, M)AKE FILE, Q)UIT, W)RITE PARAMS       C  <cr>

CHANGE WHAT VARIABLE?      DRIVES  <cr>
INPUT NEW VALUE(S)         0 2 0  <cr>

C)HANGE, D)ISPLAY, M)AKE FILE, Q)UIT, W)RITE PARAMS       M  <cr>

NAME OF COMMAND FILE?      COM  <cr>
NUMBER OF ENTRIES?         1  <cr>

C)HANGE, D)ISPLAY, M)AKE FILE, Q)UIT, W)RITE PARAMS       W  <cr>

C)HANGE, D)ISPLAY, M)AKE FILE, Q)UIT, W)RITE PARAMS       D  <cr>

CURRENT VALUES OF VARIABLES

AGES:  11 10
CLX:  0.8 0.1 0.66
COEFF:  77 0.663 0.9098 4.11 156.85 0.0546 0.6354
CONVERSION:  1.5 1 1
CVS:  0.05 0.05 0
DRIVES:  0 2 0
FN:  COM
LENLIMS:  55 36 5
LIMS:  30 10 0.5 0
```

NAME: XFIL
OPT: 3 3 5
C)HANGE, D)ISPLAY, M)AKE FILE, Q)UIT, W)RITE PARAMS   Q <cr>

C)REATE X FILE, R)UN CALCULATIONS, S)ET PARAMETERS   C <cr>

Note: The program Dolphins now loads workspace XS and runs Xset, which will create an X file automatically, using the parameters just set. When Xset finishes, it will load dolphins again. The command "R)un calculations" will cause dolphins to load workspace XP and run XPROC. The run in Appendix II. is the result of the set of parameters given here and the sequence of actions described.

APPENDIX II

SAMPLE RUN

LENGTH FREQUENCY DISTRIBUTION AT 0 YEARS

INTERVALS	(TN CM)	PERCENTAGE OF POPULATION
50	55	0.000
55	60	0.000
60	65	0.000
65	70	0.067
70	75	0.549
75	80	1.201
80	85	1.276
85	90	1.293
90	95	1.299
95	100	1.323
100	105	1.355
105	110	1.402
110	115	1.466
115	120	1.563
120	125	1.742
125	130	2.082
130	135	2.640
135	140	3.369
140	145	4.266
145	150	5.500
150	155	7.032
155	160	9.211
160	165	11.768
165	170	13.256
170	175	11.977
175	180	8.200
180	185	4.137
185	190	1.522
190	195	0.409
195	200	0.081
200	205	0.000
205	210	0.000
210	215	0.000
215	220	0.000
220	225	0.000
225	230	0.000

****

Number of cohorts per year..............	10
Percentage of cohort on time............	100.000000
Maximum animal age (in years)...........	10.000000
Coefficient of variation of lengths.....	0.050000
Up to 0.000 years, final cv.............	0.050000
Survivorship through first year.........	0.660000
Intrinsic rate of population increase...	0.000000
Dental layers per year (initial)........	1.500000
Up to 1,000 years, final rate...........	1.000000

Coefficients for Laird-Gompertz Model:

77.0000	0.6630	0.9098	4.1100	156.8500	0.0546	0.6354

Dolphin length distribution histogram legend

Each symbol corresponds to 0.5 percent of total population

Dolphins up to 0.5 years................	□
Dolphins at 0.6 years to 1.0 years......	θ
Dolphins at 1.1 years to 1.5 years......	*
Dolphins at 1.6 years to 2.0 years......	[
Dolphins at 2.1 years to 2.5 years......	≠
Dolphins over 2.6 years.................	+

****

```
          DOLPHIN LENGTH DISTRIBUTIONS
            PERCENTAGE OF POPULATION
              0                             17.5
              +---------+---------+---------+---
     50   55|
     55   60|
     60   65|
     65   70|
     70   75|□
L    75   80|□□
E    80   85|□□□
N    85   90|□□□
G    90   95|□□□
T    95  100|□□□
H   100  105|□□⊙
    105  110|□⊙⊙
I   110  115|⊙⊙⊙
N   115  120|⊙⊙⊙
T   120  125|⊙⊙⊙
E   125  130|⊙⊙**
R   130  135|⊙***]
V   135  140|⊙***]]◦
A   140  145|**]]]≠≠◦◦
L   145  150|*]]]≠≠++◦◦◦
S   150  155|]]≠≠++◦◦◦◦◦◦◦◦
    155  160|]≠++◦◦◦◦◦◦◦◦◦◦◦◦◦◦
I   160  165|≠+◦◦◦◦◦◦◦◦◦◦◦◦◦◦◦◦◦◦◦◦◦◦
N   165  170|+◦◦◦◦◦◦◦◦◦◦◦◦◦◦◦◦◦◦◦◦◦◦◦◦◦◦
    170  175|◦◦◦◦◦◦◦◦◦◦◦◦◦◦◦◦◦◦◦◦◦◦◦◦
C   175  180|◦◦◦◦◦◦◦◦◦◦◦◦◦◦◦◦
M   180  185|◦◦◦◦◦◦◦◦
    185  190|◦◦◦
    190  195|◦
    195  200|
    200  205|
    205  210|
    210  215|
    215  220|
    220  225|
    225  230|
```

APPENDIX III

LISTINGS OF FUNCTIONS

FUNCTIONS IN DOLPHINS WORKSPACE

```
      ∇CREATE
[10]  ⍎')LOAD XS'
[20]  →0
      ∇


      ∇DISPLAY;DUMMY
[10]  'CURRENT VALUES OF VARIABLES',⎕V[14]
[20]  LISTV'AGES CLX COEFF CONVERSION CVS DRIVES FN LENLIMS LIMS N
      AME OPT'
[30]  ⎕V[14]
[40]  DUMMY←INPS'READY?
[50]  ⎕V[13]
      ∇


      ∇NUM←INPN OUT
[10]  NUM←⍎⍞,0⍴⍞←OUT
      ∇


      ∇STR←INPS OUT
[10]  STR←⍞,0⍴⍞←OUT
      ∇


      ∇LISTV NAMES;NAAME;I;PTR;N
[10]  ⍝LISTS VARIABLES NAMED WITH THEIR VALUES.
[20]  ⍝MUST BE AT LEAST TWO VARIABLES, AND MUST NOT HAVE SPACE AFT
      ER LAST.
[30]  PTR←(NAMES=' ')/⍳⍴NAMES
[40]  N←⍴PTR
[50]  I←0
[60]  LOOP:NAAME←((I≠0)×PTR[I+I=0])↓(((⍴NAMES)×I=N)+(I≠N)×¯1+PTR[I
      +(I≠N)])↑NAMES
[70]  NAAME,' : ';⍎NAAME
[80]  →(N≥I←I+1)/LOOP
      ∇


      ∇MAKE;N
[10]  FN←INPS'NAME OF COMMAND FILE?
[20]  N←INPN'NUMBER OF ENTRIES?
[30]  TIE←FN FMAKE ⎕U,2+N×10
[40]  ⍎')CSAVE FN'
[50]  ⎕V[13]
      ∇


      ∇QUIT
[10]  →(TIE=0)/0
[20]  FUNTIE TIE
[30]  ⎕V[13]
      ∇
```

FUNCTIONS IN DOLPHINS WORKSPACE

```
      ∇RUN
[10] ⍎')LOAD XP'
[20] →0
      ∇


      ∇SET;N;TIE;FN;VAR
[10] FN←⍳TIE←0
[20] PROMPT:N←'CDMQW'⍳INPS'C)HANGE, D)ISPLAY, M)AKE FILE, Q)UIT,
      W)RITE PARAMS  ';⎕V[13]
[30] →(1≠⍴N)/PROMPT
[40] →((N=1),(N=2),(N=3),(N=4),(N=5),(N=6))/C,D,M,Q,W,PROMPT
[50] C:VAR←INPS'CHANGE WHAT VARIABLE?
[60] VAR←⍎VAR,'←',INPS'INPUT NEW VALUE(S)
[70] ⎕V[13]
[80] →PROMPT
[90] D:DISPLAY
[100] →PROMPT
[110] M:MAKE
[120] →PROMPT
[130] Q:QUIT
[140] →0
[150] W:WRITE
[160] →PROMPT
      ∇


      ∇WRITE
[10] AGES FAPPEND TIE
[20] COEFF FAPPEND TIE
[30] CONVERSION FAPPEND TIE
[40] CVS FAPPEND TIE
[50] DRIVES FAPPEND TIE
[60] LENLIMS FAPPEND TIE
[70] NAME FAPPEND TIE
[80] CLX FAPPEND TIE
[90] LIMS FAPPEND TIE
[100] OPT FAPPEND TIE
[110] ⎕V[13]
      ∇
```

FUNCTIONS IN DOLPHINS WORKSPACE

```
       ∇Z;N
[10] ⎕V[13]
[20] PROMPT:N←'CRS'⍳INPS'C)REATE X FILE, R)UN CALCULATIONS, S)ET
       PARAMETERS    ';⎕V[13]
[30] →(1≠⍴N)/PROMPT
[40] →((N=1),(N=2),(N=3),(N=4))/C,R,S,PROMPT
[50] C:CREATE
[60] →PROMPT
[70] R:RUN
[80] →PROMPT
[90] S:SET
[100] →PROMPT
       ∇
```

FUNCTIONS IN WORKSPACE XS

```
      ∇SPACE←AREA PTS;POS;NEG;LOW;HIGH
[10]  ⍝GIVES AREA UNDER STANDARD NORMAL CURVE BETWEEN ADJACENT ELEMENTS
[20]  ⍝OF VECTOR (MUST BE IN ASCENDING ORDER) AND BETWEEN ENDS AND +/-
       INFINITY
[30]  PTS←(LOW←+/PTS≤¯3)↓PTS
[40]  PTS←(HIGH←-+/PTS≥3)↓PTS
[50]  SPACE←NORM PTS
[60]  POS←(PTS≥0)/SPACE
[70]  NEG←(PTS<0)/SPACE
[80]  POS←0.5-POS
[90]  NEG←NEG-0.5
[100]  SPACE←(SPACE,0.5)-¯0.5,SPACE←NEG,POS
[110]  SPACE←(LOW⍴0),SPACE,(-HIGH)⍴0
      ∇
```

```
      ∇C;CFP;DRIVES;NAME;CVS;AGES;DUM;TYE;LENLIMS;FN;COEFF;CONVERSION
[10]  ⍎')ERASE DESCRIBE'
[20]  ⍎')COPY FN'
[30]  TYE←FN FTIE ⎕U
[40]  CFP←0
[50]  READ:→EOF/OUT
[60]  AGES←FREAD TYE,CFP←CFP+1
[70]  COEFF←FREAD TYE,CFP←CFP+1
[80]  CONVERSION←FREAD TYE,CFP←CFP+1
[90]  CVS←FREAD TYE,CFP←CFP+1
[100]  DRIVES←FREAD TYE,CFP←CFP+1
[110]  LENLIMS←FREAD TYE,CFP←CFP+1
[120]  NAME←FREAD TYE,CFP←CFP+1
[130]  CFP←CFP+3
[140]  DUM←FREAD TYE,CFP+1
[150]  XSET
[160]  →READ
[170]  OUT:EOF←0
[180]  ⍎')LOAD DOLPHINS'
      ∇
```

```
      ∇YEARS←LAYEARS LAYERS;A;B
[10]  ⍝CONVERT AGE IN YEARS TO AGE IN LAYERS,
[20]  A←(LAYERS≤CONVERSION[2])/LAYERS
[30]  B←(LAYERS>CONVERSION[2])/LAYERS
[40]  YEARS←(A×CONVERSION[1]),CONVERSION[3]×B+CONVERSION[1]-1
      ∇
```

FUNCTIONS IN WORKSPACE XS

```
      ∇LENGTH←LEN AGE;ADULT;JUVENILE
[10]  ⍝CONVERTS ANIMAL AGE IN LAYERS INTO ROUGH LENGTHS IN CM.
[20]  ⍝USES LAIRD-GOMPERTZ MODEL WITH DATA (COEFF=COEFFICIENTS) FROM
[30]  ⍝PERRIN (1977).
[40]  JUVENILE←(AGE≤COEFF[4])/AGE
[50]  ADULT←(AGE>COEFF[4])/AGE
[60]  LENGTH←COEFF[1]×*((COEFF[2]÷COEFF[3])×1-*((-COEFF[3])×JUVENILE))
[70]  LENGTH←LENGTH,COEFF[5]×*((COEFF[6]÷COEFF[7])×1-*((-COEFF[7])×ADUL
       T-COEFF[4]))
      ∇


      ∇AREAS←MAREA MAT;CNT;DIMS
[10]  ⍝CONSTRUCTS A MATRIX FROM THE APPLICATION OF THE FUNCTION 'AREA'
       TO MAT.
[20]  AREAS←⍳0
[30]  CNT←(DIMS←⍴MAT)[1]
[40]  LOOP:AREAS←AREAS,AREA DIMS[2]↑,MAT
[50]  MAT←DIMS[2]↓,MAT
[60]  →LOOP×⍳0<CNT←CNT-1
[70]  AREAS←(DIMS[1],DIMS[2]+1)⍴AREAS
      ∇


      ∇AREA←NORM X;B;T;FX
[10]  B←0.31938153 ¯0.356563782 1.781477937 ¯1.821255978 1.3302744
[20]  T←÷1+0.2316419×|X
[30]  FX←(÷(○2)*0.5)×*-((X*2)÷2)
[40]  T←T∘.*⍳5
[50]  AREA←FX×B INPROD(TRANSPOSE T)
      ∇


      ∇RECDATA
[10]  FSZ FAPPEND TIE
[20]  INCS FAPPEND TIE
[30]  CVS FAPPEND TIE
[40]  AGES FAPPEND TIE
[50]  LENLIMS FAPPEND TIE
[60]  COEFF FAPPEND TIE
[70]  CONVERSION FAPPEND TIE
      ∇


      ∇VALS←MU STAND PTS
[10]  ⍝CREATE MATRIX IN STANDARD NORMAL FORM.
[20]  VALS←-MU∘.-PTS
[30]  VALS←VALS÷TRANSPOSE((⍴VALS)[2],⍴,MU)⍴MU×CV
      ∇
```

FUNCTIONS IN WORKSPACE XS

```
      ∇BLOCKS←BYTES SIZEOF LIMS
[10]  BLOCKS←⌈0.5+(BYTES×LIMS)÷256
      ∇


      ∇XSET;U;CV;LENGTHS;TIE;CUT;INCS;LIM;CNT;N;LENS;FSZ
[10]  ⍝THIS FUNCTION CONSTRUCTS AN X MATRIX AS DEFINED BY CLARK (1981)
[20]  ⍝WHERE X[I,J] GIVES PERCENTAGE OF AGE GROUP J AT LENGTH I
[30]  INCS←LENLIMS[1]+LENLIMS[3]×0,⍳¯1+LENLIMS[2]-1
[40]  FSZ←9+(6 SIZEOF N)+2×AGES[1]×6 SIZEOF 5×1+N←⍴INCS
[50]  TIE←NAME FCREATE DRIVES[1],(TIE←1),FSZ
[60]  LENGTHS←LEN LAYEARS(÷AGES[2])×0,⍳¯1+AGES[1]×AGES[2]
[70]  RECDATA
[80]  CUT←⌈AGES[2]÷2
[90]  LIM←(×/AGES)÷CUT+CNT←0
[100]  LOOP:LENS←CUT↑(CUT×CNT)↓LENGTHS
[110]  'PROCESSING YEAR NUMBER ';⌊0.5×CNT
[120]  CV←((CNT<CVS[3]),(CNT≥CVS[3]),0)/CVS
[130]  (MAREA LENS STAND INCS)FAPPEND TIE
[140]  →(LIM>CNT←CNT+1)/LOOP
[150]  FUNTIE TIE
      ∇
```

FUNCTIONS IN WORKSPACE XP

```
      ∇C;FN;FSZ;CFP;TYE;CLX;COEFF;DUM;CONVERSION;OPT;CVS;DRIVES;LIMS;N
      AME
[10]  ⍎')ERASE DESCRIBE'
[20]  ⍎')COPY FN'
[30]  TYE←FN FTIE ⎕U
[40]  CFP←0
[50]  READ:→EOF/OUT
[60]  CVS←FREAD TYE,CFP←CFP+4
[70]  DRIVES←FREAD TYE,CFP←CFP+1
[80]  NAME←FREAD TYE,CFP←CFP+2
[90]  CLX←FREAD TYE,CFP←CFP+1
[100] LIMS←FREAD TYE,CFP←CFP+1
[110] OPT←FREAD TYE,CFP←CFP+1
[120] DUM←FREAD TYE,CFP+1
[130] LFREQ
[140] →READ
[150] OUT:EOF←0
[160] ⍎')LOAD DOLPHINS'
      ∇


      ∇Z←CHARMAT V;A
[10]  V←(~A←V∊' ,')/V←V,' '
[20]  Z←(⍴A)⍴(,A∘.≥⍳⌈/0,A←(A≠0)/A←A-1+0,¯1↓A←A/⍳⍴A)\V
      ∇


      ∇U GETFNS F;O;N
[10]  ⎕U←U+0×O←⎕U+0×N←(⍴F←CHARMAT F)[1]
[20]  L:⍎')COPY ',F[N;]
[30]  →L×⍳0<N←N-1
[40]  ⎕U←O
      ∇
```

FUNCTIONS IN WORKSPACE XP

```
      ∇PLOT←LIMS HIST PARTS;UPTO;SCALE;INT;I;CUT;INCS;FP;PT;SCALE;OD
[10]  ⍝PRODUCES HISTOGRAMS FROM MODIFIED X FILE (TEMP) WITH DIFFERENT
[20]  ⍝SYMBOLS FOR EVERY CUT FRACTIONS OF A YEAR UP TO UPTO YEARS.
[30]  ⍝LIMS[1]←→NUMBER OF VERTICAL DIVISIONS, LIMS[2]←→MAX Y VALUE
[40]  SCALE←÷/2↑LIMS
[50]  →LIMS[3]/PIECES
[60]  PLOT←(⍴PLOT)⍴' ⎕'[1+,PLOT←(P←P×SCALE)∘.≥⍳LIMS[1]]
[70]  →END
[80]  →0
[90]  PIECES:CUT←2×UPTO←5+FP←OD←PT←PLOT←0
[100]  SCALE←÷/LIMS
[110]  LOOP:OD←OD++/TRANSPOSE FREAD PARTS[1],FP←FP+1
[120]  →LOOP×⍳(FP>2×UPTO+1)∧(FP<2×PARTS[2])
[130]  PLOT←PLOT+(PT←PT+1)×(PLOT≠0)≠(⌊0.5+SCALE×OD)∘.≥⍳PARTS[2]
[140]  →LOOP×⍳(2×PARTS[2])≥FP
[150]  PLOT←(⍴PLOT)⍴' ⎕⊖*⌷≠+∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘∘'[1+,PLOT]
[160]  END:⍎')ERASE HIST',P←⍳0
      ∇


      ∇INFO OPT
[10]  FF,(70⍴'*'),CR
[20]  'NUMBER OF COHORTS PER YEAR ............',16 0⍕LIMS[2]
[30]  'PERCENTAGE OF COHORT ON TIME ..........',16 6⍕100×LIMS[3]
[40]  'MAXIMUM ANIMAL AGE (IN YEARS) .........',16 6⍕AGES[2]
[50]  'COEFFICIENT OF VARIATION OF LENGTHS ..',16 6⍕CVS[1]
[60]  'UP TO ',(7 3⍕CVS[3]),' YEARS,  FINAL CV ........',16 6⍕CVS[2]
[70]  'SURVIVORSHIP THROUGH FIRST YEAR .......',16 6⍕CLX[3]
[80]  'INTRINSIC RATE OF POPULATION INCREASE ',16 6⍕LIMS[4]
[90]  'DENTAL LAYERS PER YEAR (INITIAL) ......',16 6⍕CONVERSION[1]
[100]  'UP TO ',(7 3⍕CONVERSION[2]),' YEARS,  FINAL RATE .....',16 6⍕CO
      NVERSION[3]
[110]  CR,'COEFFICIENTS FOR LAIRD-GOMPERTZ MODEL:',(2⍴CR),9 4⍕COEFF
[120]  CR,(70⍴'-'),CR
[130]  →OPT/BR,0⍴⍞←'DOLPHIN LENGTH DISTRIBUTION HISTOGRAM LEGEND'
[140]  →END,0⍴⍞←(2⍴CR),'⎕ = 0.5 PERCENT OF DOLPHIN POPULATION',CR
[150]  BR:(2⍴CR),'EACH SYMBOL CORRESPONDS TO 0.5 PERCENT OF TOTAL POPUL
      ATION'
[160]  CR,'DOLPHINS UP TO 0.5 YEARS ................ ⎕'
[170]  'DOLPHINS AT 0.6 YEARS TO 1.0 YEARS ..... ⊖'
[180]  'DOLPHINS AT 1.1 YEARS TO 1.5 YEARS ..... *'
[190]  'DOLPHINS AT 1.6 YEARS TO 2.0 YEARS ..... ⌷'
[200]  'DOLPHINS AT 2.1 YEARS TO 2.5 YEARS ..... ≠'
[210]  'DOLPHINS OVER 2.6 YEARS ................. +'
[220]  END:⍎')ERASE INFO',0⍴⍞←CR,70⍴'*'
      ∇
```

FUNCTIONS IN WORKSPACE XP

```
       ∇LFREQ;P;CR;FF;INFO;AGE;FP;FN;SKIP;TIE;TIE2;I;INCS
[10]  FF←⎕V[13],0⍴CR←⎕V[14]
[20]  TIE←NAME FTIE DRIVES[1]
[30]  READATA
[40]  DRIVES[2]GETFNS'INFO'
[50]  INFO|OPT[1]-2;0⍴⎕H←1+I←TIE2←0
[60]  ⎕H←0
[70]  ⍎')ERASE INFO'
[80]  AGE←(÷LIMS[2])×0,¯1↓⍳LIMS[2]×LIMS[1]
[90]  CYCLE:→NOPE×⍳SKIP←∧/OPT[1 2]≠3
[100]  'TEMP'FCREATE DRIVES[3],(TIE2←2),FSZ
[110]  NOPE:XPROC
[120]  DRIVES[2]GETFNS'XPRINT'
[130]  ⎕H←1
[140]  XPRINT
[150]  ⎕H←0
[160]  →SKIP/BR
[170]  FUNTIE TIE2
[180]  BR:AGE←AGE+÷AGES[2]
[190]  →CYCLE×⍳(÷LIMS[2])>I←I+÷AGES[2]
[200]  FUNTIE TIE
       ∇


       ∇PRINTX;T;TIENO;LIM;B;CNT
[10]  TIENO←(TIE,TIE2)[T←OPT[2]-1]
[20]  FP←(7,0)[T]
[30]  LIM←2×OPT[3]+CNT←0
[40]  LOOP:FF;I;' YEARS INTO REPRODUCTIVE CYCLE'
[50]  'COHORTS AGED ';CNT÷2;' TO ';0.4+CNT÷2;' YEARS'
[60]  →(T=2)/BR
[70]  'PERCENTAGE OF COHORTS WITHIN LENGTH INTERVALS';CR
[80]  →BR2
[90]  BR:'PERCENTAGE OF TOTAL POPULATION WITHIN LENGTH INTERVALS';CR
[100]  BR2:'      AGE →',(10⍴' '),10 1⍕(CNT÷2)+0,0.1×⍳4
[110]  '      ↓ LENGTH INTERVALS ';CR
[120]  (10 0↑TRANSPOSE INTS),10 4↑TRANSPOSE 100×FREAD TIENO,FP←FP+1
[130]  →LOOP×⍳LIM>CNT←CNT+1
[140]  ⍎')ERASE PRINTX'
       ∇
```

FUNCTIONS IN WORKSPACE XP

```
      ∇READATA
[10]  FSZ←FREAD TIE,1
[20]  INCS←FREAD TIE,2
[30]  CVS←FREAD TIE,3
[40]  AGES←FREAD TIE,4
[50]  LENLIMS←FREAD TIE,5
[60]  COEFF←FREAD TIE,6
[70]  CONVERSION←FREAD TIE,7
      ∇


      ∇XPROC;DIST;L;CX;D;LX;CNT;CUT;MX;PI
[10]  CX←(L×LX)÷+/(L←(*LIMS[4])*-AGE)×LX←*(CLX[1]×*-0.5×AGE)-CLX[1]÷CLX
      [2]×AGE
[20]  CX←(D↑CX),+/[1]((1+LIMS[1]-AGES[1]),LIMS[2])⍴(D←LIMS[2]×AGES[1]-1
      )↓CX
[30]  CX←(0=(AGES[2]÷LIMS[2])|¯1+⍳×/AGES)\CX
[40]  DIST←((LIMS[3]≠1)×(1-LIMS[3])÷2)×CX←(⍴CX)⍴((10×I)⍴0),CX
[50]  CX←(1↓DIST,0)+(0,¯1↓DIST)+CX×LIMS[3]
[60]  CUT←5+P←CNT←0×FP←7
[70]  LOOP:PI←CUT↑(CUT×CNT)↓CX
[80]  PI←TRANSPOSE((1+⍴INCS),⍴PI)⍴PI
[90]  MX←PI×FREAD TIE,FP←FP+1
[100]  →SKIP/NO
[110]  MX FAPPEND TIE2
[120]  NO:P←P++/[1]MX
[130]  →LOOP×⍳(2×AGES[1])>CNT←CNT+1
      ∇
```

FUNCTIONS IN WORKSPACE XP

```
      ∇XPRINT;PRINTX;R;HEAD;SIDE;TOP;SP1;SP2;A;B;C;FP;INC;INTS;HIST;PL
      OT
[10]  FP,'LENGTH FREQUENCY DISTRIBUTION AT ';I;' YEARS',CR
[20]  '           INTERVALS (IN CM)    PERCENTAGE OF POPULATION',CR
[30]  INTS←(INCS[1]-INC),INCS,INCS[ρ,INCS]+INC←INCS[2]-INCS[1]
[40]  INTS←((FP←2),¯1+ρINTS)ρ(¯1↓INTS),1↓INTS
[50]  (10 0⍕TRANSPOSE INTS),10 3⍕100×((ρP),1)ρP
[60]  →((OPT[2]=2)∨1=OPT[2])/CHOICE
[70]  DRIVES[2]GETFNS'PRINTX'
[80]  PRINTX
[90]  CHOICE:→(OPT[1]=0)/0
[100] DRIVES[2]GETFNS'HIST'
[110] PLOT←(A←35 0.175,OPT[1]-2)HIST TIE2,LIMS[1]
[120] INTS←5 0⍕TRANSPOSE INTS
[130] R←ρPLOT←INTS,(((SP2←(ρPLOT)[1]),1)ρ'|'),PLOT
[140] HEAD←(((ρINTS)[2]-6)ρ' '),(7 0⍕0),(((R[2]-(ρINTS)[2])-8)ρ' '),(7
       1⍕100×A[2])
[150] HEAD←(2,R[2])ρHEAD,((ρINTS)[2]ρ' '),'-+'[((9|0,⍳A[1])=0)+1]
[160] A←'DOLPHIN LENGTH DISTRIBUTIONS',0↑B←'PERCENTAGE OF POPULATION'
[170] C←'LENGTH INTERVALS IN CM',0↑SP1←R[2]
[180] TOP←((⌊(SP1-ρA)÷2)ρ' '),A,(⌈(SP1-ρA)÷2)ρ' '
[190] TOP←TOP,((⌊(SP1-ρB)÷2)ρ' '),B,(⌈(SP1-ρB)÷2)ρ' '
[200] SIDE←(SP2,1)ρ((⌊(SP2-ρC)÷2)ρ' '),C,(⌈((SP2←4+SP2)-ρC)÷2)ρ' '
[210] FP;PLOT←SIDE,(SP2,R[2])ρ(,TOP),(,HEAD),,PLOT
[220] ⍎')ERASE XPRINT'
      ∇
```