

NOAA Technical Memorandum ERL GLERL-62

A Method for Using a **CalComp** 9480 Digitizer Coupled
With Menu-Driven Software to Measure Amphipod Body
Length and Gut Fullness

Gregory A. Lang
Michael A. Quigley

Great Lakes Environmental Research Laboratory
Ann Arbor, Michigan
March 1987



UNITED STATES
DEPARTMENT OF COMMERCE

Malcolm Baldrige,
Secretary

NATIONAL OCEANIC AND
ATMOSPHERIC ADMINISTRATION

Anthony J. Calio,
Administrator

Environmental Research
Laboratories

Vernon E. Derr,
Director

NOTICE

Mention of a commercial company or product does not constitute an endorsement by NOAA/ERL. Use for publicity or advertising purposes, of information from this publication concerning proprietary products or the tests of such products, is not authorized.

For sale by the National Technical Information Service, 5285 Port Royal Road
Springfield, VA 22161

CONTENTS

	PAGE
ABSTRACT	1
1. INTRODUCTION	1
2. METHOD	2
2.1 Procedure for Using CRITTER	2
2.2 Total Body Length	4
2.3 Gut Fullness	6
2.3.1 Continuous gut	6
2.3.2 Segmented gut	6
2.3.3 Empty gut	6
2.3.4 Proceeding to new animal	6
2.4 Exit From CRITTER	7
3. APPLICATIONS	7
4. REFERENCES	8
Appendix A: Initial CRITTER Prompts, Explanations, and Example Responses	9
Appendix B: Example Output File Based on Information from Appendix A and Fig. 1	10
Appendix C: Source Code Listing of CRITTER	11

FIGURES

Figure 1. --Example of tracing sheet with CRITTER menu	3
Figure 2. --Image of Amphipod <u>Pontoporeia hoyi</u> indicating body length and gut content	4
Figure 3. --Flow chart showing steps for measuring amphipod body length and gut fullness using menu-driven software	5

A METHOD FOR USING A CALCOMP 9480 DIGITIZER COUPLED
WITH MENU-DRIVEN SOFTWARE TO MEASURE AMPHIPOD BODY
LENGTH AND GUT FULLNESS¹

Gregory A. Lang and Michael A. Quigley

ABSTRACT. A method for measuring amphipod body length and gut fullness, using a CalComp 9480 digitizer coupled with interactive, menu-driven software, is described. In addition to having a high degree of precision and accuracy, the digitizer method constitutes a rapid and efficient route for transferring raw length data from the microscope to an on-line data file. Procedure for use of the FORTRAN-77 software program CRITTER is fully detailed and a complete listing of the program is supplied.

1. INTRODUCTION

The study of dynamic processes in benthic invertebrate populations often involves the measurement of specific body parts. For instance, total body length is routinely used to estimate biomass, growth, and production (Cooper, 1965; Winberg, 1971; Winnell and White, 1984). In addition, gut fullness provides inferences about feeding frequencies relative to size class and season (Quigley, 1986). The credibility and value of these inferences depends greatly on the precision and accuracy of the length measurements as well as on the time and effort spent.

In conjunction with studies of population dynamics and feeding behavior of the benthic amphipod Pontoporeia hoyi, we have developed a method² that uses a CalComp 9480 Digitizer coupled with interactive, menu-driven software to measure and record amphipod body length and gut fullness. The method is more accurate and precise than more conventional methods. The coefficient of variation (a measure of precision) of the digitizer method is <1.2%, as determined by comparing replicate measurements of body length and gut fullness of 20 Pontoporeia hoyi. The accuracy of the digitizer method was determined by measuring a series of known-distance lines; regression results indicate that the measured values compare very well with the actual values ($r^2=0.9999$). Also, since inferences at the population level involve measurements taken on a large number of individuals, the method provides a rapid and efficient route for transferring raw length data from the microscope to an online data file with a minimum of intermediate steps. This report presents the digitizer method and provides the user with a step-by-step procedure for its use.

¹GLERL Contribution No. 548

²The digitizer method was developed and tested using the following hardware located at the Great Lakes Environmental Research Laboratory, Ann Arbor, Michigan: CalComp 9480 high resolution Digitizer with 12-button cursor, CalComp 9000 Digitizer Processing Unit, Anchor Automation Volksmodem 12 modem, and Lear Siegler ADM-3A CRT. Refer to the operator's manual (California Computer Products, Inc, 1983) for proper use of the CalComp equipment.

2. METHOD

Our method uses a CalComp 9480 high-resolution digitizer interfaced with a menu-driven, FORTRAN-77 program, CRITTER, to measure amphipod body length and gut fullness. The amphipod is placed under a dissecting microscope with a camera *lucida* attached. A camera *lucida* is a device that allows the user to view, through the microscope, the magnified image of the animal superimposed on the image of an 8.5" by 11" tracing sheet. The size of the projected image depends on the magnification of the microscope. The user then traces the image of the amphipod on the tracing sheet; a typical sheet may contain up to 30 such tracings. For reference, the individual tracings should be numbered in the order they are to be digitized (start number is arbitrary, numbers increasing by 1). A 1.75" by 11" margin along the right-hand side of the tracing sheet is reserved for the CRITTER menu consisting of four equal sized boxes containing the phrases END BODY, END SEGMENT, END GUT, and EXIT. An example tracing sheet is shown in Fig. 1.

The tracing sheet containing the amphipod images is placed on the tablet of the CalComp 9480 and each tracing is digitized with frequent entry of coordinate points. From this set of closely spaced points, CRITTER calculates total body length and gut fullness and records the results in an on-line computer file. Total body length of the amphipod is defined as the scalar distance along the amphipod's dorsal margin, beginning at the tip of the rostrum and ending at the base of the telson (Fig. 2). Gut fullness is defined as the length of gut (along the central axis of the gut) that contains food matter. Gut content is easily visible if the animal had been cleared in Amman's lactophenol. If the gut content is segmented, possibly because of intermittent feeding, then the total gut fullness is the sum of the individual segment lengths. The procedure for using CRITTER is described in sections 2.1-2.4 and a complete listing CRITTER's source code is contained in Appendix C.

2.1 Procedure for Using CRITTER

Once the digitizer is set up and initialized and the tracing sheet is properly oriented, the user executes CRITTER from the terminal interfaced with the CalComp 9480 Digitizer. CRITTER is interactive and will prompt the user for the name of the output file, a set of **descriptives**, and the x- and y-coordinates (in amphipod length units) of the upper right corner of the tracing sheet; the coordinates of the lower left corner are assumed to be (0,0). These coordinates are based on the ocular scale and magnification of the dissecting microscope. All subsequent digitized points will be assigned coordinate values relative to the scale defined by the four corners of the tracing sheet. Appendix A contains the initial prompts, explanations of the prompts, and typical user-supplied responses. The prompts appear on the terminal screen, and the responses are made using the terminal keyboard.

When the user has responded to the last prompt in **Appendix A**, the message "***** BEGIN MEASUREMENTS *****" appears on the screen and the user may begin digitizing the amphipod tracings. The user must measure each animal's total body length and gut fullness, in that order, before moving on to another animal. Moving from body length measurement to gut fullness measurement or from animal to animal is controlled by entering appropriate menu items.





















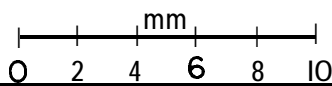
STA 21		June 15, 1985		of 5	
Station		Date		Replicate	
1		11		END BODY	
2		12			
3		13		END SEGMENT	
4		14			
5		15		END GUT	
6		16			
7		17		EXIT	
8		18			
9		19		EXIT	
10		20			
					

Figure 1.--Example of tracing sheet with CRITTER menu. The images represent 20 Pontoporeia hoyi collected at station 21 (Lake Michigan) on June 15, 1985.

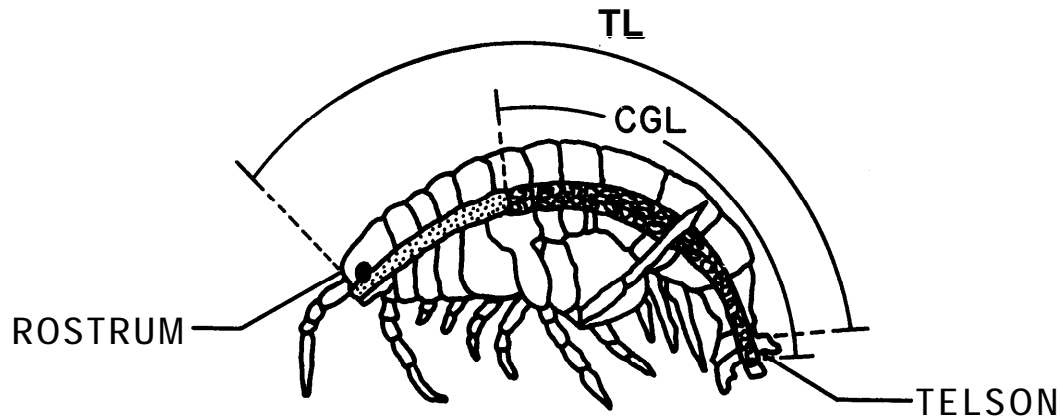


Figure 2.-- Image of Amphipod *Pontonoreia hoyi*. Total body length (TL) and continuous gut length (CGL) are indicated.

A menu item is entered by digitizing a point anywhere within the menu box. Throughout execution, CRITTER displays the animal number and section to be digitized (e.g., "BODY LENGTH - Animal No. 1"). The flow chart in Fig. 3 shows the steps for measuring amphipod body length and gut fullness using the menu-driven software program CRITTER.

2.2 Total Body Length

After "BODY LENGTH - Animal No. n" appears on the screen (where n refers to the number associated with the animal being measured), the user digitizes along the dorsal margin of the amphipod's image with the CalComp 9480 cursor, entering frequent coordinate points. Frequent is a relative term; the authors suggest that the user enter more points when digitizing around curves and less points when digitizing straight sections (30-50 points per animal should be adequate). CRITTER reads in the coordinate values, displays them on the screen for confirmation, and calculates the distance between successive data points. If an error occurs during a read (usually caused by computer line noise), three bells sound from the terminal and the following messages appear: "Bad Input Value - Possibly Line Noise - Reenter" and "Redo Body - Animal No. n." The user should then redigitize the body length of the current animal. Previous body length and gut fullness calculations will not be affected by this read error.

After the final set of points is entered, the user enters END BODY, instructing CRITTER to calculate the total body length of the current animal. Total body length is the sum of the individual lengths between successive data points. For confirmation, the terminal displays the animal number, the letter B denoting total body length, the resultant length measurement, and the number of digitized points that were used to determine the length.

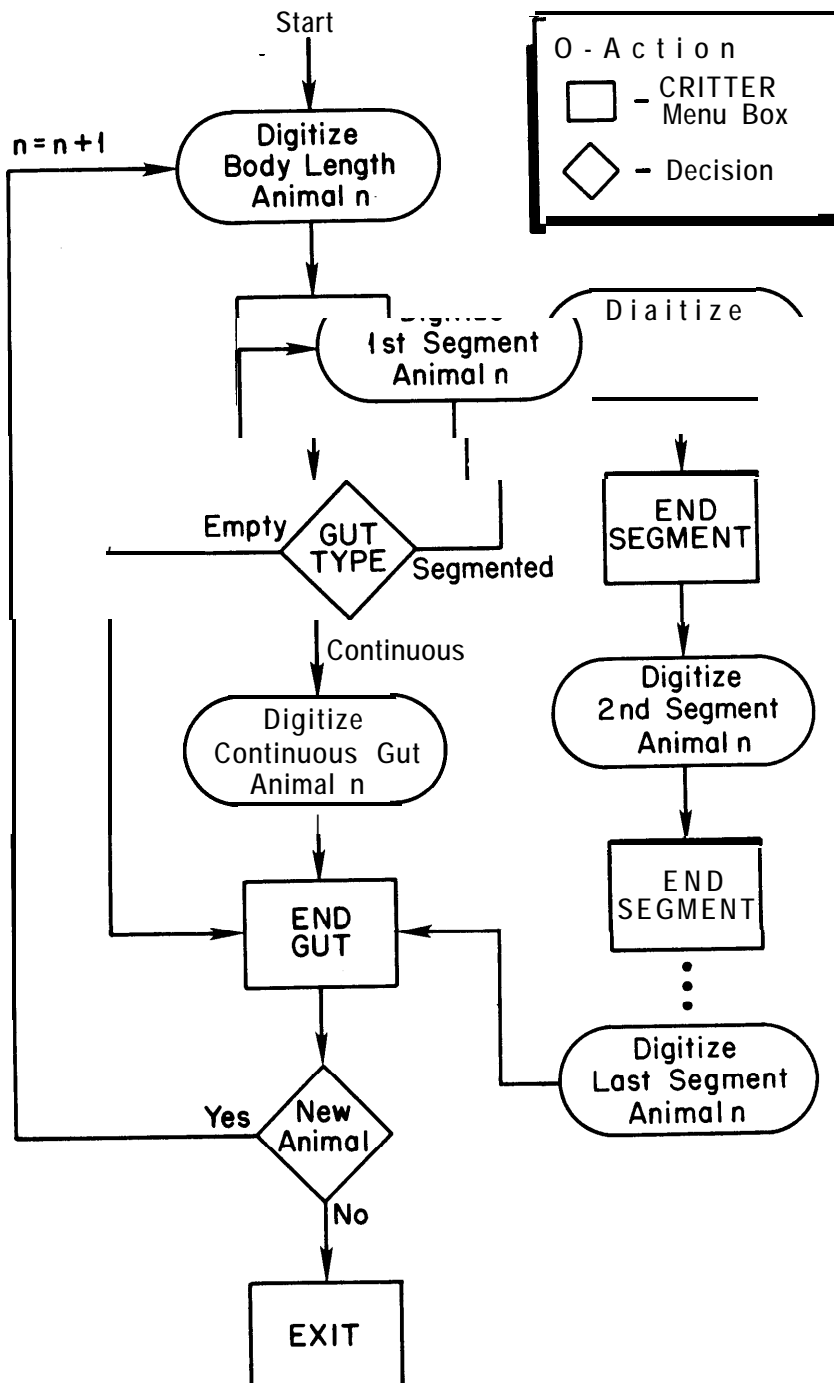


Figure 3. --Flow chart showing steps for measuring amphipod body length and gut fullness using menu-driven software.

2.3 Gut Fullness

After the body length is digitized and calculated, the following message appears on the screen: "GUT FULLNESS - Animal No. n." Three degrees of gut fullness are defined--continuous, segmented, and empty. A continuous gut is one in which the food matter is uninterrupted. A segmented gut is one in which the food matter is separated into distinct sections, possibly because of intermittent feeding. An empty gut contains no food matter.

2.3.1 Continuous gut

If the animal being measured has a continuous gut, the user digitizes along the central axis of the filled portion of the gut. As before, CRITTER reads in the coordinate values, displays them on the screen, and calculates the distance between successive data points. If an error occurs during a read, the gut content of the animal being measured must be redigitized. After the final set of points is entered, the user enters END GUT, instructing CRITTER to calculate the length of continuous gut. The terminal will display the animal number, the letter C corresponding to continuous gut length, the resultant length measurement, and the number of digitized points that were used to determine the length.

2.3.2 Segmented gut

If the animal being measured has a segmented gut, the user digitizes along the central axis of each segment, entering END SEGMENT between segments. After the final segment is digitized, the user enters END GUT, instructing CRITTER to calculate the total length of the segmented gut. Total segmented gut length is the sum of the individual segment lengths. If an error occurs during a read, the entire segmented gut of the animal being measured must be redigitized. When the calculations are complete, the terminal displays the animal number, the letter S denoting segmented gut, the resultant length measurement, and the number of digitized points.

2.3.3 Empty gut

If the animal being measured has an empty gut, the user enters END GUT without digitizing any coordinates. The terminal will display the animal number, the letter E denoting empty gut, the length measurement, and the number of digitized points. The length measurement and number of points will both be equal to zero.

2.3.4 Proceeding to a new animal

Upon completion of the gut fullness measurement the following message will appear on the screen: Body Length - Animal No. n (where n is now the number of the previously measured animal plus 1). The user is then ready to proceed to the next amphipod. Note, the menu items END BODY and END GUT may be used only once per animal and they may not be entered out of order. END SEGMENT is used as necessary. If, at any time following the measurement of body length or gut fullness, an improper menu box is entered, three bells will

sound and the message "Bad Menu Box • XXX" will appear on the screen, where XXX is the name of the bad menu box. This message will be followed by "Reenter Desired **Box**" and the user should enter the correct menu box. This will not affect the current or previous length calculations. For example, if the user enters END BODY after measuring an amphipod gut length, the above messages will appear and the user should then enter the correct box, END GUT, When the final amphipod has been measured the user is ready to terminate execution and store the results.

2.4 Exit From CRITTER

The user terminates execution of CRITTER by entering EXIT. This should be done only after an animal's gut fullness has been calculated and the message "BODY LENGTH • Animal No. n" has appeared (where n is now the number corresponding to the last animal plus 1). After EXIT is entered, the message "Enter 1 to Exit and Save, 2 to Continue, 3 to Quit (No Output Saved)" will appear on the screen. The user should respond using the terminal keyboard:

- 1: Instructs CRITTER to terminate execution and save the output file.
- 2: Instructs CRITTER to abort termination and continue calculating body length and gut fullness starting with animal number n.
- 3: Instructs CRITTER to terminate execution without saving the output file (i.e., the user wishes to cancel the entire session).

With a clean exit from CRITTER (option 1) the message "Output Saved in File: FILENAME" appears on the screen, where FILENAME is the name of the file the user entered at the start of execution. This file has a VAX file type OUT. The first three lines of the output file contain the station name, date, and replicate information as originally input. These **descriptives** are followed by the body length and gut fullness of each animal measured. An example output file, based on information in Appendix A and Fig. 1, is shown in Appendix B. If CRITTER is accidentally terminated (e.g., terminal interrupt) a scratch file will save the calculations completed up to the time of termination. It will have the same name as the output file but a file type SAV.

3. APPLICATIONS

With modification of the software, the digitizer method could be used effectively for a variety of applications. Additional amphipod parameters such as head length, number of antenna segments, body area, and gut volume could be easily computed and stored. The digitizer method is not limited to amphipods; photographic or traced images of anything from single-celled algae

to fish or macrophyte species could be placed on the digitizer tablet and any number of ecologically relevant lengths and areas could be measured and recorded. Although we used a high-resolution (1000 pints/inch) digitizer, we believe that increasing the magnification of the dissecting microscope (and thus the size of the image) would allow use of a smaller, and less expensive, lower resolution digitizer that might be readily interfaced with a personal computer.

4. REFERENCES

- California Computer Products, Inc., 1983. CalComp 9000 Series Digitizer Operator's Manual, Manual No. 50090-1, Revision B. Talos Systems, Inc, Scottsdale, 70 pp.
- Cooper, W.E., 1965. Dynamics and production of a natural population of a freshwater amphipod Hyaella azteca. Ecol. Monogr. 35:377-394.
- Quigley, M.A., 1986. Gut fullness of the deposit-feeding amphipod Pontoporeia hoyi in Southeastern Lake Michigan (unpublished manuscript). Great Lakes Environmental Research Laboratory, Ann Arbor, Michigan, 25 pp.
- Winberg, G.G., 1971. Methods for the Estimation of Production of Aquatic Animals. Academic Press, London, 175 pp.
- Winnell, M.H. and D.S. White, 1984. Ecology of shallow and deepwater populations of Pontoporeia hoyi (Smith) Amphipoda in Lake Michigan. Freshwater Invert. Biol. 3:118-138.

Appendix A: Initial CRITTER Prompts, Explanations,
and Example Responses

Prompt	Explanation	Example of response
Enter Output File Name (without file type)	Name of the output file to contain calculated amphipod lengths	EXAMPLE
Enter Station Name	Name of station where amphipods were collected	STA21
Enter Date	Date of collection	6/15/85
Enter Replicate Information	Information about replicates	1 of 5
Enter Number of First Animal to be Measured	Number associated with the first amphipod image to be measured during current session.	1
Enter Coordinates of Upper Right Corner (mm)	Coordinates (in amphipod length units) of upper right corner of tracing sheet. Assumes lower left coordinates = (0.,0.). Used for scaling purposes.	31.98,41.30

Appendix B: Example Output File Based on Information
from Appendix A and Fig. 1

EXAMPLE.OUT

Station: STA21
Date: 6/15/85
Replicate: 1 of 5

Animal Number	Total Body Length (mm)	n	Gut Fullness Length (mm)	Gut Type	n
1	0.6611D+01	48	0.5272D+01	C	33
2	0.3901D+01	24	0.2094D+01	C	15
3	0.6302D+01	44	0.4142D+01	S	37
4	0.5573D+01	39	0.0000D+00	E	0
5	0.6538D+01	52	0.4449D+01	S	41
6	0.4344D+01	27	0.0000D+00	E	0
7	0.5766D+01	35	0.4142D+01	C	22
8	0.6660D+01	4 1	0.4485D+01	S	35
9	0.5572D+01	44	0.0000D+00	E	0
10	0.5704D+01	38	0.4608D+01	C	22
11	0.4110D+01	40	0.0000D+00	E	0
12	0.7188D+01	49	0.5380D+01	S	39
13	0.5463D+01	42	0.0000D+00	E	0
14	0.5500D+01	32	0.0000D+00	E	0
15	0.7285D+01	53	0.5581D+01	C	32
16	0.5582D+01	46	0.3050D+01	C	24
17	0.5749D+01	34	0.0000D+00	E	0
18	0.5854D+01	36	0.4442D+01	S	35
19	0.5943D+01	39	0.0000D+00	E	0
20	0.7262D+01	45	0.5018D+01	C	33

Appendix C: Source Code Listing of Critter

```

PROGRAM CRITTER
c
c   Program to calculate amphipod total body length and gut fullness
c. using a CalComp 9480 digitizer. Amphipod images, projected onto a tracing
c   sheet by a camera lucida attached to a dissecting microscope, are traced
c   and placed on the tablet of the digitizer. The user executes CRITTER and
c   is prompted for an output file name, station name, date of sample,
c   replicate information, number of the first animal, and the coordinates of
c   the upper right corner of the tracing sheet (in amphipod length units).
c   The coordinates of the lower left corner are assumed to be (0,0).

c   The user then begins digitizing the amphipod tracings. The user must
c   measure each animal's total body length and gut fullness, in that order,
c   before moving on to another animal. Moving from body length measurement
c   to gut length measurement or from animal to animal is controlled by
c   entering appropriate menu items from the CRITTER menu.

c   From the set of digitized points, CRITTER calculates total body length
c   and gut fullness and records the results in an on-line computer file.
c   Total body length of the amphipod is defined as the scalar distance along
c   the amphipod's dorsal margin, beginning at the tip of the rostrum and
c   ending at the base of the telson. Gut fullness is defined as the length
c   of gut (along the central axis of the gut) that contains food matter. If
c   the gut content is segmented, possibly because of intermittent feeding,
c   then the total gut fullness is the sum of the individual segment lengths.

c   Input is from unit 5, the terminal or the digitizer. Output is to
c   units 1 and 2. On successful termination of CRITTER, the results are
c   written to a file (unit 1) named by the user at the start of execution
c   with a file type OUT. If CRITTER is accidentally terminated (terminal
c   interrupt) a scratch file (unit 2) will save the calculations completed
c   up to the time of termination. It will have the same name as the output
c   file but a file type of SAV.
c
c   implicit double precision (a-h,o-z)
c   real*8 llx, lly, urx, ury, llx1, lly1, lly2, lly3
c   real*8 length1(0:1000), length2(0:1000), lc
c   integer box, ans, pts, count, start_count, count_plus_1
c   integer no1(1000), no2(1000)
c   character L1(1000)*1, L2(1000)*1, dataf1*50
c   character station*50, date*20, repl*20
c   common /boxes/ llx, lly, urx, ury, llx1, lly1, lly2, lly3
c
c   data no1, no2/1000*0, 1000*0/
c   data L1, L2/1000*' ', 1000*' '/
c   data length1, length2/1001*0.d0, 1001*0.d0/
c   data llx, lly/0.d0, 0.d0/
c
c...Open output and scratch files
c   type *, ' '

```

```

104 type *, 'Enter Output File Name (without file type)'
    read(5,1,err=104) datafl
    open(1,file=datafl,defaultfile='.out',status='new')
    open(2,file=datafl,defaultfile='.sav',status='new')
c
c...Inquire station name, date, and replicate information
    type *, ''
105 type *, 'Enter Station Name'
    read(5,1,err=105) station
106 type *, 'Enter Date'
    read(5,1,err=106) date
107 type *, 'Enter Replicate Information'
    read(5,1,err=107) repl
c
c...Write station name, date, and replicate to first three lines of
c...scratch file
    write(2,4) station
    write(2,5) date
    write(2,6) repl
c
c...Inquire number of 1st measurement (for continuity)
    type *, ''
100 type *, 'Enter Number of First Animal to be Counted'
    read(5,*,err=100) count
    start count-count
    count-count-1
c
c...Inquire coordinates of upper right corner
103 type *, 'Enter Coordinates of Upper Right Corner (mm)'
    read(5,*,err=103) urx,ury
    llx1=0.815*urx
    lly1=ury-ury/4.
    lly2=ury-2.*ury/4.
    lly3=ury-3.*ury/4.
c
c...Calculate grace margin around page border (about 0.085")
    xrange=urx-llx
    yrange=ury-lly
    llx=llx-0.01*xrange
    urx=urx+0.01*xrange
    lly=lly-0.01*yrange
    ury=ury+0.01*yrange
c
c...Begin accepting data
    type *, ''
    type *, '***** BEGIN MEASUREMENTS *****'
    type *, ''
c
c
c...BODY LENGTH MEASUREMENT
10 ica-0
    count-plus-1=count+1
    write(6,9) count-plus-1

```



```

c...read digitized data point
  20 read(5,2,err=97) x2,y2
c...locate data point
  call find(x2,y2,box)
c...if this is the first data point then...
  if(ica.eq.0) then
    if(box.eq.1) then
      type *, ' '
      type *, char(7),char(7),char(7)
      type *, ' Bad Menu Box - End Body'
      type *, ' Reenter Desired Box'
      go to 20
    else if(box.eq.2) then
      type *, ' '
      type *, char(7),char(7),char(7)
      type *, ' Bad Menu Box - End Segment'
      type *, ' Reenter Desired Box'
      go to 20
    else if(box.eq.3) then
      type *, ' '
      type *, char(7),char(7),char(7)
      type *, ' Bad Menu Box - End Gut'
      type *, ' Reenter Desired Box'
      go to 20
    else if(box.eq.4) then
      type *, ' '
      type *, char(7),char(7),char(7)
      type *, ' Exit'
      go to 99
    endif
  endif
c
c...if this is not the first data point then...
  else if(ica.gt.0) then
    if(box.eq.1) then
      go to 30
    else if(box.eq.2) then
      type *, ' '
      type *, char(7),char(7),char(7)
      type *, ' Bad Menu Box - End Segment'
      type *, ' Reenter Desired Box'
      go to 20
    else if(box.eq.3) then
      type *, ' '
      type *, char(7),char(7),char(7)
      type *, ' Bad Menu Box - End Gut'
      type *, ' Reenter Desired Box'
      go to 20
    else if(box.eq.4) then
      count=count-1
      type *, ' '
      type *, char(7),char(7),char(7)
      type *, ' Exit'
      go to 99
    endif
  endif

```

```

        endif
    endif
c...number of body length data points
    ica=ica+1
c...number of animals
    if(ica.eq.1) count=count+1
c...calculate body length
    if(ica.gt.1) then
        length1(count)=length1(count)+dsqrt((x2-x1)
& *(x2-x1)+(y2-y1)*(y2-y1))
    endif
    x1=x2
    y1=y2
    go to 20
30 L1(count)='B'
    nol(count)=ica
c
c...write to terminal
    write(6,3) count,L1(count),length1(count),nol(count)
c
c...write to scratch file
    write(2,3) count,L1(count),length1(count),nol(count)
c
c
c
c...GUT LENGTH MEASUREMENT
39 icc=0
    icd=0
    lc=0
    iseg=0
    write(6,11) count-plus-1
c...read digitized data point
40 read(5,2,err=98) x2,y2
c...locate data point
    call find(x2,y2,box)
c...if this is the first data point then,..
    if(icc.eq.0) then
        if(box.eq.1) then
            type *, ' '
            type *, char(7),char(7),char(7)
            type *, ' Bad Menu Box - End Body'
            type *, ' Reenter Desired Box'
            go to 40
        else if(box.eq.2) then
            type *, ' '
            type *, char(7),char(7),char(7)
            type *, ' Bad Menu Box - End Segment'
            type *, ' Reenter Desired Box'
            go to 40
        else if(box.eq.3) then
            lc=0
            go to 50
        else if(box.eq.4) then

```

```

        count=count-1
        type *, ' '
        type *, char(7),char(7),char(7)
        type *, ' Exit'
        go to 99
    endif
c
c...if this is not the first data point then...
    else if(icc.gt.0) then
        if(box.eq.1) then
            type *, ' '
            type *, char(7),char(7),char(7)
            type *, ' Bad Menu Box - End Body'
            type *, ' Reenter Desired Box'
            go to 40
        else if(box.eq.2 .or. box.eq.3) then
            go to 50
        else if(box.eq.4) then
            count=count-1
            type *, ' '
            type *, char(7),char(7),char(7)
            type *, ' Exit'
            go to 99
        endif
    endif
c...number of data points in this segment
    icc=icc+1
c...number of data points in entire gut
    icd=icd+1
c...calculate segment length
    if(icc.gt.1) then
        lc=lc+dsqrt((x2-x1)*(x2-x1)+(y2-y1)*(y2-y1))
    endif
    x1=x2
    y1=y2
    go to 40
c...number of segments
    50 iseg=iseg+1
c...calculate entire gut length
    length2(count)=length2(count)+lc
    if(box.eq.2) then
        icc=0
        lc=0
        go to 40
    endif
    if(icd.gt.0 .and. iseg.eq.1) then
        L2(count)='C'
    else if(icd.eq.0 .and. iseg.eq.1) then
        L2(count)='E'
    else
        L2(count)='S'
    endif
    no2(count)=icd

```

```

c
c . ..write to terminal
      write(6,3) count,L2(count),length2(count),no2(count)
c . ..write to scratch file
      write(2,3) count,L2(count),length2(count),no2(count)
      go to 10

c
c...On error during body length read
  97 type *,''
      type *, char(7),char(7),char(7)
      type *, ' Bad Input Value - Possibly Line Noise - Reenter'
      type *,''
      write(6,7) count-plus-1
      length1(count_plus_1)=0.d0
      L1(count_plus_1)=''
      nol(count_plus_1)=0
      if(ica.gt.0) count-count-1
      go to 10

c
c...On error during gut length read
  98 type *,''
      type *, char(7),char(7),char(7)
      type *, ' Bad Input Value - Possibly Line Noise - Reenter'
      type *,''
      write(6,8) count-plus-1
      length2(count_plus_1)=0.d0
      L2(count_plus_1)=''
      no2(count_plus_1)=0
      go to 39

c
c . ..On exit
  99 type *,''
      type *, 'Enter 1 to Exit, 2 to Continue,'
      type *, '      3 to Quit (No Output Saved)'
      read(5,*,err=99) ans
c...Quit without saving any output
      if(ans.eq.3) then
        type *,''
        type *, ' No Output Saved'
        close(1,status='delete')
        close(2,status='delete')
        stop
c . ..Continue with same critter
      else if(ans.eq.2) then
        type *,''
        write(6,7) count_plus_1
        length1(count_plus_1)=0.d0
        length2(count_plus_1)=0.d0
        L1(count_plus_1)=''
        L2(count_plus_1)=''
        nol(count_plus_1)=0
        no2(count_plus_1)=0
        go to 10

```

```

        else if(ans.gt.4) then
            type *, char(7),char(7),char(7)
            type *, ' Bad Answer'
            go to 99
        endif
c...Exit cleanly
c...Write station name, date, and replicate to first three lines of
c...output file
        write(1,4) station
        write(1,5) date
        write(1,6) repl
c...Write count, animal type, and length to output file
        write(1,12)
        do j=start_count,count
            write(1,13) j,length1(j),no1(j),length2(j),L2(j),no2(j)
        end do
c...On clean exit from program - delete scratch file (unit 2)
        type *, ''
        type *, ' Output Saved in File: ',datafl
        close(1)
        close(2,status='delete')
c
        stop
1 format(a)
2 format(f7.3,1x,f7.3)
3 format(i4,2x,a1,2x,d15.8,3x,i4)
4 format(1x,'Station: ',a)
5 format(1x,'Date: ',a)
6 format(1x,'Replicate: ',a)
7 format('/' Redo Body - Animal #',i4)
8 format('/' Redo Gut - Animal #',i4)
9 format('/' BODY LENGTH - Animal #',i4)
11 format('/' GUT LENGTH - Animal #',i4)
12 format('/' Animal Total Body Gut',
        &' Fullness Gut'/' Number Length (mm) n',
        &' Length (mm) Type n'/)
13 format(i4,3x,d12.4,1x,i4,8x,d12.4,4x,a1,1x,i4)
        end
c
c
        subroutine find(x,y,box)
c...Subroutine to locate x-y pair. Box-1 is END BODY, box-2 is END SEGMENT,
c...box-3 is END GUT, box-4 is EXIT and all out-of-range pts, box-5 is all
c...other in-range pts.
        implicit double precision (a-h,o-z)
        real*8 llx,lly,urx,ury,llx1,lly1,lly2,lly3
        integer box
        common /boxes/ llx,lly,urx,ury,llx1,lly1,lly2,lly3
c
c...check to see if pair is out of range
        if(x.lt.llx .or. x.gt.urx .or. y.lt.lly .or. y.gt.ury) then
            box=4
            return

```

```

c
c...check to see if pair is in 1st box
    else if(x.gt.llx1 .and. y.gt.lly1) then
        box=1
        return
c
c...check to see if pair is in 2nd box
    else if(x.gt.llx1 .and. y.gt.lly2) then
        box=2
        return
c
c...check to see if pair is in 3rd box
    else if(x.gt.llx1 .and. y.gt.lly3) then
        box=3
        return
c
c...check to see if pair is in 4th box
    else if(x.ge.llx1 .and. y.le.lly1) then
        box=4
        return
c
c...If not out-of-range nor in any boxes then in-range
    else
        box=5
    endif
    return
end

```