Working Plan Implementation
(Working Paper)

IPTP/85/WP/13

INDO-PACIFIC TUNA DEVELOPMENT AND MANAGEMENT PROGRAMME

(INT/81/034)

This document forms part of publications of the Investigation on Indian Ocean and Western Pacific Small Tuna Resources Project (GCP/RAS/099/JPN) which is sponsored and funded by the Government of Japan

Colombo, Sri Lanka
February 1985

IPTP/85/WP/13 February 1985

MANUAL FOR STORING TUNA TAGGING DATA IN COMPUTER READABLE FORM

INDONESIA

M. Honma and T. Yonemori
GCP/RAS/099/JPN

INDO-PACIFIC TUNA DEVELOPMENT AND MANAGEMENT PROGRAMME
Colombo, Sri Lanka

PREFACE

The Research Institute for Marine Fisheries, Jakarta Indonesia (BPPL), has carried out tuna tagging in the Eastern Indonesian waters from December 1983 through May 1984, in cooperation with the project of the Investigation on Indian Ocean and Western Pacific Small Tuna Resources (GCP/RAS/099/JPN) funded by the Japanese Government. The tagging procedure was reported in detail in an IPTP working paper (IPTP/85/WP/12) - "TUNA TAGGING IN EASTERN INDONESIAN WATERS".

This manual was prepared originally by Mr. M. Honma, Computer Consultant, for local scientists of the tuna research group in the Institute to process tagging data using a computer. The scientist of tuna research group were made fully conversant with the procedures described in this manual.

Release and recovery data are separately stored in flexible discs in order to economize the recording space. Both data files are randomly accessible. The configuration of the computer system used was as follows.

1. HEWLETT PACKARD 87 XM (124 KB)
2. HEWLETT PACKARD 7470 A PLOTTER
3. HEWLETT PACKARD 82901 M FLEXIBLE DISC DRIVE
4. HEWLETT PACKARD 82905 B PRINTER

TABLE OF CONTENTS

3. Storing of data associates with Tagged Fish

3.1. Input data items

No.	Input item	Columns	Bytes	Remarks
1.	Area	6	3 + 6 = 9	
2.	Name of ship	2	3 + 2 = 5	
3.	Fishing method	1	8	1: pole and line, 2: troll 3: purse seine, 4: longline
4.	Operation No	3	8	
5.	Date of release	6	8 x 3 = 24	day - month - year
6.	Time of release	4	8 x 2 = 16	O'clock, minute
7.	Location of release	11	(3+1+8+8)x2 =40	Lat: N or S, degree, minute Long: E, degree, minute
8.	Weather	1	8	1: clear, 2: cloudly 3: rain 0: unknown
9.	Air temperature	4	8	Until (^{o}C): XX.X Unknown: 00.0
10.	Water temperature	4	8	Unit (^{o}C): XX.X Unknown: 00.0
11.	Type of school	1	8	1: Jumping, 2: free swiming 3: Drift log, 4: whale 5: Dolpin, 6: Payao, 0: Unknown
12.	Type of taggine	1	8	1: Single, 2: Double
13.	Tag No.	5	3+1+8 = 12	
14.	Species	1	3 + 1 = 4	S: skipjack, Y: yellowfin
15.	Fork length	3	8	Unit: cm
16.	Condition of fish	1	8	1: food, 2: bad, 0: unknown
17.	Catch	5	8	Unit: number, unknown: 000.0
18.	Paired No. when double tagging	5	3+1+8 = 12	

3.2 Explanation of input item codes

No. 1. Area -- 6 column (1-6)
Tagging area or port name: Less than 6 characters or digits should be used. Input "XXXXXX".

No. 2. Code for name of ship -- 2 column (7 - 8). Input "XX"

No. 3. Fishing method -- 1 column (9)
1: Pole and line 2: Trolling
3: Purse seine 4: Longline
5: Handline

No. 4. Operation No. -- 3 columns (10 - 12)

No. 5. Date of release -- 6 columns (13 - 18)
Day (2 col.), Month (2 col.), Year (2 col.)

No. 6. Time of release -- 4 columns (19-22). Hour (2 col.), Time (2 col.)

No. 7. Location of release -- 11 columns (23 - 33)
Latitude N or S (1 col.), Degree (2 col.), Minute (2 col.)
Longitude E (1 col.), Degree (2 col.), Minute (2 col.)
Example S 01°26'; E 131°06' -- "S", 01,26, "E", 131, 06

No. 8. Weather -- 1 column (34).
1: Clear, 2: Cloudly, 3: Rainy, 0:unknown

No. 9. Air temperature (°C) -- 4 columns (35-38). Input "XX.X".
Unknown: 00.0

No. 10. Water temperature (°C) -- 4 columns (39-42). Input "XX.X"
Unknown: 00.0

No. 11. Type of school -- 1 col (43).
1: Jumping with birds, 2: Free swimming
3: associated with drifting object, 4: associated with whale
5: associated with dolphin, 6: Payao, 0: unknown

No. 12. Type of tagging -- 1 column (44). Single : 1, Double tag : 2

No. 13. Tag No. -- 5 columns (45 - 47). 1 Character + 4 digits

No. 14. Spacies -- 1 column (50)
S: Skipjack, Y: Yellowfin, B: Bigeye

No. 15. Fork Length -- 3 column (51 - 53)
In nearest upper cm (For example, 54.1 - 55.0 → 55)

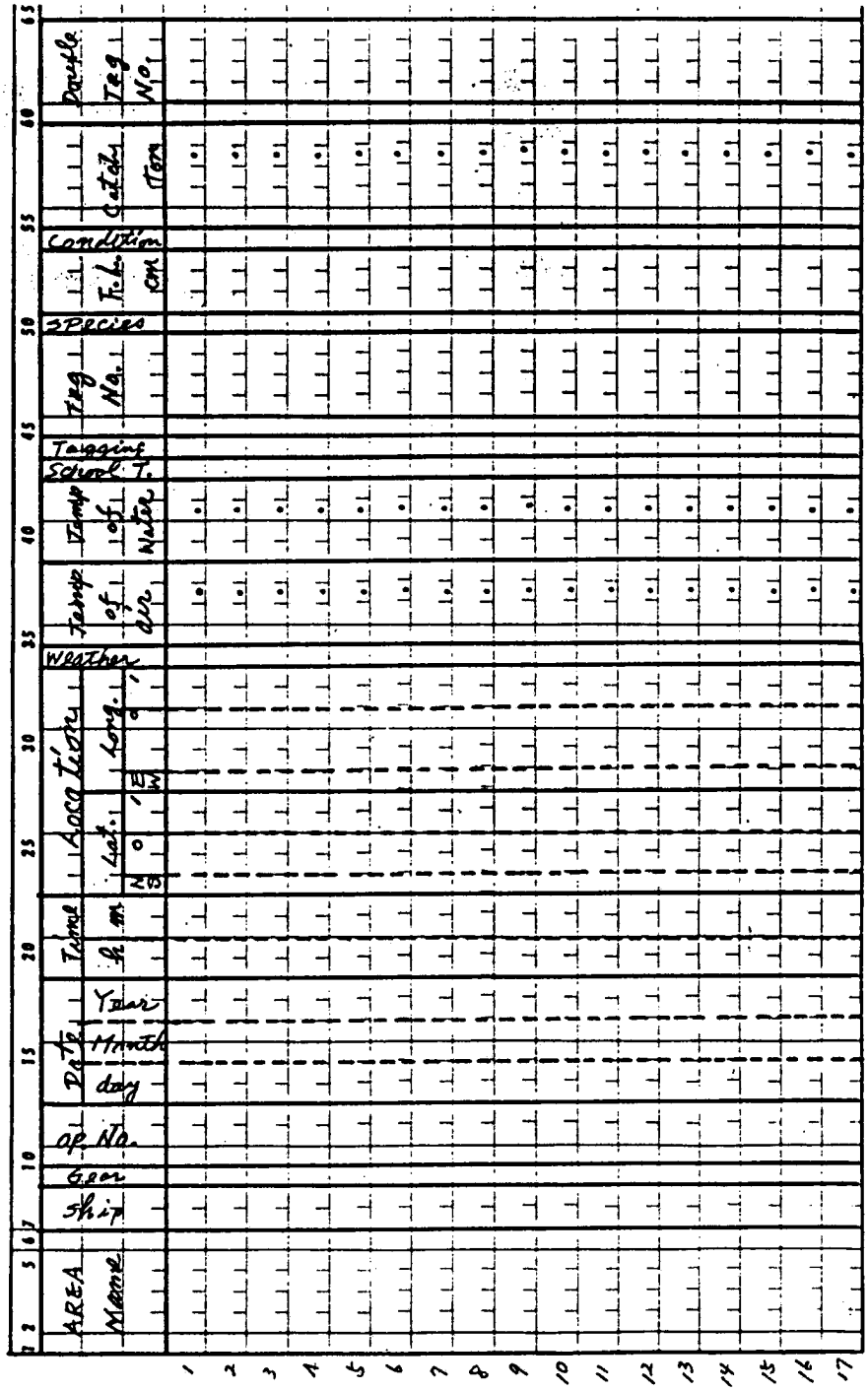
No. 16. Physical condition of tagged fish -- 1 column (54)
1: Good, 2: bad 0:unknown

No 17. Catch by species -- 5 columns (55 - 59)
Input in number of fish caught. Unknown 000.0

No. 18. Paired tag number when double tagging -- 5 columns (60-64).
1 character + 4 digits

3.3 Data Input Format

III INPUT DATA FOR RELEASE DATA



4. Storing of data associated with recovered fish

4.1 Input data items

No.	Input item	Columns	Bytes	Remarks
1.	Organization of tag-sender	4	8 x 2 = 16	Organization (2 col), Country name (2 col)
2.	Tag No.	5	3+1+8 = 12	1 Character 4 digits
3.	Gear	1	8	1:P&L, 2:Troll, 3:Purse seine, 4:Longline, 5: Handline
4.	Species	1	3 + 1 = 4	S: Skipjack, Y: Yellowfin
5.	Date of recovery	6	8 x 3 = 24	Day, Month, Year
6.	Location of recovery	11	(3+1+8+8)x2 = 40	Lat: N or S, Degree, Minute Long: E, Degree, Minute
7.	Water temperature	4	8	Unit : ^{o}C. Input XX.X Unknown: 00.0
8.	Fork length	3	8	Unit: cm Input XX
9.	Body weight	4	8	Unit: Kg. Input XX.X
10.	Finding place	1	8	
11.	Single tag or double	1	8	
12.	Paired tag No.	5	3+1+8 = 12	1 character 4 digits
13.	Days at liberty	4	8	Input value calculated by computer
14.	Distance	5	8	Input value calculated by computer
15.	Series No.	4	8	

4.2 Explanation on input item codes

No. 1. Organization of tag sender -- 4 columns (1 - 4)

Organization
01: Commercial boat, 02: Research boat
03: Fishing body (Company, Corperation, Fish market etc)
04: Processor (cannery etc)

Country
01: Home country, 02: Foreign country, 03: Others

No. 2. Tag No. -- 5 columns (5 - 9)
1 Character 4 digits

No. 3. Gear -- 1 column (10)
1: Pole and line, 2: Trolling, 3: Purse seine,
4: Longline, 5: Handline, 9: Unknown

No. 4. Species -- 1 column (11)
S: Skipjack, Y: Yellowfin

No. 5. Date of recovery -- 6 columns (12 - 17)
Day (2 col), Month (2 col), Year (2 col)

No 6. Location of recovery -- 11 columns (18 - 28)
Latitute -- N or S (1 col), Degree (2 col), Minute (2 col)
UNknown -- 00000000000
Longtitude -- E (1 col), Degree (2 col), Minute (2 col)

No. 7. Water temperature -- 4 columns (29 - 32)
Unit: ^{o}c. Input XX.X Unknown: 00.0

No. 8. Fork length -- 3 columns (33 - 35)
Unit: cm (nearest upper cm)
Example 51.1 - 52.0 --→ 052

No. 9. Weight -- 4 columns (36 -39)
Unit: Kg. Input XX.X
Unknown: 00.0

No. 10. Location of finding -- 1 column (40)
1: Aboard, 2: Landing Site, 3: Cannery
4: Others, 9: Unknown

No. 11. Number of tag retrieved -- 1 columns (42 - 46)

No. 12. Paired tag number when double tagging -- 5 columns (42 - 46)

No 13. Number of days at liberty -- 4 columns (47 - 50)
Input a value calculated by computer

No 14. Distance of Movement -- 5 columns (51 - 55)
Input a value calculated by computer

No. 15. Series recovery No. -- 5 columns (56 - 59)

4.3 Data input format

IV INPUT FORMAT FOR RECOVERY DATA

Recovery

Season S. No.	Tag No.	Species Code	Date Day	Month	Year	Location Lat.	Long.	Temp of water	F.L. cm	B.W. kg	Female Tag No.	Double Tag No.	No. of days	Distance	Consecutive Numbers
						MS ° '	EW ° '								

Release

AREA NAME	Ship No.	Species Code	Age No.	Date Day	Month	Year	Time h	m	Location Lat.	Long.	Weather	Temp of air	Temp of water	Tissue Sample T.	Tag No.	Species	F.L. cm	Condition	Catch Number	Double Tag No.

Alongwith recovery data the corresponding release data should also be entered

5. Variables in data files

HP - 87 Micro-computer has limited capacity in statement access and reading, the names of variables should be as compact as possible. The following common variable names are adopted in this manual. When a long statement is required, the above is converted into a dimensioned variable name.

Names of Variables

Release

AREA NAME	Ship No.	Species Code	Age No.	Date Day	Month	Year	Time h	m	Location Lat.	Long.	Weather	Temp of air	Temp of water	Tissue Sample T.	Tag No.	Species	F.L. cm	Condition	Catch Ton	Double Tag No.
A$	S$		D$	D	M	Y	H	Mi	A1	A2	A3	A4	A.T	WT		NO		FL		CA	NO2

Q$ NS$ EW$ W S N MA$ F$ C NA2$

Recovery

Species System S.	Tag No.	Species Code	Sample Code	Date Day	Month	Year	Location Lat.	Long.	Temp of water	F.L. cm	B.W. kg	Female Tag No.	Double Tag No.	No. of days	Distance	Consecutive Numbers
AA	BB	RNO		DB	MM	YY	L5	L6	WZ	FL2	BW		RNO2	DA	DI	NNN

RNA$ Q2 RF$ N2$ E2$ P FF RNA2$

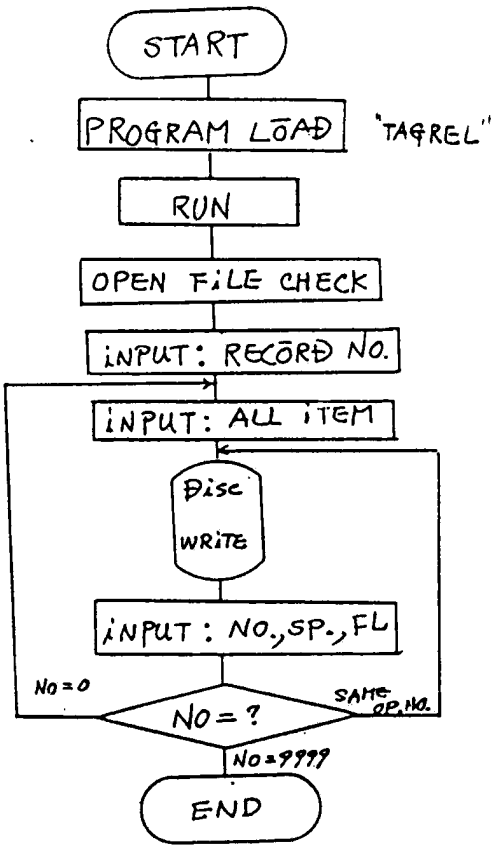
6. Programs for storing and listing of data

6.1. Storing of data of released fish, "TAGREL"

6.1.1 Procedure

(1). Load "TAGREL"
(2). Press RUN key
(3). Put in FILE NAME to be assigned
(4). Put in RECORD NUMBER
(5). Put in all the data items
(6). In the same OPERATION No., Just put in TAG No. SPECIES and LENGTH
(7). When OPERATION No. changes, put in 0,0,0 and repeat (5) and (6)
(8). When data in-put is stopped or finished, put in 9999,0,0

6.1.2 Flow chart



6.1.3 Program listing

```
10 REM ********** PROGRAM NAME "TAGREL" **********
20 REM ********** RELEASE INPUT PROGRAM **********
30 OPTION BASE 1
40 !
50 DISP "PLEASE FILE SELECT AND INPUT"
60 DISP "     SORONG.TG8401"
70 DISP "     SORONG.TG8402"
80 DISP "     SORONG.TG8403"
90 DISP "     AMBON.TG8411"
100 DISP "     AMBON.TG8412"
110 DISP "     BITUNG.TG8421"
120 DISP "     BITUNG.TG8422"
130 INPUT FINAME$
140 ASSIGN# 1 TO  FINAME$
150 DIM A$(1)[6],S$(1)[2],NS$(1)[1],EW$(1)[1],NA$(1)[1],F$(1)[1],NA2$(1)[1]
160 DIM YYY(7),ZZZ(7),NAMA$(3)[1]
170 NAMA$(1)="Y" @ NAMA$(2)="S" @ NAMA$(3)="A"
180 DISP "HOW MUCH RECORD NUMBER ";
190 INPUT NUMB
200 NUMB=NUMB-1
210 NEWENTRY:
220 DISP "INPUT----> AREA NAME, SHIP, .............,DOUBLE TAG NO."
230 INPUT A$,S$,G,OP,D,M,Y,H,MI,NS$,L1,L2,EW$,L3,L4,W,AT,WT,S,N,NA$,NO,FFFF,FL,C
,CA,NA2$,NO2
240 YYY(1)=G @ YYY(2)=OP @ YYY(3)=D @ YYY(4)=M @ YYY(5)=Y @ YYY(6)=H @ YYY(7)=MI

250 ZZZ(1)=L3 @ ZZZ(2)=L4 @ ZZZ(3)=W @ ZZZ(4)=AT @ ZZZ(5)=WT @ ZZZ(6)=S @ ZZZ(7)
=N
260 GOTO 320
270 CONTINUE:
280 DISP "INPUT----> NO...........F.L."
290 INPUT NO,FFFF,FL
300 IF NO=9999 THEN GOTO 360
310 IF NO=0 THEN GOTO 210
320 F$=NAMA$(FFFF)
330 NUMB=NUMB+1
340 PRINT# 1,NUMB ; A$,S$,YYY(),NS$,L1,L2,EW$,ZZZ(),NA$,NO,F$,FL,C,CA,NA2$,NO2
350 GOTO 270
360 ASSIGN# 1 TO  *
370 END
```

6.2. Data listing program for released fish, "TAGLST"

6.2.1 Procedure

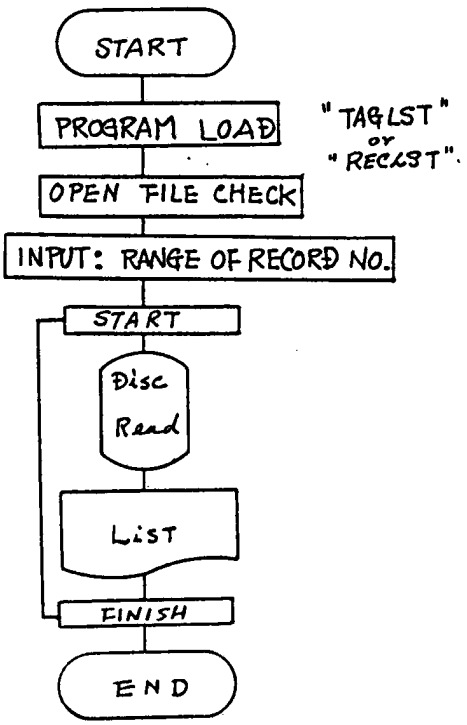
(1). Load "TAGLST"
(2). Press RUN key
(3). Put in FILE NAME to be assigned
(4). Put in the range of records to be listed up
(5). Out-put

Sample of out-put

***** TAGGING DATA LIST-UP *****

NO.	AREA	SH	G	OP	DATE	TIME	LAT.	LONG.	W	AT	WT	S	T	TAGNO	SP	FL	C
1	SORONG	01	1	1	18 12 83	13 58	S0026	E12946	1	0.0	29.0	1	1	A0001	Y	55	1
2	SORONG	01	1	1	18 12 83	13 58	S0026	E12946	1	0.0	29.0	1	1	A0002	Y	31	1
3	SORONG	01	1	1	18 12 83	13 58	S0026	E12946	1	0.0	29.0	1	1	A0003	Y	51	1
4	SORONG	01	1	1	18 12 83	13 58	S0026	E12946	1	0.0	29.0	1	1	A0004	Y	53	1
5	SORONG	01	1	1	18 12 83	13 58	S0026	E12946	1	0.0	29.0	1	1	A0005	Y	46	1
6	SORONG	01	1	1	18 12 83	13 58	S0026	E12946	1	0.0	29.0	1	1	A0006	S	42	1
7	SORONG	01	1	1	18 12 83	13 58	S0026	E12946	1	0.0	29.0	1	1	A0007	Y	49	1
8	SORONG	01	1	1	18 12 83	13 58	S0026	E12946	1	0.0	29.0	1	1	A0008	Y	51	1
9	SORONG	01	1	1	18 12 83	13 58	S0026	E12946	1	0.0	29.0	1	1	A0009	Y	58	1
10	SORONG	01	1	1	18 12 83	13 58	S0026	E12946	1	0.0	29.0	1	1	A0010	Y	48	1

6.2.2 Flow chart



6.2.3 Program listing

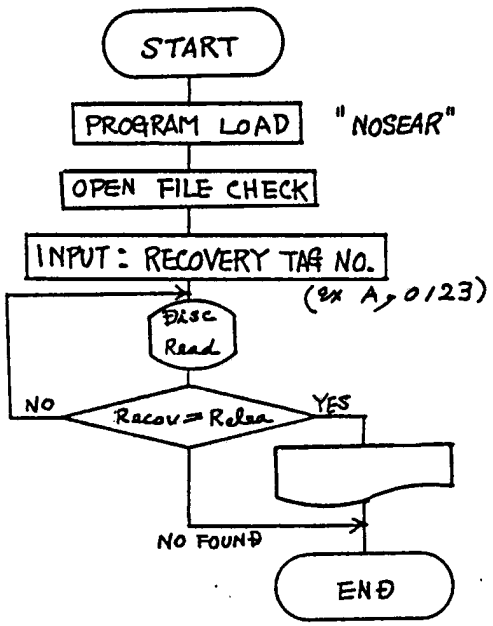
```
10 REM ********** PROGRAM NAME "TAGLST" **************************
20 REM ********** TAGGING RELEASE DATA LIST-UP PROGRAM **********
30 OPTION BASE 1
40 DISP "PLEASE ASSIGN FILE SELECT AND INPUT"
50 DISP "      SORONG.TG8401"
60 DISP "      SORONG.TG8402"
70 DISP "      SORONG.TG8403"
80 DISP "      AMBON.TG8411"
90 DISP "      AMBON.TG8412"
100 DISP "      BITUNG.TG8421"
110 DISP "      BITUNG.TG8422"
120 INPUT FINAME$
130 ASSIGN# 1 TO  FINAME$
140 PRINTER IS 701
150 DIM A$(1)[6],S$(1)[2],NS$(1)[1],EW$(1)[1],NA$(1)[1],F$(1)[1],NA2$(1)[1]
160 DIM YYY(7),ZZZ(7)
170 DISP "INPUT-----> RECORD NUMBER OF LIST-UP RANGE"
180 INPUT START,FINISH
190 NUMBER=0
200 PRINT "     ***** TAGGING DATA LIST-UP *****"
210 PRINT
220 PRINT " NO.  AREA  SH G  OP   DATE   TIME   LAT.  LONG. W  AT   WT  S T TAGN
O SP FL C"
230 PRINT
240 FOR I=START TO FINISH
250 READ# 1,I ; A$,S$,YYY(),NS$,L1,L2,EW$,ZZZ(),NA$,NO,F$,FL,C,CA,NA2$,NO2
260 G=YYY(1) @ OP=YYY(2) @ D=YYY(3) @ M=YYY(4) @ Y=YYY(5) @ H=YYY(6) @ MI=YYY(7)

270 L3=ZZZ(1) @ L4=ZZZ(2) @ W=ZZZ(3) @ AT=ZZZ(4) @ WT=ZZZ(5) @ S=ZZZ(6) @ N=ZZZ(
7)
280 PRINT USING 290 ; I,A$,S$,G,OP,D,M,Y,H,MI,NS$,L1,L2,EW$,L3,L4,W,AT,WT,S,N,NA
$,NO,F$,FL,C
290 IMAGE 4D,1X,6A,1X,2A,DD,1X,3D,5(1X,2Z),1X,1A,2(2Z),1X,1A,3Z,2Z,2D,2(3D.D),2D
,2D,1X,1A,4Z,1X,1A,4D,2D
300 NEXT I
310 ASSIGN# 1 TO  *
320 END
```

6.3. Pick-up program of recoverd fish from release list, "NOSEAR".

6.3.1 Procedure

(1). Load "NOSEAR"
(2). Press RUN key
(3). Put in FILE NAME to be assigned
(4). Put in TAG. No. to be picked up (Example: A,0121).
(5). Out-put (release record).

6.3.2 Flow chart




```
RECOVERY TAG NO.----- A0121

--------RELEASE DATA --------

 NO.  AREA  SH G  OP   DATE   TIME   LAT.  LONG. W  AT   WT  S T TAGNO SP FL C
 121 SORONG 01 1   3 19 12 83 09 20 S0014 E12955 1  0.0 29.0 1 1 A0121 Y  45 1
       0.0 00000
     CATCH TAGNO
```

6.3.3 Program listing

```
10 REM ********** PROGRAM NAME "NOSEAR" **********
20 REM ********** TAG NO. SEARCH PROGRAM *********
30 OPTION BASE 1
40 PRINTER IS 701
50 DISP "PLEASE ASSIGN FILE SELECT AND INPUT"
60 DISP "       SORONG.TG8401"
70 DISP "       SORONG.TG8402"
80 DISP "       SORONG.TG8403"
90 DISP "       AMBON.TG8411"
100 DISP "       AMBON.TG8412"
110 DISP "       BITUNG.TG8421"
120 DISP "       BITUNG.TG8422"
130 INPUT FINAME$
140 ASSIGN# 1 TO  FINAME$
150 DIM A$(1)[6],S$(1)[2],NS$(1)[1],EW$(1)[1],NA$(1)[1],F$(1)[1],NA2$(1)[1]
160 DIM RNA$(1)[1],RNA2$(1)[1],YYY(7),ZZZ(7)
170 DISP "INPUT-----> RECOVERY TAG NO.(CORD AND NUMBER)"
180 INPUT RNA$,NUMBER
190 FOR I=1 TO 1310
200 READ# 1,I ; A$,S$,YYY(),NS$,L1,L2,EW$,ZZZ(),NA$,NO,F$,FL,C,CA,NA2$,NO2
210 IF NA$=RNA$ THEN GOTO 220 ELSE GOTO 230
220 IF NO=NUMBER THEN GOTO OUTPUT
230 IF NA2$=RNA$ THEN GOTO 240 ELSE GOTO AGAIN
240 IF NO2=NUMBER THEN GOTO OUTPUT
250 AGAIN:
260 NEXT I
270 GOTO FINISH
280 OUTPUT:
290 G=YYY(1) @ OP=YYY(2) @ D=YYY(3) @ M=YYY(4) @ Y=YYY(5) @ H=YYY(6) @ MI=YYY(7)

300 L3=ZZZ(1) @ L4=ZZZ(2) @ W=ZZZ(3) @ AT=ZZZ(4) @ WT=ZZZ(5) @ S=ZZZ(6) @ N=ZZZ(
7)
310 PRINT
320 PRINT USING 330 ; RNA$,NUMBER
330 IMAGE "RECOVERY TAG NO.---- ",1A,4Z
340 PRINT
350 PRINT "-------RELEASE DATA ------- "
360 PRINT
370 PRINT " NO.  AREA  SH G  OP   DATE   TIME   LAT.  LONG. W  AT   WT  S T TAGN
O SP FL C"
380 PRINT USING 390 ; I,A$,S$,G,OP,D,M,Y,H,MI,NS$,L1,L2,EW$,L3,L4,W,AT,WT,S,N,NA
$,NO,F$,FL,C,CA,NA2$,NO2
390 IMAGE 4D,1X,6A,1X,2A,DD,1X,3D,5(1X,2Z),1X,1A,2(2Z),1X,1A,3Z,2Z,2D,2(3D.D),2D
,2D,1X,1A,4Z,1X,1A,4D,2D,7X,3D.D,1X,1A,4Z
400 PRINT "      CATCH TAGNO"
410 FINISH:
420 ASSIGN# 1 TO  *
430 END
```

6.4 Program for storing of data of recovered fish, "RECOVE"

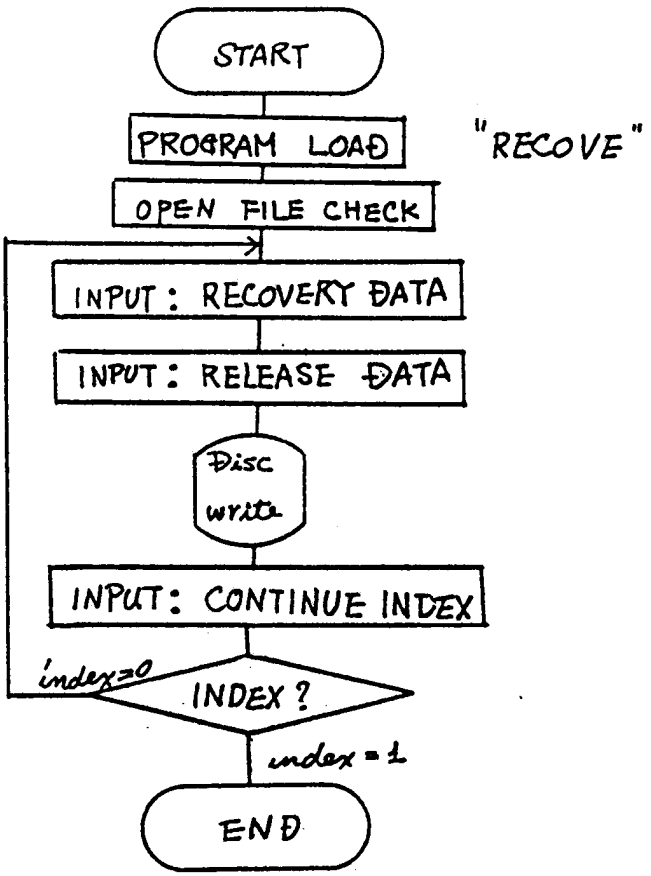
6.4.1 Procedure

(1). Load "RECOVE"
(2). Press RUN key
(3). FILE NAME to be assigned is pre-collaborated in the programme
(4). Put in recovery data
(5). Put in release data
(6). When succesively put in data, put in 0, and then repeat (4) and (5).
When stopping or finishing, put in 1.

* Recovery and release data are recorded separately in the disc (202 bytes, each). Random access can be done by the consecutive No. of recovery

** Days at liberty and distance of movement should be calculated using Programme in 7 and 8

6.4.2 Flow chart



6.4.3 Program listing

```
10 REM ********** PROGRAM NAME "RECOVE" **********
20 REM ********** RECOVERY INPUT PROGRAM **********
30 !
40 OPTION BASE 1
50 PRINTER IS 701
60 ASSIGN# 1 TO  "RECOVE.RC8485" !     THIS IS USE FOR FILE 1984-1985
70 DIM A$(1)[6],S$(1)[2],NS$(1)[1],EW$(1)[1],NA$(1)[1],F$(1)[1],NA2$(1)[1],YYY(7
),ZZZ(7)
80 DIM RNA$(1)[1],RF$(1)[1],N2$(1)[1],E2$(1)[1],RNA2$(1)[1],XXX(7)
90 CCC:
100 DISP "INPUT-----> RECOVERY DATA"
110 INPUT AA,BB,RNA$,RNO,G2,RF$,DD,MM,YY,N2$,L5,L6,E2$,L7,L8,W2,FL2,BW,P,FF,RNA2
$,RNO2,DA,DI,NNN
120 DISP "INPUT-----> RELEASE DATA"
130 INPUT A$,S$,G,OP,D,M,Y,H,MI,NS$,L1,L2,EW$,L3,L4,W,AT,WT,S,N,NA$,NO,F$,FL,C,C
A,NA2$,NO2
140 XXX(1)=L7 @ XXX(2)=L8 @ XXX(3)=W2 @ XXX(4)=FL2 @ XXX(5)=BW @ XXX(6)=P @ XXX(
7)=FF
150 YYY(1)=G @ YYY(2)=OP @ YYY(3)=D @ YYY(4)=M @ YYY(5)=Y @ YYY(6)=H @ YYY(7)=MI

160 ZZZ(1)=L3 @ ZZZ(2)=L4 @ ZZZ(3)=W @ ZZZ(4)=AT @ ZZZ(5)=WT @ ZZZ(6)=S @ ZZZ(7)
=N
170 RECN2=NNN*2
180 RECN1=RECN2-1
190 PRINT# 1,RECN1 ; AA,BB,RNA$,RNO,G2,RF$,DD,MM,YY,N2$,L5,L6,E2$,XXX(),RNA2$,RN
O2,DA,DI,NNN
200 PRINT# 1,RECN2 ; A$,S$,YYY(),NS$,L1,L2,EW$,ZZZ(),NA$,NO,F$,FL,C,CA,NA2$,NO2
210 DISP "INPUT CONTINUE ?      YES-----> 0      NO-----> 1"
220 INPUT CONTIN
230 IF CONTIN=0 THEN GOTO CCC
240 ASSIGN# 1 TO  *
250 END
```

6.5. Data listing for recovered fish, "RECLST", "RECLST2"

6.5.1 Procedure

(1). Load "RECLST"
(2). Press RUN key
(3). FILE NAME to be read is pre-collaborated in the programme
(4). Put in the range of records to be listed up
(5). Out-put

6.5.2 Flow chart

Same as 6. 2

6.5.3 Program listing - 1

"RECLST"

```
10 REM ********** PROGRAM NAME "RECLST" **********************
20 REM ********** TAG RECOVERY DATA LIST-UP PROGRAM **********
30 !
40 OPTION BASE 1
50 PRINTER IS 701
60 ASSIGN# 1 TO  "RECOVE.RC8485" !     THIS IS USE FOR FILE 1984-1985
70 DIM A$(1)[6],S$(1)[2],NS$(1)[1],EW$(1)[1],NA$(1)[1],F$(1)[1],NA2$(1)[1],YYY(7
),ZZZ(7)
80 DIM RNA$(1)[1],RF$(1)[1],N2$(1)[1],E2$(1)[1],RNA2$(1)[1],XXX(7)
90 DISP "INPUT----> RECOVERY NUMBER OF LIST-UP RANGE";
100 INPUT START,FINISH
110 START=START*2-1
120 FINISH=FINISH*2
130 FOR I=START TO FINISH STEP 2
140 READ# 1,I ; AA,BB,RNA$,RNO,G2,RF$,DD,MM,YY,N2$,L5,L6,E2$,XXX(),RNA2$,RNO2,DA
,DI,NNN
150 READ# 1,I+1 ; A$,S$,YYY(),NS$,L1,L2,EW$,ZZZ(),NA$,NO,F$,FL,C,CA,NA2$,NO2
160 L7=XXX(1) @ L8=XXX(2) @ W2=XXX(3) @ FL2=XXX(4) @ BW=XXX(5) @ P=XXX(6) @ FF=X
XX(7)
170 G=YYY(1) @ OP=YYY(2) @ D=YYY(3) @ M=YYY(4) @ Y=YYY(5) @ H=YYY(6) @ MI=YYY(7)
180 L3=ZZZ(1) @ L4=ZZZ(2) @ W=ZZZ(3) @ AT=ZZZ(4) @ WT=ZZZ(5) @ S=ZZZ(6) @ N=ZZZ(
7)
190 II=(I+1)/2
200 PRINT
210 PRINT USING 220 ; "***** RECOVERY RECORD *****    NO...",II
220 IMAGE 36A,5D
230 PRINT
240 PRINT USING 670 ; "SENDER SYSTEM.....",AA,BB
250 PRINT USING 680 ; "TAG NO.....",RNA$,RNO
260 PRINT USING 630 ; "GEAR.....",G2
270 PRINT USING 690 ; "SPECIES.....",RF$
280 PRINT USING 670 ; "DATE.....",DD,MM,YY
290 PRINT USING 700 ; "LATITUDE.....",N2$,L5,L6
300 PRINT USING 710 ; "LONGITUDE.....",E2$,L7,L8
310 PRINT USING 640 ; "WATER TEMP.....",W2
```

(continued)

```
320 PRINT USING 630 ; "FORK LENGTH(cm)...",FL2
330 PRINT USING 640 ; "BODY WEIGHT(kg)...",BW
340 PRINT USING 630 ; "FOUND PLACE.......",P
350 PRINT USING 630 ; "FOUND TAG NO......",FF
360 PRINT USING 680 ; "DOUBLE TAG NO.....",RNA2$,RNO2
370 PRINT USING 630 ; "NO. OF DAYS.......",DA
380 PRINT USING 630 ; "DISTANCE..........",DI
390 PRINT USING 630 ; "CONSECUTIVE NO....",NNN
400 PRINT
410 PRINT "***** RELEASE RECORD *****"
420 PRINT
430 PRINT USING 650 ; "AREA NAME........",A$
440 PRINT USING 660 ; "SHIP NAME.........",S$
450 PRINT USING 630 ; "GEAR..............",G
460 PRINT USING 630 ; "OPERATION NO......",OP
470 PRINT USING 670 ; "DATE..............",D,M,Y
480 PRINT USING 670 ; "TIME..............",H,MI
490 PRINT USING 700 ; "LATITUDE..........",NS$,L1,L2
500 PRINT USING 710 ; "LONGITUDE.........",EW$,L3,L4
510 PRINT USING 630 ; "WEATHER...........",W
520 PRINT USING 640 ; "AIR TEMP..........",AT
530 PRINT USING 640 ; "WATER TEMP........",WT
540 PRINT USING 630 ; "SCHOOL TYPE.......",S
550 PRINT USING 630 ; "NUMBER OF TAGS....",N
560 PRINT USING 680 ; "TAG NO............",NA$,NO
570 PRINT USING 690 ; "SPECIES...........",F$
580 PRINT USING 630 ; "FORK LENGTH(cm)...",FL
590 PRINT USING 630 ; "RELEASE CONDITION.",C
600 PRINT USING 640 ; "OP. CATCH.........",CA
610 PRINT USING 680 ; "DOUBLE TAG NO.....",NA2$,NO2
620 PRINT
630 IMAGE 18A,5D
640 IMAGE 18A,5D.D
650 IMAGE 17A,6A
660 IMAGE 18A,3X,2A
670 IMAGE 18A,3(3X,2Z)
680 IMAGE 18A,1A,4Z
690 IMAGE 18A,4X,1A,4Z
700 IMAGE 18A,4X,1A,2(3X,2Z)
710 IMAGE 18A,4X,1A,2X,3Z,3X,2Z
720 PRINT
730 NEXT I
740 ASSIGN# 1 TO  *
750 END
```

```
***** RECOVERY RECORD *****    NO...    1

SENDER SYSTEM.....   03   01
TAG NO............A0121
GEAR..............    1
SPECIES...........    Y
DATE..............   28   02   84
LATITUDE..........    S   00   31
LONGITUDE.........    E  130   36
WATER TEMP........    0.0
FORK LENGTH(cm)...   46
BODY WEIGHT(kg)...    2.5
FOUND PLACE.......    1
FOUND TAG NO......    1
DOUBLE TAG NO.....00000
NO. OF DAYS.......   71
DISTANCE..........   44
CONSECUTIVE NO....    1

***** RELEASE RECORD *****

AREA NAME........SORONG
SHIP NAME.........   01
GEAR..............    1
OPERATION NO......    3
DATE..............   19   12   83
TIME..............   09   20
LATITUDE..........    S   00   14
LONGITUDE.........    E  129   55
WEATHER...........    1
AIR TEMP..........    0.0
WATER TEMP........   29.0
SCHOOL TYPE.......    1
NUMBER OF TAGS....    1
TAG NO............A0121
SPECIES...........    Y
FORK LENGTH(cm)...   45
RELEASE CONDITION.    0
OP. CATCH.........  315.0
DOUBLE TAG NO.....00000
```

Program listing

"RECLST2"

```
10 REM ********** PROGRAM NAME "RECLST2" **********************
20 REM ********** TAG RELEASE AND RECOVERY LIST-UP ************
30 !
40 OPTION BASE 1
50 PRINTER IS 701
60 ASSIGN# 1 TO  "RECOVE.RC8485" !     THIS IS USE FOR FILE 1984-1985
70 DIM A$(1)[6],S$(1)[2],NS$(1)[1],EW$(1)[1],NA$(1)[1],F$(1)[1],NA2$(1)[1],YYY(7
),ZZZ(7)
80 DIM RNA$(1)[1],RF$(1)[1],N2$(1)[1],E2$(1)[1],RNA2$(1)[1],XXX(7)
90 DISP "INPUT-----> RECOVERY NUMBER OF LIST-UP RANGE";
100 INPUT START,FINISH
110 START=START*2-1
120 FINISH=FINISH*2
130 PRINT
140 PRINT "RECNO          TAGNO. SP.    DATE     G.    LAT.      LONG.  LENGTH  DAYS
  DIST."
150 FOR I=START TO FINISH STEP 2
160 READ# 1,I ; AA,BB,RNA$,RNO,G2,RF$,DD,MM,YY,N2$,L5,L6,E2$,XXX(),RNA2$,RNO2,DA
,DI,NNN
170 READ# 1,I+1 ; A$,S$,YYY(),NS$,L1,L2,EW$,ZZZ(),NA$,NO,F$,FL,C,CA,NA2$,NO2
180 L7=XXX(1) @ L8=XXX(2) @ W2=XXX(3) @ FL2=XXX(4) @ BW=XXX(5) @ P=XXX(6) @ FF=X
XX(7)
190 G=YYY(1) @ OP=YYY(2) @ D=YYY(3) @ M=YYY(4) @ Y=YYY(5) @ H=YYY(6) @ MI=YYY(7)

200 L3=ZZZ(1) @ L4=ZZZ(2) @ W=ZZZ(3) @ AT=ZZZ(4) @ WT=ZZZ(5) @ S=ZZZ(6) @ N=ZZZ(
7)
210 II=(I+1)/2
220 PRINT
230 PRINT USING 250 ; II,"RELEASE ",NA$,NO,F$,D,M,Y,G,NS$,L1,L2,EW$,L3,L4,FL
240 PRINT USING 250 ; II,"RECOVERY",RNA$,RNO,RF$,DD,MM,YY,G2,N2$,L5,L6,E2$,L7,L8
,FL2,DA,DI
250 IMAGE 4D,1X,8A,2X,1A,4Z,2X,1A,1X,3(1X,2Z),3D,2X,1A,2(1X,2Z),2X,2A,3Z,1X,2Z,1
X,5D,1X,5D,6D.D
260 NEXT I
270 ASSIGN# 1 TO  *
280 END
```

Out-put by "RECLST2"

RECNO		TAGNO.	SP.	DATE	G.	LAT.	LONG.	LENGTH	DAYS	DIST.
1	RELEASE	A0121	Y	19 12 83	1	S 00 14	E 129 55	45		
1	RECOVERY	A0121	Y	28 02 84	1	S 00 31	E 130 36	46	71	44.4
2	RELEASE	B0016	S	12 04 84	1	N 01 28	E 125 25	44		
2	RECOVERY	B0016	S	07 05 84	1	N 01 40	E 125 30	46	25	13.0
3	RELEASE	A5121	S	06 05 84	1	N 00 19	E 124 16	44		
3	RECOVERY	A5121	S	16 05 84	1	S 00 20	E 123 10	45	10	76.7

7. Program for calculating the number of days at liberty, "TAGDAY"

Using in dates of release and recovery, the number of days at sea, after tagging, is calculated. Consideration for leap year is not necessary

7.1 Procedure

(1). Load "TAGDAY"
(2). Press RUN key
(3). Put in the data as follows:

i. Day, month and year of release (in that order)
ii. Day, month and year of recovery (in that order)
Example June 10, 1984 ---→ 10, 6, 1984

7.2 Out-put

(1). Original (in-put) date
(2). Days at liberty (difference)

Example:

RELEASE	10	6	1984
RECOVERY	15	7	1984
NUMBER OF DAYS			35

7.3 Program listing

```
10 REM ********** PROGRAM NAME "TAGDAY" ********************
20 REM ********** CALCULATE OF DAY ..., "TAGDAY" **********
30 PRINTER IS 1
40 PRINTER IS 701
50 OPTION BASE 1
60 DIM YY(2),MM(2),DD(2),M(2),FM(2),JJ(2)
70 FOR I=1 TO 2
80     DISP "INPUT: DAY, MONTH, YEAR ";
90     INPUT DD(I),MM(I),YY(I)
100 NEXT I
110 FOR I=1 TO 2
120    M(I)=YY(I)*12+MM(I)-3
130    K1=IP ((367*M(I)+7)/12)
140    K2=IP (M(I)/12)
150    K3=IP (M(I)/48)
160    K4=IP (M(I)/1200)
170    K5=IP (M(I)/4800)
180    FM(I)=K1-2*K2+K3-K4+K5
190    JJ(I)=FM(I)+DD(I)+1721119
200 NEXT I.
210    IM3=JJ(2)-JJ(1)
220 PRINT USING JFORM ; "RELEASE          ",DD(1),MM(1),YY(1),"RECOVERY          ",DD
(2),MM(2),YY(2),"NUMBER OF DAYS",IM3
230 JFORM: IMAGE 5X,15A,2(4D),6D/5X,15A,2(4D),6D//5X,22A,7D
240 END
```

8. Program for calculating the distance and direction of movement, DISTANT. IPTP 01

Using the locations (Latitudes, Longtidues) of release and recovery, the distance between the two points and the direction of the movement are calculated.

The direction is indicated by degree in a clockwise direction at the point of release.

8.1 Procedure

(1). Load "DISTAN"
(2). Press RUN key
(3). Put in data as follow:

i Latitude and Longtude of release
ii Latitude and Longtude of recovery
Example N 00° 00', E 135° 00' --- 1, 0, 0, 1, 135, 0
S 01° 10', E 135° 00' --- 2,1, 10, 1, 135, 0
(North or East = 1, South or West = 2)

8.2 Out-put

(1). Original (in-put) data
(2). Distance (Unit: Nautical miles = 1852 m)
Direction (Unit: Degree)

---------- RAW DATA LIST ----------

RELEASE	N	0	19	E	124	16
RECOVERY	S	0	20	E	123	0

---------- RESULT ----------

DISTANCE	85.422
DIRECTION	242.836

8.3.2 Program listing

```
10 REM ********** PROGRAM NAME "DISTAN" **********************
20 REM ***** DISTANSES OF BETWEEN FROM RELEASE TO RECOVERY *****
30 PRINTER IS 701
40 OPTION BASE 1
50 PAI=3.1415926
60 DIM NSEW(2),LAT$(2),LONG$(2)
70 NSEW(1)=1
80 NSEW(2)=-1
90 LAT$(1)="N"
100 LAT$(2)="S"
110 LONG$(1)="E"
120 LONG$(2)="W"
130 DISP "CODE OF LATITUDE AND LONGITUDE"
140 DISP " 1 = NORTH        1 = EAST "
150 DISP " 2 = SOUTH        2 = WEST "
160 DISP "INPUT : POSITION OF RELEASE (CODE(1,2),XX,XX' AND CODE(1,2),XXX,XX')"
170 INPUT N1,NN1,NNN1,E1,EE1,EEE1
180 DISP "INPUT : POSITION OF RECOVERY (CODE(1,2),XX,XX' AND CODE(1,2),XXX,XX')"

190 INPUT N2,NN2,NNN2,E2,EE2,EEE2
200 NX1=(NN1+NNN1/60)*NSEW(N1)*PAI/180
210 NE1=(EE1+EEE1/60)*NSEW(E1)*PAI/180
220 NX2=(NN2+NNN2/60)*NSEW(N2)*PAI/180
230 NE2=(EE2+EEE2/60)*NSEW(E2)*PAI/180
240   A=SIN (NX1)
250   B=SIN (NX2)
260   P=COS (NX1)
270   Q=COS (NX2)
280   R=COS (NE2-NE1)
290   D=60*(ACS (A*B+P*Q*R)*180/PAI)
300   T=SIN (D/60*PAI/180)
310   S=COS (D/60*PAI/180)
320   DO=ACS ((B-A*S)/(T*P))*180/PAI
330   DDD=SIN (NE2-NE1)
340   IF DDD>= 0 THEN DOFUN=DO ELSE DOFUN=360-DO
350   DOSUU=DOFUN
360 PRINT
370 PRINT "---------- RAW DATA LIST ----------"
380 PRINT
390 PRINT USING FORM ; "RELEASE",LAT$(N1),NN1,NNN1,LONG$(E1),EE1,EEE1
400 FORM: IMAGE 5X,10A,2A,5D,5D,2X,2A,5D,5D
410 PRINT USING FORM ; "RECOVERY",LAT$(N2),NN2,NNN2,LONG$(E2),EE2,EEE2
420 PRINT
430 PRINT "----------    RESULT    ----------"
440 PRINT USING RFORM ; "DISTANCE",D
450 PRINT USING RFORM ; "DIRECTION",DOSUU
460 RFORM: IMAGE 5X,10A,7D.3D
470 END
```

SUPPLEMENT

COMPUTER PROGRAMS RELATED TO DATA ANALYSIS OF TAGGING

BY

M. Honma and T. Yonemori
GCP/RAS/099/JPN

P R E F A C E

This paper describe 6 computer programs, which were originally developed by Mr. M. Honma during this assignment as computer consultant to assist tuna scientists in the Institute for tagging data processing from August through October 1984.

Most of these programs, however, could be applicable not only for the tagging data analysis but also for gerenal purposes.

The configuration of the computer system used was as follows.

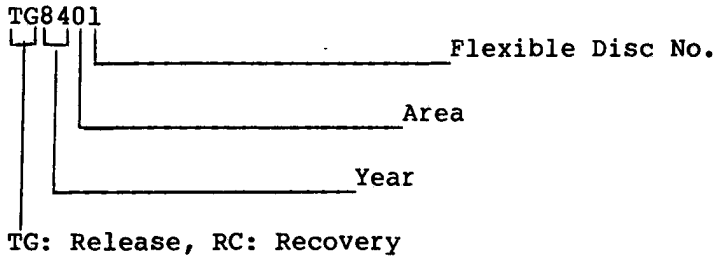
1. HEWLETT PACKARD 87 XM (124 KB)
2. HEWLETT PACKARD 7470 A PLOTTER
3. HEWLETT PACKARD 82901 M FLEXIBLE DISC DRIVE
4. HEWLETT PACKARD 82905 B PRINTER

Table of Contents

1. Initialization of flexible disk

1.1 Volume identification

Six figures (or characters) are used, as indicated below:



Ex: For 1984 Tagging, the volumn identifications are as follows:

"TG8401" ------ Tagging 1984, Sorong No. 1
"TG8411" ------ Tagging 1984, Ambon, No. 1
"TG8421" ------ Tagging 1984, Bitung, No. 1
"RC8485" ------ Recovery 1984 - 1985

1.2 Procedure

Insert the flexible disk into the left drive and type "INITIALIZE" TG8401", ": D700".

2. File Identification

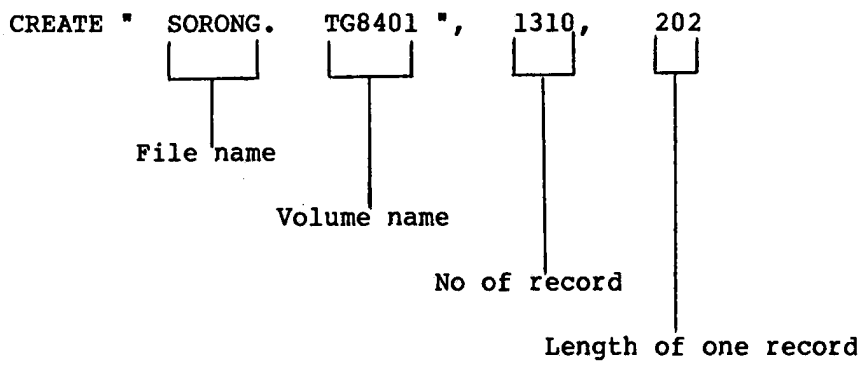
2.1 File name

File names correspond to area names and those in 1984 are as follows:

"SORONG"
"AMBON"
"BITUNG"
"RECOVE"

2.2 Procedure

Insert the flexible disk into the left drive and type as follows:



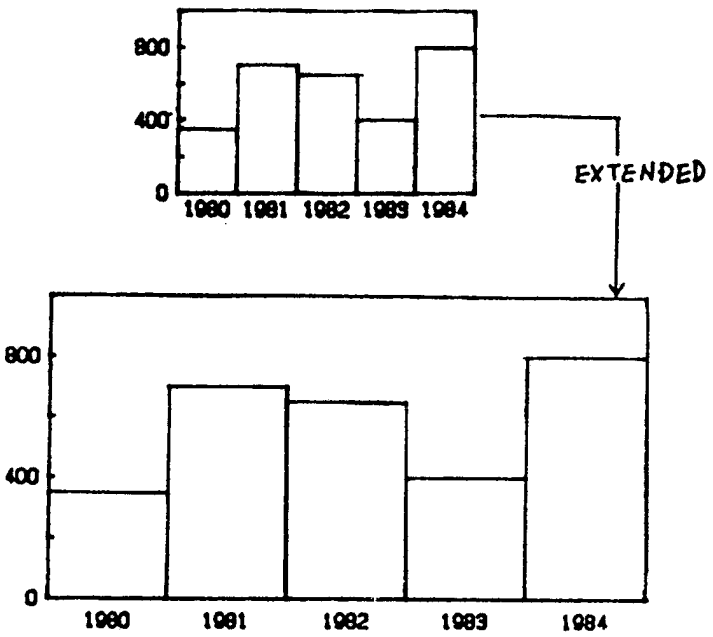
1. A program to draw LINE/BAR GRAPH PROGRAM, "FIG"

This program will draw line or bar graph.

1.1 Procedure

1). Load "FIG"
2). Press RUN key
3). In-put: total number of data
4). In-put: data Y_1, Y_2, Y_3, -----Y_T.
5). In-put: X axis -- MINIMUM, MAXIMUM, INTERVAL and LABEL UNIT.
6). In-put: Y axis -- " " " "
7). In-put: X and Y axes length (mm)
8). In-put: kind of figure (0 = Line, 1 = Bar).
9). In-put: question --- Figure OK ? (Yes = Y, No = N)
Answer = N -----→ go to 7)
Answer = Y -----→ finish

1.2 Example



1.3 Program listing

```
10 REM ********** PROGRAM NAME "FIG" **********
20 REM ********** OPTION FIGURE ***************
30 OPTION BASE 1
40 DIM Y(500)
50 PRINTER IS 701
60 DISP "INPUT ----> TOTAL OF DATA ";
70 INPUT TN
80 PRINT
90 PRINT "LAW DATA " @ PRINT
100 FOR I=1 TO TN @ INPUT Y(I)@ PRINT I;Y(I) @ NEXT I
110 YN$="Y"
120 DISP "INPUT : SCALE MINIMUM,MAXIMUM,INTERVAL AND LABEL ENTER UNIT FOR X-AXIS
"
130 INPUT XMI,XMA,INTERX,UNX
140 DISP "INPUT : SCALE MINIMUM,MAXIMUM,INTERVAL AND LABEL ENTER UNIT FOR Y-AXIS
"
150 INPUT YMI,YMA,INTERY,UNY
160 AGAIN:
170 DISP "INPUT ----> X AND Y AXIS LENGTH(mm)"
180 INPUT XX,YY
190 DISP "WHAT IS KIND FIGURE ? ---> INPUT: LINE=0  BAR=1"
200 INPUT LB
210 XR=(45+XX)/150*100
220 YR=(45+YY)/150*100
230 PLOTTER IS 705
240 GCLEAR
250 LIMIT 0,200,0,150
260 LOCATE 30,XR,30,YR
270 FRAME
280 SCALE XMI,XMA+INTERX,YMI,YMA
290 AXES INTERX,INTERY,XMI,YMI,INTERX,INTERY,2
300 FOR J=YMI TO YMA STEP INTERY*UNY
310 LDIR 0 @ LORG 8
320 MOVE XMI,J
330 LABEL J
340 NEXT J
350 IF LB=1 THEN JJ=INTERX/2 ELSE JJ=0
360 FOR J=XMI TO XMA STEP INTERX*UNX
370 XL$=VAL$ (IP (J))
380 MOVE J+JJ,YMI-INTERY/3
390 LDIR 0 @ LORG 6
400 LABEL XL$
410 NEXT J
420 IF LB=0 THEN GOTO 480
430 FOR I=1 TO TN
440 CLIP I*INTERX-INTERX+XMI,I*INTERX+XMI,YMI,Y(I)
450 FRAME
460 NEXT I
470 GOTO 530
480 MOVE XMI,Y(1)
490 FOR I=2 TO TN
500 PLOT I*INTERX-INTERX+XMI,Y(I),-1
510 NEXT I
520 PEN UP
530 DISP "FIGURE SIZE AND TYPE OK ?      YES=Y     NO=N"
540 INPUT YESNO$
550 IF YESNO$=YN$ THEN GOTO 580
560 DISP "AGAIN STROKE FIGURE----------> PLEASE CHANGE SET PAPER"
570 GOTO AGAIN
580 END
```

2. A Program to determine size composition "FREQUENCY

This program will determine frequency distribution (in munber and %) from length (or weight) data.

2.1 Procedure

1). Load "FREQUENCY"
2). In-put: START LENGTH (or WEIGHT), MAXIMUM LENGTH (or WEIGHT) and INTERVAL.
3). In-put: When there are no sex data, press 0 and put in only length data.
When there are sex data, press 1 and put in length and sex
(male = 1, female = 2, ? = 3)
4). In-put: Repeat 3
When wrong data were put in, the data are eliminated by putting in the same figures with (-)
Case No = 0 ------→ - XX
Case Yes = 1 ------→ - XX, X
length, sex
5). In-put: Data end sign is 999
Case No = 0 ------→ 999
Case Yes = 0 ------→ 999,1.
6). Out-put: Raw data, size composition by sex and % size composition by sex.
7). In-put: Continue data in-put ? ----→ No = 0 or Yes = 1
(No = 0 --→ finish,
Yes = 1 --→ again go to 2)

2.2 Example of out-put

RAW DATA

20 1
21 1
22 1
24 1
26 1
25 2
27 2
29 2
30 3
22 3

LENGTH (cm)	FREQUENCY MALE	FEMALE	UNKNOWN	TOTAL	PERCENTAGE MALE	FEMALE	UNKNOWN	TOTAL
20 - 21	2	0	0	2	40.00	0.00	0.00	20.00
22 - 23	1	0	1	2	20.00	0.00	50.00	20.00
24 - 25	1	1	0	2	20.00	33.33	0.00	20.00
26 - 27	1	1	0	2	20.00	33.33	0.00	20.00
28 - 29	0	1	0	1	0.00	33.33	0.00	10.00
30 - 31	0	0	1	1	0.00	0.00	50.00	10.00
TOTAL	5	3	2	10				

2.3 Program listing

```
10 REM ********* PROGRAM NAME "FREQUENCY" **********
20 REM ********* CASE BY LENGTH FOR EACH FISH ******
30 OPTION BASE 1
40 PRINTER IS 701
50 DIM F(150),M(150),U(150),T(150),PF(150),PM(150),PU(150),PT(150)
60 START:
70 FOR I=1 TO 150
80 F(I)=0 @ M(I)=0 @ T(I)=0 @ U(I)=0
90 PF(I)=0 @ PM(I)=0 @ PT(I)=0 @ PU(I)=0
100 NEXT I
110 FT=0 @ MT=0 @ TT=0 @ KK=0 @ UT=0
120 PRINT
130 PRINT "RAW  DATA"
140 PRINT
150 DISP "INPUT ----> STARTING LENGTH,MAXIMUM LENGTH AND INTERVAL"
160 INPUT SL,ML,INTER@ C=SL/INTER-1 @ C=IP (C)
170 DISP "INPUT ----> DATA BY SEX ?    NO=0,  YES=1"
180 INPUT SEX
190 IF SEX=0 THEN GOTO TOTAL
200 FOR I=1 TO 3000
210 A=1
220 DISP "F.L., SEX ..... MALE=1, FEMALE=2, UNKNOWN=3"
230 INPUT FL,SE
240 IF FL=999 THEN GOTO OUTPUT
250 IF FL<0 THEN A=-1
260 PRINT FL;SE
270 K=FL/INTER*A-C @ K=IP (K)
280 GOSUB EEE
290 IF FL<0 THEN GOTO 320
300 IF FL>ML THEN GOTO 390
310 IF FL<SL THEN GOTO 390
320 IF SE=1 THEN GOTO 340
330 IF SE=2 THEN GOTO 350 ELSE GOTO 360
340 M(K)=M(K)+A @ MT=MT+A @ GOTO 370
350 F(K)=F(K)+A @ FT=FT+A @ GOTO 370
360 U(K)=U(K)+A @ UT=UT+A
370 T(K)=T(K)+A @ TT=TT+A
380 IF KK<K THEN KK=K
390 NEXT I
400 !
410 TOTAL:
420 FOR I=1 TO 3000
430 A=1
440 DISP "FL .......FL. ONLY INPUT"
450 INPUT FL@ IF FL=999 THEN GOTO OUTPUT
460 IF FL<0 THEN A=-1
470 PRINT FL
480 K=FL/INTER*A-C @ K=IP (K)
490 GOSUB EEE
500 IF FL<0 THEN GOTO 530
510 IF FL>ML THEN GOTO 550
520 IF FL<SL THEN GOTO 550
530 T(K)=T(K)+A @ TT=TT+A
540 IF KK<K THEN KK=K
550 NEXT I
560 !
570 OUTPUT:
580 IF MT=0 THEN GOTO 600
```

(continued)

```
590 FOR I=1 TO KK @ PM(I)=M(I)*100/MT @ NEXT I
600 IF FT=0 THEN GOTO 620
610 FOR I=1 TO KK @ PF(I)=F(I)*100/FT @ NEXT I
620 IF UT=0 THEN GOTO 640
630 FOR I=1 TO KK @ PU(I)=U(I)*100/UT @ NEXT I
640 FOR I=1 TO KK @ PT(I)=T(I)*100/TT @ NEXT I
650 PRINT
660 PRINT "        LENGTH              FREQUENCY                                  PERCENTAGE"
670 PRINT "         (cm)       MALE FEMALE UNKNOWN TOTAL       MALE FEMALE UNKNOWN T
OTAL"
680 PRINT
690 FOR I=1 TO KK
700 L1=(I+C)*INTER
710 L2=L1+INTER-1
720 PRINT USING FORM ; L1,L2,M(I),F(I),U(I),T(I),PM(I),PF(I),PU(I),PT(I)
730 FORM: IMAGE 4X,4D," -",4D,3(6D),8D,5X,4(4D.2D)
740 NEXT I
750 PRINT
760 PRINT USING 770 ; MT,FT,UT,TT
770 IMAGE 9X,"TOTAL",3(6D),8D
780 DISP "INPUT ----> DATA AGAIN ?        NO=0,    YES=1"
790 INPUT NY
800 IF NY=0 THEN GOTO FINISH
810 GOTO START
820 FINISH:
830 END
840 EEE:
850 IF FL>ML THEN PRINT "*** ERROR DATA ***"
860 IF FL<SL THEN PRINT "*** ERROR DATA ***"
870 RETURN
```

3. A program for regression analysis and calculate correlation coefficient, "STTRLIN"

This program will draw and analyse correlation between variables X and Y from a data set,
(X_1,Y_1), (X_2,Y_2), --- (X_n,Y_n) --

3.1 Equation (regression): $Y_i = \alpha + \beta(X_i - \bar{X}) + \varepsilon_i$

3.2 Preparation
1). Programme is filed in "IPTP01"
2). Set pen and paper in the plotter

3.3 Procedure
1). Load "STRLIN"
2). Press RUN key
3). Enter number of data set (n)
4). Enter data: X_1,Y_1 input
X_2,Y_2 input
. .
. .
. .
X_n,Y_n input
5). Display maximum X and Y values. Referring to those values, enter following data.

i) Maximum values of X-axis and labelling interval of obscissa
ii) Maxmium values of Y-axis and labelling interval of ordinate

3.4 Out-put
1). In-put data
2). Result of analysis of variance: The result is tested by F-Table.
3). Regression (Y=a+bx)
4). Correlation coefficient
5). Data plotting and regression

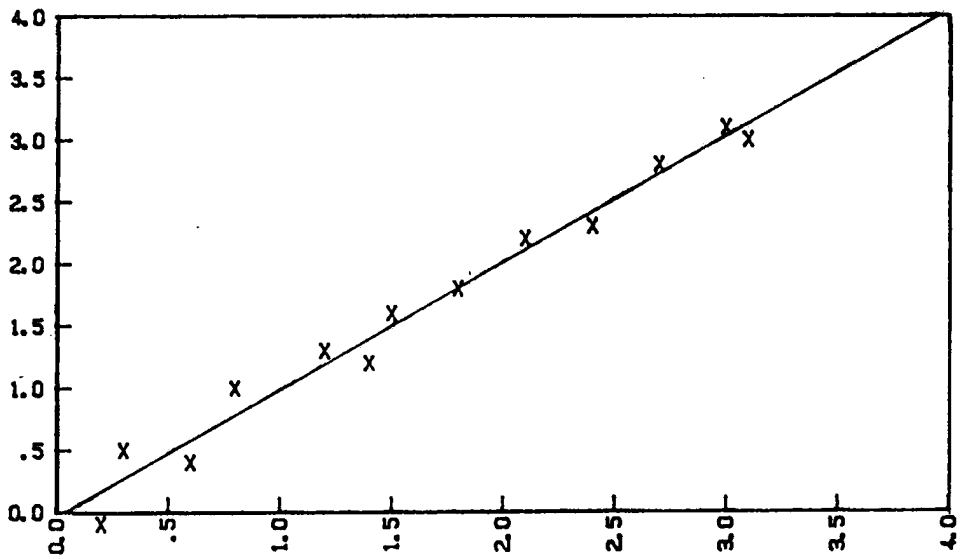
3.5 Example of Out-put

---------- RAW DATA LIST ----------

1	X =	.20	Y =	-.10
2	X =	.30	Y =	.50
3	X =	.60	Y =	.40
4	X =	.80	Y =	1.00
5	X =	1.20	Y =	1.30
6	X =	1.40	Y =	1.20
7	X =	1.50	Y =	1.60
8	X =	1.80	Y =	1.80
9	X =	2.10	Y =	2.20
10	X =	2.40	Y =	2.30
11	X =	2.70	Y =	2.80
12	X =	3.00	Y =	3.10
13	X =	3.10	Y =	3.00

FACTOR	S	FD	V	FO
R	12.3688	1	12.3688	432.8988
E	.3143	11	.0286	
T	12.6831	12		

********** Y = -.03563 + 1.02195X **********
********** R = .98753 **********



3.6 Program listing

```
 10 REM ********** PROGRAM NAME "STRLIN" **********
 20 REM ***** STRAIGHT LINE.....Y = A + BX ********
 30 OPTION BASE 1
 40 DIM X(500),Y(500)
 50 PRINTER IS 701
 60 SX=0
 70 SXX=0
 80 SY=0
 90 SYY=0
100 SXY=0
110 NUMB=0
120 DISP "INPUT : TOTAL NUMBER OF DATA";
130 INPUT TN
140 DISP "INPUT : FARST DATA IS X(I), NEXT DATA IS Y(I)";
150 PRINT "       ---------------- RAW DATA LIST ----------------"
160 FOR I=1 TO TN
170 INPUT X(I),Y(I)
180       SX=SX+X(I)
190       SXX=SXX+X(I)*X(I)
200       SY=SY+Y(I)
210       SYY=SYY+Y(I)*Y(I)
220       SXY=SXY+X(I)*Y(I)
230       NUMB=NUMB+1
240 PRINT USING FORM ; NUMB," X = ",X(I)," Y = ",Y(I)
250 FORM: IMAGE 5X,3D,5X,5A,5D.DD,5X,5A,5D.DD
260 NEXT I
270       SXX=SXX-SX*SX/NUMB
280       SYY=SYY-SY*SY/NUMB
290       SXY=SXY-SX*SY/NUMB
300       SR=SXY*SXY/SXX
310       SE=SYY-SR
320       ST=SYY
330       PT=NUMB-1
340       PE=NUMB-2
350       PR=1
360       VR=SR/PR
370       VE=SE/PE
380       IF VE=0 THEN GOTO 400
390       FO=VR/VE
400       IF VE=0 THEN FO=0
410 PRINT
420 PRINT USING VFORM ; "FACTOR"," S","FD","    V","    FO"
430 VFORM: IMAGE 5X,10A,10A,3A,10A,10A
440 PRINT USING NFORM ; "R",SR,PR,VR,FO
450 NFORM: IMAGE 7X,5A,5D.4D,5D,5D.4D,5D.4D
460 PRINT USING NFORM ; "E",SE,PE,VE
470 PRINT USING NFORM ; "T",ST,PT
480 XBAR=SX/NUMB
490 B=SXY/SXX
500 A=SY/NUMB-B*XBAR
510 PRINT
520 PRINT USING FFORM ; "********** Y =",A," +",B,"X"," **********"
530 FFORM: IMAGE 5X,14A,5D.5D,2A,5D.5D,1A,11A
540 !
550 RR=SXY/SQR (SXX*SYY)
560 PRINT USING RFORM ; "********** R = ";RR;" **********"
570 PRINT
580 RFORM: IMAGE 5X,22A,2D.5D
590 XMA=0
600 YMA=0
610 FOR I=1 TO TN
620     IF XMA<X(I) THEN XMA=X(I)
630     IF YMA<Y(I) THEN YMA=Y(I)
640 NEXT I
650 DISP "X-MAXIMUM..... ";XMA
660 DISP "Y-MAXIMUM..... ";YMA
670 !
```

(continued)

```
680 DISP "INPUT : SCALE MAXIMUM AND INTERVAL OF X-AXIS"
690 INPUT XMA,INTERX
700 DISP "INPUT : SCALE MAXIMUM AND INTERVAL OF Y-AXIS"
710 INPUT YMA,INTERY
720 PLOTTER IS 705
730 GCLEAR
740 LIMIT 0,200,0,150
750 LOCATE 30,130,30,85
760 FRAME
770 SCALE 0,XMA,0,YMA
780 FXD 1,1
790 LAXES INTERX,INTERY,0,0,INTERX,INTERY,3
800 XY$="X"
810 FOR I=1 TO TN
820     MOVE X(I),Y(I)
830 LORG 5
840     LABEL XY$
850 NEXT I
860 !
870 MOVE 0,A
880 FOR I=INTERX TO XMA STEP INTERX
890     Z=A+B*I
900     PLOT I,Z,-1
910 NEXT I
920 PEN UP
930 END
 122503
```

4. A program to determine Von Bertalanffy growth curve by Fabens method, "FABENS"

This program will determine growth curve from length data at release and recovery using Fabens method.

4.1 Procedure

1). Road "FABENS"
2). Press RUN key
3). Input: Any TITLE
4). Input: Total number of data
5). Input: l_t (length at release), l_{t+d} (length at recovery) and d (days after release)
6). Input: Number of set of (age, l_a). When data are not available, press 1.
7). Input: (age, l_a) data. When data are not available, press 0,0.

4.2 Output: Following 3 parts are put out.

1). PART A
    Input data
  Parameter

2). PART B
    Input data
    Parameter

3). PART C
    Result of calculated values for plotting.

4.3 Example of Out-put

```
FITTING THE DECAYING EXPONENTIAL GROWTH CURVE
     SKIPJACK

    X=A(1-B*EXP(-KT)) WHERE X IS LINEAR SIZE

-------------------------------------------------
PART A     4 OBSERVATIONS

   NO.  LENGTH AT RELEASE   LENGTH AT RECAPTURE  TIME INTERVAL

    1          28.00              45.00             365.00
    2          45.00              63.00             365.00
    3          63.00              73.00             365.00
    4          73.00              77.00             365.00

     FINAL VALUES(   4 ITERATIONS)
K=       .000965     A=      93.5209
1 CHRON =        718.459 UNIT OF TIME
1 UNIT OF TIME =         .001392 CHRONS

-------------------------------------------------

PART B         6 OBSERVATIONS

  NO.       AGE        LENGTH

    1     0.0000          .2500
    2     1.0000        28.0000
    3     2.0000        45.0000
    4     3.0000        63.0000
    5     4.0000        73.0000
    6     5.0000        77.0000
B=    1.01855          TO=   19.04837
```

PART C CALCULATED VALUES FOR PLOTTING

AGE	LENGTH
0.0000	-1.7346
50.0000	2.7514
100.0000	7.0260
150.0000	11.0994
200.0000	14.9809
250.0000	18.6796
300.0000	22.2042
350.0000	25.5627
400.0000	28.7631
450.0000	31.8128
500.0000	34.7188
600.0000	40.1268
700.0000	45.0374
800.0000	49.4964
900.0000	53.5453
1000.0000	57.2218
1100.0000	60.5602
1200.0000	63.5916
1300.0000	66.3442
1400.0000	68.8436
1600.0000	73.1740
1800.0000	76.7445
2000.0000	79.6884
2200.0000	82.1157
2400.0000	84.1171
2600.0000	85.7673
2800.0000	87.1279
3000.0000	88.2498
3500.0000	90.2670
4000.0000	91.5122
4500.0000	92.2810
5000.0000	92.7555
5500.0000	93.0484
6000.0000	93.2292
6500.0000	93.3409
7000.0000	93.4098
ASYMPTOTE	93.5209
EXTRAP ZERO	-.02651 CHRONS BEFORE BIRTH

4.4 Program listing

```
10 REM **********.PROGRAM NAME "FABENS" **********
20 REM *** ESTIMATION OF PARAMETERS IN VBGC BY MEANS OF FABENSS METHOD ***
30 OPTION BASE 1
40 DIM X(401),Y(401),D(401),NAME$[70]
50 PRINTER IS 701
60 !
70 ! ****** READ PART A DATA *****
80 !
90 DISP "INPUT ----> ANY TITLE"
100 INPUT NAME$ ! OPTION TITLE
110 PRINT
120 PRINT "FITTING THE DECAYING EXPONENTIAL GROWTH CURVE"
130 PRINT USING 140 ; NAME$
140 IMAGE 5X,70A
150 PRINT
160 PRINT "    X=A(1-B*EXP(-KT)) WHERE X IS LINEAR SIZE"
170 PRINT
180 DISP "INPUT ----> TOTAL OF DATA"
190 INPUT NN
200 FOR N=1 TO NN
210 DISP "INPUT ----> X=RELEASE LENGTH, Y=RECOVERY LENGTH, D=PERIOD"
220 INPUT X(N),Y(N),D(N)
230 NEXT N
240 DISP "PLEASE WAITING"
250 !
260 ! ***** WRITE PART A DATA *****
270 !
280 PRINT "------------------------------------------------"
290 PRINT USING 300 ; NN
300 IMAGE "PART A",5D," OBSERVATIONS"
310 PRINT
320 PRINT
330 PRINT "   NO.  LENGTH AT RELEASE   LENGTH AT RECAPTURE  TIME INTERVAL"
340 PRINT
350 FOR JD=1 TO NN
360 PRINT USING 370 ; JD,X(JD),Y(JD),D(JD)
370 IMAGE 5D,3(13D.2D,4X)
380 NEXT JD
390 PRINT
400 !
410 ! ***** CALCULATE INITIAL K *****
420 !
430 F=0 @ G=0 @ H=0 @ P=0 @ M=0
440 FOR I=1 TO NN
450 R=(Y(I)-X(I))/D(I)
460 S=(X(I)+Y(I))/2
470 F=F+R
480 G=G+S
490 H=H+R*S
500 P=P+S*S
510 NEXT I
520 T=NN
530 C=(F*G-T*H)/(T*P-G*G)
540 AP=1
550 !
560 ! ***** ITERATE TO IMPROVE K AND A
570 !
580 F=0 @ G=0 @ H=0 @ M=M+1
590 FOR I=1 TO NN
600 P=EXP (-(C*D(I)))
610 F=F+(1-P)*(Y(I)-X(I)*P)
620 G=G+(1-P)*(1-P)
630 NEXT I
640 A=F/G
650 FOR I=1 TO NN
660 P=EXP (-(C*D(I)))
```

(continued)

```
670 H=H+D(I)*P*(Y(I)+X(I)*(1-2*P)-2*A*(1-P))
680 NEXT I
690 DA=H/G
700 F=0 @ G=0
710 FOR I=1 TO NN
720 P=EXP (-(C*D(I)))
730 R=A-X(I)
740 S=A-Y(I)
750 F=F+D(I)*P*R*(S-R*P)
760 G=G+D(I)*P*(DA*(S+R*(1-2*P))+D(I)*R*(2*P*R-S))
770 NEXT I
780 CP=C-F/G
790 F=(CP-C)/C
800 C=CP
810 G=(A-AP)/AP
820 AP=A
830 IF F-.000002<= 0 THEN GOTO 840 ELSE GOTO 870
840 IF -F-.000002<= 0 THEN GOTO 850 ELSE GOTO 870
850 IF G-.000002<= 0 THEN GOTO 860 ELSE GOTO 870
860 IF -G-.000002<= 0 THEN GOTO 950 ELSE GOTO 870
870 IF M-15>= 0 THEN GOTO 880 ELSE GOTO 580
880 PRINT USING 890 ; M,C,A
890 IMAGE "AFTER",5D,"   ITERATIONS K = ",10D,"  A = ",7D.7D
900 IF M-20>= 0 THEN GOTO 910 ELSE GOTO 580
910 PRINT
920 PRINT USING 890 ; F,G
930 PRINT
940 GOTO 970
950 PRINT USING 1010 ; M
960 PRINT USING 1020 ; C,A
970 F=.69314718/C
980 PRINT USING 1030 ; F
990 F=1/F
1000 PRINT USING 1040 ; F
1010 IMAGE "     FINAL VALUES(",4D," ITERATIONS)"
1020 IMAGE "K= ",5D.6D,5X,"A=",7D.4D
1030 IMAGE "1 CHRON =  ",8D.3D," UNIT OF TIME"
1040 IMAGE "1 UNIT OF TIME =",8D.6D," CHRONS"
1050 PRINT
1060 PRINT
1070 PRINT "-------------------------------------------------"
1080 !
1090 ! ***** PART B READ PART B DATA *****
1100 !
1110 DISP "INPUT ----> NUMBER OF AGE-LENGTH DATA"
1120 INPUT N
1130 DISP "INPUT ----> X=AGE,  D=LENGTH"
1140 FOR I=1 TO N
1150 INPUT X(I),D(I)
1160 NEXT I
1170 PRINT
1180 PRINT USING 1190 ; N
1190 IMAGE "PART B",5X,4D," OBSERVATIONS"
1200 !
1210 ! ***** WRITE DATA b *****
1220 !
1230 PRINT
1240 PRINT "  NO.       AGE        LENGTH"
1250 PRINT
```

(continued)

```
1260 FOR JDD=1 TO N
1270 PRINT USING 1280 ; JDD,X(JDD),D(JDD)
1280 IMAGE 5D,5D.4D,5X,5D.4D
1290 NEXT JDD
1300 !
1310 ! ***** CALCULATE B
1320 !
1330 F=0 @ G=0
1340 FOR I=1 TO N
1350 P=EXP (-(C*D(I)))
1360 F=F+(A-X(I))*P
1370 G=G+P*P
1380 NEXT I
1390 B=F/(A*G)
1400 TO=LOG (B)/C
1410 PRINT USING 1420 ; B,TO
1420 IMAGE "B=",4D.5D,10X,"TO=",4D.5D
1430 PRINT
1440 !
1450 ! ***** PART C *****
1460 !
1470 PRINT
1480 PRINT "-------------------------------------------------------"
1490 PRINT "PART C CALCULATED VALUES FOR PLOTTING"
1500 PRINT
1510 PRINT
1520 PRINT "         AGE            LENGTH"
1530 PRINT
1540 P=.1/C
1550 I=LGT (P)-1.0000005 @ I=IP (I)
1560 N=P*.1^I
1570 N=IP (N)
1580 P=N
1590 P=P*10^I
1600 R=0
1610 T=R*P
1620 S=A*(1-B*EXP (-(C*T)))
1630 PRINT USING 1640 ; T,S
1640 IMAGE 5X,4D.4D,5X,5D.4D
1650 IF R-5>= 0 THEN GOTO 1670 ELSE GOTO 1660
1660 R=R+.5 @ GOTO 1610
1670 IF R-14>= 0 THEN GOTO 1690 ELSE GOTO 1680
1680 R=R+1 @ GOTO 1610
1690 IF R-30>= 0 THEN GOTO 1710 ELSE GOTO 1700
1700 R=R+2 @ GOTO 1610
1710 IF R-70>= 0 THEN GOTO 1730 ELSE GOTO 1720
1720 R=R+5 @ GOTO 1610
1730 PRINT
1740 PRINT USING 1750 ; A
1750 IMAGE 5X,"ASYMPTOTE",5X,7D.4D
1760 F=-(LOG (B)/.69314718)
1770 PRINT
1780 PRINT USING 1790 ; F
1790 IMAGE 5X,"EXTRAP ZERO",7D.5D," CHRONS BEFORE BIRTH"
1800 END
 110161
```

5. A program to calculate exploitation rate, "EXPLOI"

This program will analyse correlation between population size, catch and natural mortality.

Catch equation is usually formulated as follows:

$$C = N \cdot \frac{F}{Z}\left(1 - e^{-Z}\right) \qquad \text{----- 1}$$

Catch rate, $E = C/N$

$$E = C/N = \frac{F}{Z}\left(1 - e^{-Z}\right) \qquad \text{----- 2}$$

Z = F + M
F = Fishing mortality coefficient
M = Natural mortality coefficient
C = Catch in number
N = Population size in number

Here, unknow parameters are N, F and M. Relation between F and M is obtained from E - Table. If F and M are assumed from another sources, E and N are estimated from E - table.

5.1 Procedure

1). Load "EXPLOI"
2). Press RUN key
3). Put in the range of M to be calculated

5.2 Out-put

Against given M values, E values are calculated for F values ranging from 0.00 to 3.99 with 0.01 interval.

5.3 Program listing

```
10 REM ***** PROGRAM NAME "EXPLOI" *****
20 REM ***** EPLOITATION RATE **********
30 CLEAR
40 OPTION BASE 1
50 PRINTER IS 7,01
60 DIM Y(10),NUMB(10)
70 FOR I=0 TO 9
80 NUMB(I+1)=I/100
90 NEXT I
100 DISP "RANGE OF NATURAL MORTALITY OF FROM START TO END"
110 INPUT MSTA,MEND
120 PRINT
130 PRINT "***** EXPLOITATION***** E=F/(F+M)(1-EXP(-(F+M)))*****"
140 FOR M=MSTA TO MEND STEP .1
150 PRINT
160 PRINT "   NATURAL MORTALITY =  ";M
170 PRINT
180 PRINT " F";
190 FOR I=1 TO 10
200 PRINT TAB (7*I);NUMB(I);
210 NEXT I
220 PRINT
230 PRINT
240 LLINE=0
250 FOR F=0 TO 3.9 STEP .1
260 F1=F
270 F2=F+.09
280 K=0
290 FOR FF=F1 TO F2 STEP .01
300 Z=M+FF
310 K=K+1
320 Y(K)=FF/Z*(1-EXP (-Z))
330 NEXT FF
340 PRINT USING 350 ; F1,Y(1),Y(2),Y(3),Y(4),Y(5),Y(6),Y(7),Y(8),Y(9),Y(10)
350 IMAGE D.DD,2X,10(D.5D)
360 LLINE=LLINE+1
370 IF LLINE MOD 5=0 THEN PRINT
380 NEXT F
390 NEXT M
400 END
```

5.4 Example of Out-put

***** EXPLOITATION***** E=F/(F+M)(1-EXP(-(F+M)))*****

NATURAL MORTALITY = 0

F	0	.01	.02	.03	.04	.05	.06	.07	.08	.09
0.00	0.00000	.00995	.01980	.02955	.03921	.04877	.05824	.06761	.07688	.08607
.10	.09516	.10417	.11308	.12190	.13064	.13929	.14786	.15634	.16473	.17304
.20	.18127	.18942	.19748	.20547	.21337	.22120	.22895	.23662	.24422	.25174
.30	.25918	.26655	.27385	.28108	.28823	.29531	.30232	.30927	.31614	.32294
.40	.32968	.33635	.34295	.34949	.35596	.36237	.36872	.37500	.38122	.38737
.50	.39347	.39950	.40548	.41140	.41725	.42305	.42879	.43447	.44010	.44567
.60	.45119	.45665	.46206	.46741	.47271	.47795	.48315	.48829	.49338	.49842
.70	.50341	.50836	.51325	.51809	.52289	.52763	.53233	.53699	.54159	.54616
.80	.55067	.55514	.55957	.56395	.56829	.57259	.57684	.58105	.58522	.58934
.90	.59343	.59748	.60148	.60545	.60937	.61326	.61711	.62092	.62469	.62842
1.00	.63212	.63578	.63941	.64299	.64655	.65006	.65354	.65699	.66040	.66378
1.10	.66712	.67044	.67372	.67697	.68018	.68336	.68651	.68962	.69272	.69578
1.20	.69881	.70180	.70477	.70771	.71062	.71350	.71635	.71917	.72196	.72473
				.73552	.73815	.74				

NATURAL MORTALITY = .6

F	0	.01	.02	.03	.04	.05	.06	.07	.08	.09
0.00	0.00000	.00749	.01491	.02226	.02954	.03677	.04392	.05102	.05805	.06501
.10	.07192	.07876	.08554	.09226	.09892	.10553	.11207	.11856	.12498	.13135
.20	.13767	.14393	.15013	.15628	.16237	.16841	.17439	.18033	.18621	.19203
.30	.19781	.20354	.20921	.21484	.22041	.22594	.23142	.23684	.24223	.24756
.40	.25285	.25809	.26328	.26843	.27354	.27860	.28361	.28859	.29351	.29840
.50	.30324	.30804	.31280	.31752	.32219	.32683	.33142	.33598	.34049	.34497
.60	.34940	.35380	.35816	.36248	.36677	.37102	.37523	.37940	.38354	.38765
.70	.39171	.39575	.39974	.40371	.40764	.41153	.41540	.41922	.42302	.42678
.80	.43052	.43422	.43788	.44152	.44513	.44870	.45224	.45576	.45924	.46270
.90	.46612	.46952	.47288	.47622	.47953	.48282	.48607	.48930	.49250	.49567
1.00	.49881	.50193	.50503	.50809	.51113	.51415	.51714	.52011	.52305	.52596
1.10	.52885	.53172	.53456	.53738	.54018	.54295	.54570	.54842	.55113	.55381
1.20	.55647	.55910	.56172	.56431	.56688	.56943	.57196	.57447	.57696	.57943
1.30	.58187	.58430	.58671	.58909	.59146	.59381	.59614	.59845	.60074	.60301
1.40	.60527	.60750	.60972	.61192	.61410	.61626	.61841	.62053	.62265	.62474
1.50	.62682	.62888	.63092	.63295	.63496	.63695	.63893	.64089	.64284	.64477
1.60	.64669	.64859	.65047	.65234	.65420	.65604	.65787	.65968	.66147	.66326
1.70	.66503	.66679	.66853	.67025	.67196	.67366	.67535	.67702	.67868	.68032
1.80	.68194	.68355	.68514	.68672	.68828	.68982	.69135	.69287	.69437	.69586
1.90	.69733	.69879	.70023	.70166	.70307	.70447	.70586	.70723	.70859	.70993

6. A program to determine lenght frequency distribution (in number and %) of tagged fish, "TAGFRQ"

This program will tabulate length frequency distribution of tagged fish.

6.1 Procedure

1). Load "TAGRFQ"
2). Press RUN key
3). Put in FILE NAME to be assigned
4). Enter AREA NAME
5). ENTER SPECIES NAME
6). Enter length interval

6.2 Program listing

```
10 REM ********** PROGRAM NAME "TAGFRQ" **************************
20 REM ********** TAG RELEASE LENGTH FREQUENCY PROGRAM ***********
30 OPTION BASE 1
40 DISP "PLEASE ASSIGN FILE SELECT AND INPUT"
50 DISP "      SORONG.TG8401"
60 DISP "      SORONG.TG8402"
70 DISP "      SORONG.TG8403"
80 DISP "      AMBON.TG8411"
90 DISP "      AMBON.TG8412"
100 DISP "      BITUNG.TG8421"
110 DISP "      BITUNG.TG8422"
120 INPUT FINAME$
130 ASSIGN# 1 TO  FINAME$
140 PRINTER IS 701
150 DIM A$(1)[6],S$(1)[2],NS$(1)[1],EW$(1)[1],NA$(1)[1],F$(1)[1],NA2$(1)[1]
160 DIM YYY(7),ZZZ(7),LLL(200),AREA$(1)[6],SP$(1)[1]
170 DISP "SELECT ITEM INPUT"
180 DISP "WHAT IS AREA ?"
190 INPUT AREA$
200 DISP "WHAT IS SPECIES ?      Y = YELLOWFIN,  S = SKIPJACK"
210 INPUT SP$
220 IF SP$="Y" THEN SSP$="YELLOWFIN" ELSE SSP$="SKIPJACK"
230 DISP "LENGTH INTERVAL ?"
240 INPUT CLASS
250 FOR I=1 TO 200
260 LLL(I)=0
270 NEXT I
280 LMIN=1000
290 LMAX=0
300 LLT=0
310 DISP "INPUT ----> RECORD NUMBER OF RANGE"
320 INPUT START,FINISH
330 FOR I=START TO FINISH
340 READ# 1,I ; A$,S$,YYY(),NS$,L1,L2,EW$,ZZZ(),NA$,NO,F$,FL,C,CA,NA2$,NO2
350 G=YYY(1) @ OP=YYY(2) @ D=YYY(3) @ N=YYY(4) @ Y=YYY(5) @ H=YYY(6) @ MI=YYY(7)

360 L3=ZZZ(1) @ L4=ZZZ(2) @ W=ZZZ(3) @ AT=ZZZ(4) @ WT=ZZZ(5) @ S=ZZZ(6) @ N=ZZZ(
7)
370 IF A$=AREA$ THEN GOTO 380 ELSE GOTO 470
380 IF F$=SP$ THEN GOTO 390 ELSE GOTO 470
390 FL=(FL+CLASS-1)/CLASS
400 FL=IP (FL)
```

(continued)

```
410 IF FL<1 THEN GOTO 470
420 IF FL>200 THEN GOTO 470
430 LLL(FL)=LLL(FL)+1
440 LLT=LLT+1
450 IF LMIN>= FL THEN LMIN=FL
460 IF LMAX<= FL THEN LMAX=FL
470 NEXT I
480 PRINT
490 PRINT "SIZE COMPOSITION.... ";SSP$
500 PRINT
510 PRINT " LENGTH  FREQUENCY  PERCENT"
520 PRINT
530 FOR I=LMIN TO LMAX
540 II=I*CLASS
541 III=II-CLASS+1
550 PP=LLL(I)*100/LLT
560 PRINT USING 570 ; III,"-",II," -- ",LLL(I),"  -- ",PP
570 IMAGE 3D,1A,3D,4A,5D,5A,3D.2D
580 NEXT I
590 PRINT
600 PRINT USING 610 ; " TOTAL --  ",LLT
610 IMAGE 11A,4D
620 ASSIGN# 1 TO  *
630 END
```

6.3 Example of out-put

SIZE COMPOSITION.... YELLOWFIN

LENGTH	FREQUENCY	PERCENT
31- 31 --	1 --	5.26
32- 32 --	0 --	0.00
33- 33 --	0 --	0.00
34- 34 --	1 --	5.26
35- 35 --	1 --	5.26
36- 36 --	0 --	0.00
37- 37 --	0 --	0.00
38- 38 --	0 --	0.00
39- 39 --	0 --	0.00
40- 40 --	0 --	0.00
41- 41 --	0 --	0.00
42- 42 --	0 --	0.00
43- 43 --	0 --	0.00
44- 44 --	0 --	0.00
45- 45 --	0 --	0.00
46- 46 --	1 --	5.26
47- 47 --	0 --	0.00
48- 48 --	1 --	5.26
49- 49 --	1 --	5.26
50- 50 --	2 --	10.53
51- 51 --	3 --	15.79
52- 52 --	3 --	15.79
53- 53 --	1 --	5.26
54- 54 --	1 --	5.26
55- 55 --	1 --	5.26
56- 56 --	0 --	0.00
57- 57 --	0 --	0.00
58- 58 --	0 --	0.00
59- 59 --	0 --	0.00
60- 60 --	0 --	0.00
61- 61 --	0 --	0.00
62- 62 --	0 --	0.00
63- 63 --	0 --	0.00
64- 64 --	0 --	0.00
65- 65 --	0 --	0.00
66- 66 --	0 --	0.00
67- 67 --	1 --	5.26
68- 68 --	0 --	0.00
69- 69 --	1 --	5.26

TOTAL -- 19

SIZE COMPOSITION.... YELLOWFIN

LENGTH	FREQUENCY	PERCENT
31- 32 --	1 --	5.26
33- 34 --	1 --	5.26
35- 36 --	1 --	5.26
37- 38 --	0 --	0.00
39- 40 --	0 --	0.00
41- 42 --	0 --	0.00
43- 44 --	0 --	0.00
45- 46 --	1 --	5.26
47- 48 --	1 --	5.26
49- 50 --	3 --	15.79
51- 52 --	6 --	31.58
53- 54 --	2 --	10.53
55- 56 --	1 --	5.26
57- 58 --	0 --	0.00
59- 60 --	0 --	0.00
61- 62 --	0 --	0.00
63- 64 --	0 --	0.00
65- 66 --	0 --	0.00
67- 68 --	1 --	5.26
69- 70 --	1 --	5.26

TOTAL -- 19

PUBLICATIONS OF THE
INDO-PACIFIC TUNA DEVELOPMENT AND MANAGEMENT PROGRAMME

WORKING PAPERS

IPTP/82/WP/1
SCS/80/WP/90
SKILLMAN. R.A.Tuna fishery statistics for the Indian Ocean and the Indo-Pacific. Colombo, Indo-Pacific Tuna Development and Management Programme. June, 1982. 86p.

IPTP/82/WP/2
SCS/82/WP/111
DE JESUS. A.S. Tuna fishing gears of the Philippines. Colombo, Indo-Pacific Tuna Development and Management Programme. June, 1982 47p.

IPTP/82/WP/3
SCS/82/WP/112
WHITE. T.F. and M. YESAKI The status of tuna in Indonesia and the Philippines. Colombo, Indo-Pacific Tuna Development and Management Programme. September, 1982 62p.

IPTP/82/WP/4
SCS/82/WP/113
YESAKI. M. Illustrated key to small and or immature species of tuna and bonitos of the Southeast Asian region. Colombo, Indo-Pacific Tuna Development and Management Programme. October, 1982 16p.

IPTP/82/WP/5
SCS/82/WP/114
WHITE. T. F. and M. YESAKI. Tuna fisheries in the Philippines. Colombo, Indo-Pacific Tuna Develpment and Management Programme. December, 1982.

IPTP/83/WP/6
SCS/83/WP/6
YESAKI. M. The Pelagic Fisheries of the Philippines. Colombo, Indo-Pacific Tuna Development and Management Programme. March, 1983.

IPTP/82/WP/7
SCS/82/WP/119
YESAKI. M. Observations on the biology of yellowfin (Thunnus albacares) and skipjack (Katsuwonus pelamis) tunas in the Philippine waters. Indo-Pacific Tuna Development and Management Programme. July, 1983.

IPTP/83/WP/8
WHITE T.F. and M. YESAKI. The Balinese Tuna Fishery. Indo- Pacific Tuna Development and Management Programme. October, 1983. 15p.

IPTP/83/WP/9
WHITE T.F. and J.C.B. UKTOLSEJA. The West Java Tuna Fishery. Indo-Pacific Tuna Development and Management Programme. 1983. 21p.

IPTP/84/WP/10
JOSEPH B.D.L. Review of Tuna Fishery in Sri Lanka. Indo-Pacific Tuna Development and Management Programme. July, 1984. 35p.

IPTP/84/WP/11
SAKURAI T. Major Findings from the Indo-Pacific Historical Tuna Fisheries Data Summary. Indo-Pacific Tuna Development and Management Programme. September, 1984. 11p.

IPTP/85/WP/12
YONEMORI T., UKTOLSEJA J. and MERTA G.S. Tuna Tagging in Eastern Indonesian Waters. Indo-Pacific Tuna Development and Management Programme. February, 1985. 38p.