
* *
* Programa VIT *
* *

para análisis de pesquerías

Jordi Lleonart

Institut de Ciències del Mar (CSIC)
Passeig Nacional s/n
08003 Barcelona

Barcelona, junio de 1989

INDICE

	Introducción	1
	Agradecimientos	2
	Correspondencia	2
1	Manual del usuario	3
	1.1 Fichero de datos	3
	1.2 Ejecución del programa	4
	1.2.1 Lectura del fichero de datos	4
	1.2.2 Elección del modelo de VPA	4
	1.2.3 Parámetros	5
	1.2.4 Proporciones	7
	1.2.5 Menús	8
	1.2.5.1 Menú general	8
	1.2.5.2 Modificación de parámetros ..	9
	1.2.5.3 Rendimiento por recluta	9
	1.2.5.4 Análisis de sensibilidad	10
	1.2.5.5 Análisis de transición	11
	1.2.5.6 Paso de tallas a edades	13
	1.3 Fichero de salida de resultados	13
	1.3.1 Encabezamiento	13
	1.3.2 VPA	15
	1.3.3 Rendimiento por recluta	17
	1.3.4 Análisis de sensibilidad	18
	1.3.5 Análisis de transición	19
2	Especificaciones técnicas	19
	2.1 Programa principal y subrutinas	19
	2.2 Ficheros	19
	2.3 Nomenclatura	20
3	Organización del programa y metodología	22
4	Listados del programa principal y las subrutinas ..	34

Introducción

VIT es un programa destinado al análisis de pesquerías de las cuales se tiene información limitada. Ha sido diseñado con el objetivo de estudiar los datos generados por el proyecto "La pesca en Cataluña y el País Valenciano: descripción global y bases para el estudio de su seguimiento", financiado por la Dirección General XIV de la Comisión de las Comunidades Europeas entre 1986 y 1990, y ejecutado por el Instituto de Ciencias del Mar de Barcelona (CSIC). Por lo tanto está especialmente diseñado para resolver los problemas surgidos en este proyecto, básicamente, análisis a partir de frecuencias de tallas e interacción de artes.

El programa permite realizar diversos análisis cuando se dispone de frecuencias de tallas o edades (hasta 100 clases) de uno o varios (hasta 4) artes distintos. Se considera que la captura de cada uno de los artes está representada por una única frecuencia de tallas o edades. El programa requiere la entrada de un cierto número de parámetros: 3 correspondientes al modelo de crecimiento de von Bertalanffy, 2 correspondientes a la relación talla peso, la mortalidad natural (única para todas las clases y edades), la mortalidad por pesca terminal, y de 1 a 4 (según el número de artes considerados) correspondientes a los factores de proporción de la captura según arte.

Un menú de opciones generales permite acceder a distintos análisis: VPA (Análisis de Poblaciones Virtuales), Y/R (Rendimiento por recluta), Análisis de sensibilidad a los parámetros y Análisis de transición (posible solamente cuando los datos están organizados en edades). Permite asimismo el cambio de parámetros y la transformación de datos en tallas a datos en edades.

El VPA, el análisis de rendimiento por recluta y el análisis de sensibilidad a los parámetros se realizan bajo la hipótesis de equilibrio. En el análisis de transición no.

Este documento tiene la función de dar al usuario del VIT toda la información que se precise para trabajar e interpretar correctamente los resultados.

Agradecimientos

Este programa no habría podido ser confeccionado sin ayuda. Debo agradecer, en primer lugar, el asesoramiento que, en materia de cálculo, me han proporcionado Jordi Salat y Emili García.

Asimismo han resultado muy fructíferas para el diseño de este programa las discusiones con José Antonio Pereiro, Pilar Pallarés (ambos del Instituto Español de Oceanografía), Ana Gordo y Begoña Campos.

Por otro lado estoy también en deuda con Lluís del Cerro, Montserrat Demestre, Manuel Marhuenda, Paloma Martín, Balbina Moli, Charo Obarti, Juan Pablo Pertierra, Laura Recasens, Pere Rubiés y Pilar Sánchez, todos ellos miembros del equipo de trabajo del proyecto "La pesca en Cataluña y el País Valenciano, descripción global y planteamiento de bases para su seguimiento", inmediatos usuarios del proyecto han contribuido de forma definitiva a su perfeccionamiento. Asimismo todos los usuarios que han realizado pruebas han contribuido a su mejora, especialmente Luis Eduardo Calderón Aguilar, Khrisi Mitilineou y Montserrat Ramón.

Correspondencia

Cualquier consulta, sugerencia de mejora, detección de errores, etc. pueden dirigirla a

Jordi Lleonart

Institut de Ciències del Mar
Passeig Nacional s/n
08003 Barcelona.

teléfono: (93) 3106416 o 3106450

telex: 59367 INPBE

fax: 3199842

1 Manual del usuario

En este capítulo se pretende dar una guía de utilización al usuario con el fin de que pueda ejecutar el VIT e interpretar correctamente los menús y los resultados.

1.1 Fichero de datos

El fichero de datos, cuyo nombre elige el usuario, debe estar en formato ASCII con los siguientes registros:

Registro (fila)	Contenido
1	Comentario (a elección del usuario)
2	<u>n m iet clas1 xincr</u> (en formato libre) n = Número de clases de edad o de talla (XXXXXXXXXX) m = Número de artes (Max 4) iet = 1 si las clases son edades, = 2 si las clases son tallas clas1 = Valor del límite inferior de la primera clase. i = 3, n+2 <u>x(i,j)</u> Frecuencia de la clase <u>i</u> para cada arte (en formato libre)

Nota importante: Se debe tener en cuenta que la suma de las frecuencias de los artes no debe dar 0 para ninguna clase. En caso contrario se produce un error. Si existe una clase con frecuencias 0 para todos los artes se recomienda substituir estos ceros por 1 o cualquier otro número pequeño.

EJEMPLO

Dades de prova inventades

17	4	2	1	0.5	
50		0	0	0	
100		0	0	0	
200	10	0	0	0	
150	20	0	0	0	
125	30	5	0	0	
100	60	20	0	0	
50	125	40	1	1	
20	130	50	6	6	
10	130	30	10	10	
6	100	10	12	12	
1	80	4	16	16	
0	30	1	19	19	
1	12	0	23	23	
0	3	0	17	17	
0	1	0	11	11	
0	0	0	4	4	
0	0	0	1	1	

1.2 Ejecución del programa

Antes de proceder a la ejecución del programa debe existir ya el fichero con los datos.

Una ejecución del VIT trabaja con un fichero de datos y genera un único fichero de resultados. El usuario que desee hacer varias pasadas del programa con distintos datos deberá ejecutarlo al menos tantas veces como ficheros de datos tenga.

El programa está organizado por menús, de forma que se especifica en cada caso que es lo que se requiere de forma clara.

La ejecución del programa tiene dos partes:

- 1a parte: Proceso inicial. Lectura del nombre del fichero de datos, elección de modelo de VPA y lectura del conjunto de los parámetros. Se debe necesariamente dar valor a todos los parámetros que se solicitan. Solamente pasa una vez por esta parte. Una vez finalizada se pasa a la:
- 2a parte: Es el núcleo del programa y gira alrededor de un menú de opciones (cambiar parámetros, realizar diversos análisis y generar datos según edades). La realización de cualquier análisis provoca la adición al fichero de salida XXXXXXXXXX de un listado de resultados. El cambio de parámetros no genera ninguna salida. La opción de datos por edades genera un fichero nuevo. Este menú contiene la opción de finalización de la tarea.

1.2.1. Lectura del fichero de datos

Inmediatamente después de invocar el programa aparece el mensaje:

entra el nom del fitxer de dades

Se le dá dicho nombre y se procede al siguiente paso

1.2.2 Elección del modelo de VPA

El programa prevé la posibilidad de utilizar dos métodos para la reconstrucción de la población y las correspondientes mortalidades:

- 1 El VPA clásico, utilizando la ecuación de captura cuyas F (mortalidades por pesca) son calculadas por el método de solución aproximada de la secante (subrutinas VPA1, SECANT y FUNCT), y
- 2 El método del análisis de cohortes desarrollado por Pope, según el cual no es necesario proceder a métodos iterativos para reconstruir la población.

El menú que aparece es:

```

*****
*
* 1   fa VPA estàndard
*
* 2   fa anàlisi de cohorts de Pope
*
*****
*
*   Entra la teva opció
*
*****

```

Una vez elegida la opción esta será usada a lo largo de todo un RUN y no podrá ser modificada

1.2.3 Parámetros

Los parámetros de crecimiento, talla-peso, mortalidad natural, mortalidad por pesca final, proporciones, y proporción de maduros, se piden en la primera fase de la ejecución del programa y pueden ser modificados via menú. Excepto la proporción de maduros por clase, estos parámetros son fundamentales para los cálculos. La proporción de maduros solamente sirve para el calculo de la biomasa de reproductores en el rendimiento por recluta. Por supuesto todos los parámetros deberán haber sido calculados o estimados por el usuario previamente a la ejecución del VIT.

La entrada y/o modificación de parámetros no tiene problema, excepto los de proporciones que, por su complejidad, se le dedica específicamente el capítulo siguiente.

Los parámetros son entrados por primera vez en la primera parte de la ejecución y podrán ser modificados en la segunda. En este último caso se presenta el valor que tienen y se pregunta si el usuario desea cambiarlo.

Uno de los parámetros, el denominado a de la relación talla peso, es especialmente importante ya que sus unidades determinan las unidades (unidad de peso) en que se realizarán todos los cálculos de peso del programa. Por supuesto esto es cierto en todos los parámetros con dimensiones (L_{∞} , $t(0)$, M , etc.) pero es habitual no cambiar las unidades de longitud o tiempo, y sí lo es hacerlo con las de peso.

La secuencia de entrada es la siguiente:

1) para el crecimiento.

Solamente se usa el modelo de von Bertalanffy:

$$talla = L_{\infty} (1 - \exp(-K(edad-t(0))))$$

El mensaje que aparece es

entra els paràmetres de creixement

[blacked out] [blacked out]
 $L(w)$ K $t(0)$

El programa hace una comprobación de que L_{∞} no sea inferior a la talla máxima presentada en los datos. En caso que esto se produzca aparece el mensaje

L(inf) massa petit. No pot ser inferior a <talla(n)>

y vuelve a solicitar los datos. Tengase en cuenta que las unidades de longitud de L_{∞} , y de tiempo de K y de $t(0)$ deben mantenerse para cualquier otro parámetro.

2) para la relación talla peso

Se usa únicamente el modelo:

$$\text{peso} = a \text{ talla}^b$$

El mensaje es:

entra els parmetres de la relació talla-pes

a **b**

Como en el caso anterior a tiene dimensiones de masa y se deben mantener para todas las masas que se entren. b no tiene dimensiones.

3) La mortalida natural. Aparece el mensaje:

entra la mortalitat natural

Se debe incluir según las unidades de tiempo correspondientes. El programa toma la misma M para todas las clases.

4) Mortalidad por pesca terminal global. Aparece el mensaje

entra la mortalitat per pesca terminal

Se debe entrar la **terminal** (de la última clase considerada) global para todos los artes incluidos. Tiene unidades de tiempo.

5) Proporciones

Por su complejidad se explican en el próximo apartado.

6) **proporción de maduros**

En primer lugar aparece el mensaje

entra les proporcions de madurs'
quan ja tot siguin 1 entra -1'

Es decir, cuando, a partir de la clase sean todo 1, se debe entrar la cantidad -1. Inmediatamente salen n mensajes, cada uno de los cuales requiere su respuesta (que debe ser un número comprendido entre 0 = todos inmaduros y 1 = todos maduros). Si en el

- fichero de datos las clases están como edades aparece

```
classe <i> edat <edat(i)>
```

si son tallas

```
classe <i> talla <talla(i)>
```

Estos números no tienen unidades.

1.2.4 Proporciones

Las proporciones constituyen un apartado fundamental, y dado que su exposición es algo compleja, se le dedican este apartado.

La proporción para un arte es el número por el que se tienen que multiplicar todas sus frecuencias por clase con el fin que represente la frecuencia por clase según el esfuerzo que se le ha dedicado por la unidad de tiempo considerada (generalmente un año).

Supongamos también que las frecuencias entradas en el fichero de datos representan la captura media de una calada de cada arte. No obstante sabemos que por cada calada de arte 1 se producen 3 de arte 2 (suponemos que hay dos artes), y que en la unidad de tiempo considerada se han realizado 1200 caladas del arte 1. en este caso los factores de proporción pedidos serán: para el arte 1: 1200, y para el arte 2: 3600 (también puede trabajar en otras unidades, y quizás en este ejemplo estarían recomendadas, como 12 y 36 o 1.2 y 3.6 etc.).

El ejemplo propuesto es una de las opciones que el programa permite (proporciones respecto a los números), donde el usuario entra directa y explícitamente las proporciones por las que desea multiplicar sus frecuencias con el fin de obtener unas frecuencias que representen las reales en la unidad de tiempo determinada.

No obstante esto no siempre es posible o recomendable. La otra opción (denominada proporciones por pesos) permite calcular indirectamente las proporciones de otra forma.

Siguiendo el ejemplo anterior, supongamos que conocemos la captura total real de todos los artes en la unidad de tiempo considerada, y que conocemos, también, los porcentajes de esta captura que ha aportado cada uno de los artes. Entrando esta información con la ~~opción pesos~~, el programa calcula las proporciones que se deben aplicar.

En el programa aparece el siguiente mensaje

TRANSFORMACIO A LA CAPTURA TOTAL

```
0 per transformar segons nombres d'exemplars
1 per transformar segons pesos
```

Si la opción elejida ha sido 0 aparece

entra proporcions dels <m> arts

Estos deben ser números superiores a 0. Si la elección ha sido 1:

entra pes total

Se debe entrar el peso total de la captura en las unidades determinadas por el parámetro a de la relación talla-peso. Inmediatamente aparece:

entra percentatges dels <m> arts

Los porcentajes deben, por supuesto, sumar 100. En caso contrario se repite el mensaje.

1.2.5 Menús

VIT presenta varios menús, uno para las opciones principales y algunos otros para otras opciones.

1.2.5.1 Menú general

El menú general es el que permite dar las órdenes para realizar los análisis (y de ahí generar listados), cambiar los parámetros, generar un fichero de datos por edad y terminar. Se debe añadir que éste menú es algo distinto cuando se trabaja con tallas (donde falta la opción de análisis de la transición) y con edades (donde, obviamente, falta la opción de paso de tallas a edades).

El menú general (que no aparece nunca en esta forma) es:

```
*****
*
*          OPCIONS POSSIBLES          *
*
*****
*
*      1          Canvi de paràmetres  *
*
*      2          Execució de la VPA   *
*
*      3          Execució del YPR     *
*
*      4          Anàlisi de la transició *
*
*      5          Anàlisi de sensibilitat *
*
*      6          Passa de talles a edats *
*
*      7          Acaba                 *
*
*****
```

De todas las opciones que permite la de terminar y la de VPA son las únicas que no presentan submenú. La primera por razones triviales, y la segunda porque, en realidad (consúltese el

capítulo 3), la opción de VPA lo único que hace es grabar en el fichero de resultados el listado de salida del VPA.

Cuando se trabaja en tallas no se presenta ni se acepta la opción 4. Cuando las clases son edades ni se presenta ni se acepta la opción 6. Existe una opción 8 (que no se presenta en el menú) cuya función es cambiar el valor del parámetro de sensibilidad para la convergencia del VPA clásico.

1.2.5.2 Modificación de parámetros

Este menú, como indica su nombre, permite modificar los parámetros con los que trabaja el programa. En este caso siempre presenta los valores actuales de los parámetros y se pregunta si realmente se desean cambiar. Permite por lo tanto comprobar cuales son los valores actuales de los parámetros.

La presentación es:

```
*****
*
*          MODIFICACIO DE PARAMETRES          *
*
*****
*
*      1      Creixement                       *
*
*      2      Relació talla-pes                *
*
*      3      Mortalitat natural                *
*
*      4      F terminal                        *
*
*      5      Factors de proporció              *
*
*      6      Proporció de madurs               *
*
*      7      Torna al Menú principal          *
*
*****
```

1.2.5.3 Rendimiento por recluta

En la elección de este menú se imprimen (según dos posibles opciones) un listado de factor, rendimiento (en peso) por recluta total, biomasa, biomasa de reproductores, y rendimiento por recluta según arte.

En esta salida aparece también el listado de las proporciones de maduros (es el único lugar del programa donde se utilizan) y la derivada en el origen de la curva de rendimiento total (no por arte) por recluta.

El factor es, en este caso, el número por el que se multiplica cada una de las mortalidades por pesca por clase (global y por arte). Está comprendido entre 0 y 2 en intervalos de 0.01 unidades (hay por lo tanto 200 puntos, entre los que el factor 1 representa, el caso real que se estudia).

Las dos subopciones que permite son:

- 1 Solamente se imprimen los resultados correspondientes a los factores 0, 1, 2, los correspondientes a los máximos total y por arte y el F(0.1). Dado que los primeros son fáciles de localizar mientras que el F(0.1) no, este aparece separado del anterior el posterior por una línea en blanco.
- 2 Se imprimen las 200 líneas correspondientes a los factores entre 0 y 2. Su objetivo es poder dibujar la gráfica con una hoja de cálculo.

Las subrutinas de cálculo de rendimiento por recluta son usadas por otras partes del programa (particularmente los análisis de sensibilidad y de transición) y proporcionan, en consecuencia unos listados similares.

El menú presentado es:

```
*****
*
*          OPCIONS DEL RENDIMENT PER RECLUTA          *
*
*****
*      0      Altres opcions                          *
*      1      Cas present i màxims i mínims          *
*
*      2      200 punts (N punts)                   *
*
*****
```

1.2.5.4 Análisis de sensibilidad

El análisis de sensibilidad permite conocer los efectos que tienen los errores de las estimaciones de los parámetros sobre los resultados del análisis de la población.

Cuando se invoca esta opción aparece un menú que informa al usuario de los códigos numéricos de cada uno de los hasta 11 parámetros que puede incluir en el análisis de sensibilidad. También se solicitan los siguientes datos:

- 1 Cantidad de parámetros que se quieren incluir en el análisis. Estos pueden ser de 0 a 11 (si hay cuatro artes) o hasta 8 si hay uno solo.
- 2 Variación relativa de los parámetros que serán analizados.
- 3 Números identificadores de los parámetros que serán analizados.

El funcionamiento del programa se basa en el análisis de equilibrio (VPA + rendimiento por recluta, del que solamente se presenta este último), de todas las combinaciones de parámetros posibles. Por ejemplo, si queremos conocer la sensibilidad del análisis a los parámetros de crecimiento L₀, K y a la mortalidad

natural M , cuando varían en un 10% de su valor realizará 8 análisis con los siguientes valores (suponiendo valores iniciales de $L_{\infty}=10$, $K=0.3$ y $M=0.2$):

	L_{∞}	K	M
1er análisis	9.9	0.27	0.18
2º análisis	9.9	0.27	0.22
3er análisis	9.9	0.33	0.18
4º análisis	9.9	0.33	0.22
5º análisis	10.1	0.27	0.18
6º análisis	10.1	0.27	0.22
7º análisis	10.1	0.33	0.18
8º análisis	10.1	0.33	0.22

Previamente se realiza un análisis con los valores iniciales con fines comparativos. El programa informa del número de análisis que se han pedido con el fin de rectificar si fuere necesario. Si k es el número de parámetros que se quiere analizar, el número de análisis necesario será de $2^{**}k$, lo cual da la cantidad de 2048 en el caso de que se quieran analizar el máximo de 11 parámetros.

El menú es:

```

*****
*
*
*           ANALISI DE SENSIBILITAT
*
*****
*
*      1      2      3      4      5      6      7
*  L(inf)  K      to      a      b      M      F(term)
*
*      8 a 11  paràmetres de proporció
*
*****
*
*  ENTRA:      quants paràmetres vols modificar?
*              factor (expressat en tant per 1)
*              números identificadors dels paràmetres
*
*****

```

1.2.5.5 Análisis de transición

Esta es la única opción en la que se trabaja fuera del equilibrio. Su objetivo es analizar los efectos a corto plazo sobre la pesquería de cambios en la estrategia pesquera. El programa permite ver los pasos intermedios por los que pasa la pesquería desde una situación inicial de equilibrio hasta otra final, también de equilibrio, después de realizar algún cambio en la forma de pescar. Los resultados se presentan en forma de rendimiento por recluta.

Dos criterios independientes se requieren para realizar este análisis:

A) Tipo de cambio o cambios que producimos. Estos pueden ser de tres tipos:

- 1 Alteración de los factores de proporción de los artes (capítulo 1.2.4).
- 2 Alteración de las mortalidades por pesca por factores (según los artes) y
- 3 Cambio de las mortalidades por pesca

El primero y segundo tipo son muy parecidos entre sí, pero no exactamente iguales. El primero requiere la realización de un nuevo VPA con los nuevos factores para usar las nuevas mortalidades por pesca fuera del equilibrio. El segundo se limita a multiplicar las mortalidades por pesca por los factores elegidos. En el tercero simplemente se imponen nuevas mortalidades por pesca. Los dos primeros pueden ser usados cuando deseamos simular cambios en el esfuerzo aplicado por cada uno de los artes, el último es útil si queremos simular una modificación en las selectividades de los artes.

B) La modificación se realiza el primer año de simulación y se mantiene el resto del tiempo (opción B1) o bien, cada uno de los años de simulación presenta una alteración respecto del anterior (opción B2). Esto es útil, por ejemplo, para introducir un cambio repentino o paulatino (definido en A) y simular lo que ocurre en cada caso. Si las modificaciones se introducen año a año (caso B2), dado que se requerirán tantos vectores de factores como años a simular para los casos A1 y A2 y tantas matrices de mortalidades por pesca (cada matriz contiene tantas F's como clases y artes) como años se deseen simular, para comodidad del usuario, estas series de números deberán haber sido grabados previamente en un fichero (de nombre elegido por el usuario) que será leído por el programa previa petición de dicho nombre.

El menú que aparece es:

```
*****
*
*                OPCIONS DE LA TRANSICIO                *
*
*****
*
*   Mode A  1  canvi dels factors de proporció          *
*            2  alteració de F per un factor            *
*            3  canvi de mortalitats per pesca          *
*
*   Mode B  1  el canvi es produeix de cop              *
*            2  el canvi es produeix any per any        *
*
*****
*
*   Solament les opcions Mode A=1 i 2 i Mode B = 1      *
*   permeten entrada per teclat. Qualsevol altra com-  *
*   binació necessita un fitxer de lectura.            *
```

```

*
*****
*
*   Entra: anys per simular, Mode A, i Mode B
*
*****

```

1.2.5.6 Paso de tallas a edades

Construye a partir de los datos organizados por clases de edad y de los parámetros de crecimiento, un fichero de datos organizado por clases de talla, con las siguientes opciones (la tercera aún no está implementada).

- 1 el fa de les dades originals
- 2 el fa de les dades afectades per les proporcions
- 3 el fa de les dades ajustades per la VPA

El nombre del fichero de salida es impuesto por el usuario
entra el nom del fitxer de sortida

1.3 Fichero de salida de resultados

1.3.1 Encabezamiento

El encabezamiento aparece cada vez que se invoca la opción VPA (debido a que algunos de sus resultados están calculados gracias a que previamente se ha realizado tal análisis), y consta de varias partes:

En primer lugar aparece el título de los datos, los parámetros de crecimiento, los de la relación talla-peso, la mortalidad natural y las proporciones.

Dades de prova inventades

```

L(inf) =      10.00   K =          .30   t(0) =          .00
a =      .100E+01   b =          3.00
M =          .30

```

```

factors de proporció      1.0000   1.0000   1.0000   1.0000

```

A continuación, bajo el epígrafe Dades generals, aparecen los límites inferiores y las medias por clase, de edad, talla y peso. Es fácil observar en este listado de como estaban organizados los datos (si por tallas o por edades), por el tipo

de números que aparecen (en este caso las tallas inferiores). Obsérvese que las columnas de edad, talla y pesos inferiores presentan una fila más, correspondiente a las correspondientes superiores de la última clase.

Es muy importante notar que las medias son calculadas después de reconstruir la población con el VPA, y por lo tanto se calculan considerando el número de individuos que están presentes en cada clase. De esta forma, por ejemplo, la clase 4, que es la comprendida entre la talla 2.5 y la talla 3, tiene una talla media inferior a 2.75, que es la que tendría si la distribución de los individuos en la clase fuera rectangular; pero dado que, después del VPA, conocemos ya números de individuos y mortalidades no tiene sentido considerar este tipo de distribuciones sencillas. Dado que la distribución de los números dentro de una clase sigue una exponencial negativa, es lógico y natural que las media de edades, tallas o pesos sean inferiores a los puntos intermedios de la clase

Dades generals

clas	ed inf	ed mitj	tall inf	tall mitj	pes inf	pes
1	.3512	.4453	1.0000	1.2493	1.000	2.
2	.5417	.6411	1.5000	1.7485	3.375	5.
3	.7438	.8487	2.0000	2.2464	8.000	11
4	.9589	1.0710	2.5000	2.7466	15.63	20
5	1.1889	1.3090	3.0000	3.2462	27.00	34
6	1.4359	1.5649	3.5000	3.7449	42.88	52
7	1.7028	1.8412	4.0000	4.2423	64.00	76
8	1.9928	2.1428	4.5000	4.7400	91.13	10
9	2.3105	2.4742	5.0000	5.2374	125.0	14
10	2.6617	2.8434	5.5000	5.7363	166.4	18
11	3.0543	3.2560	6.0000	6.2321	216.0	24
12	3.4994	3.7341	6.5000	6.7347	274.6	30
13	4.0132	4.2841	7.0000	7.2304	343.0	37
14	4.6210	4.9466	7.5000	7.7281	421.9	46
15	5.3648	5.7611	8.0000	8.2185	512.0	55
16	6.3237	6.8688	8.5000	8.7183	614.1	66
17	7.6753	8.5718	9.0000	9.2222	729.0	78
	9.9858		9.5000		857.4	

A continuación se presentan las capturas en números (ya transformados por las proporciones). En este cuadro no hay nada más que los datos de entrada (alterados por las proporciones) y su suma por filas y por columnas. También se presentan las tallas y edades medias de las capturas total y por arte.

Captures en nombres

clas	Captura total	Captures per art			
1	50.0000	50.0000	.0000	.0000	.0000
2	100.0000	100.0000	.0000	.0000	.0000
3	210.0000	200.0000	10.0000	.0000	.0000
4	170.0000	150.0000	20.0000	.0000	.0000
5	160.0000	125.0000	30.0000	5.0000	.0000
6	180.0000	100.0000	60.0000	20.0000	.0000

7	216.0000	50.0000	125.0000	40.0000	1
8	206.0000	20.0000	130.0000	50.0000	6
9	180.0000	10.0000	130.0000	30.0000	10
10	128.0000	6.0000	100.0000	10.0000	12
11	101.0000	1.0000	80.0000	4.0000	16
12	50.0000	.0000	30.0000	1.0000	19
13	36.0000	1.0000	12.0000	.0000	23
14	20.0000	.0000	3.0000	.0000	17
15	12.0000	.0000	1.0000	.0000	11
16	4.0000	.0000	.0000	.0000	4
17	1.0000	.0000	.0000	.0000	1
Sum	1824.0000	813.0000	731.0000	160.0000	120
edat mitjana	1.8970	1.1330	2.3560	2.1128	3
talla mitjana	4.0915	2.8118	4.9511	4.6498	6

Se presentan luego las capturas por peso. El cuadro general es el mismo que el anterior pero transformando los números a pesos. En la última fila se presentan los porcentajes del peso total con los que contribuye cada arte. Obsérvese son precisamente estos números: el peso total de la captura y los mencionados porcentajes los que se fijan en la segunda opción de las proporciones (capítulo 1.2.4).

Captures en pesos

clas	Captura total	Captures per art			
1	101.3966	101.3966	.0000	.0000	
2	545.4646	545.4646	.0000	.0000	
3	2409.9470	2295.1880	114.7594	.0000	
4	3551.4680	3133.6480	417.8198	.0000	
5	5505.8610	4301.4540	1032.3490	172.0582	
6	9495.6750	5275.3750	3165.2250	1055.0750	
7	16548.1900	3830.5990	9576.4980	3064.4790	76
8	21999.4500	2135.8690	13883.1500	5339.6730	640
9	25918.0500	1439.8920	18718.5900	4319.6750	1439
10	24205.6900	1134.6420	18910.7000	1891.0690	2269
11	24486.3800	242.4394	19395.1500	969.7578	3879
12	15294.1600	.0000	9176.4960	305.8832	5811
13	13623.7300	378.4369	4541.2420	.0000	8704
14	9240.4490	.0000	1386.0670	.0000	7854
15	6667.1920	.0000	555.5994	.0000	6111
16	2652.7230	.0000	.0000	.0000	2652
17	784.9000	.0000	.0000	.0000	784
Sum	183030.7000	24814.4100	100873.7000	17117.6700	40225
Percentatges		13.5575	55.1130	9.3523	21.

1.3.2 VPA

Los resultados de este apartado son los de la reconstrucción de la población por medio de la versión elegida de VPA. Se presenta en dos cuadros. El primero y principal contiene, por

clase, el número de individuos al inicio de cada clase, el número medio de individuos en el mar, y las mortalidades total, por pesca total y por pesca por arte. Al final se presentan las sumas de números, las medias de mortalidades y la edad y la talla media de la población.

RESULTATS DE LA VPA

nombres i mortalitats

clas	Nombre	Nombre mitj	z	f(tot)	f per art			
1	3079.	566.	.3884	.0884	.0884	.0000	.0000	.0000
2	2860.	551.	.4816	.1816	.1816	.0000	.0000	.0000
3	2595.	518.	.7055	.4055	.3862	.0193	.0000	.0000
4	2229.	476.	.6573	.3573	.3153	.0420	.0000	.0000
5	1916.	436.	.6666	.3666	.2864	.0687	.0115	.0115
6	1625.	393.	.7584	.4584	.2547	.1528	.0509	.0509
7	1328.	337.	.9409	.6409	.1484	.3709	.1187	.1187
8	1011.	273.	1.0551	.7551	.0733	.4766	.1833	.1833
9	723.	208.	1.1639	.8639	.0480	.6239	.1440	.1440
10	480.	152.	1.1425	.8425	.0395	.6582	.0658	.0658
11	307.	104.	1.2683	.9683	.0096	.7670	.0384	.0384
12	174.	70.	1.0164	.7164	.0000	.4298	.0143	.0143
13	103.	46.	1.0813	.7813	.0217	.2604	.0000	.0000
14	54.	28.	1.0142	.7142	.0000	.1071	.0000	.0000
15	25.	15.	1.1047	.8047	.0000	.0671	.0000	.0000
16	9.	7.	.8784	.5784	.0000	.0000	.0000	.0000
17	3.	3.	.6000	.3000	.0000	.0000	.0000	.0000

Tot 18521. 4182.

mortalitats mitjanes .8758 .5758 .0482 .1736 .0218

població: edat mitjana 1.43 talla mitjana 3.25

El segundo cuadro contiene las dos primeras columnas del anterior transformadas a pesos.

pesos

clas	Pes	Pes mitj
1	3079.	1147.
2	9652.	3004.
3	20756.	5943.
4	34831.	9940.
5	51744.	15020.
6	69693.	20713.
7	84973.	25819.
8	92090.	29133.
9	90345.	30001.
10	79902.	28732.
11	66240.	25287.
12	47888.	21350.
13	35479.	17437.

14	22619.	12937.
15	12910.	8285.
16	5368.	4586.
17	1944.	2616.
Tot	729513.	261950.

1.3.3 Rendimiento por recluta

Se presenta, en primer lugar, la proporción de maduros por clase (es el único lugar donde se hace). A continuación el valor de la pendiente en el origen y finalmente el listado de resultados que depende de la opción elegida.

En cualquier caso el programa calcula 200 puntos entre los factores 0 y 2 (son los factores que multiplican las mortalidades por pesca) en incrementos de 0.01. Si la opción elegida es 1 solamente se presentan (ordenados según el valor del factor) los casos correspondientes a los máximos rendimientos total y por arte, los valores para los factores 0, 1 y 2 y el correspondiente al F(0.1), que es el punto para el cual la pendiente del rendimiento total es el 10% de la del origen. Este valor, para reconocerlo, se presenta separado del anterior y del posterior por sendas líneas en blanco.

Si la opción es la 2, simplemente se listan los 200 puntos considerados. En todos los casos se da el rendimiento por recluta (en peso) total y por arte, la biomasa y la biomasa de reproductores.

En el ejemplo se presenta solamente los resultados correspondientes a la primera opción.

ANALISI DE RENDIMENT PER RECLUTA

Proporcions de madurs

1	.00	2	.00	3	.00	4	.00	5	.00	6	.00	7	.00	8	.00	9
10	.25	11	.50	12	.75	13	1.00	14	1.00	15	1.00	16	1.00	17	1.00	

Pendent en l'origen = 483.198

Factor	Y/R	Biomassa	S.S.B	Y/R segons art		
.00	.00000	747.40	600.38	.00	.00	.00
.30	65.646	315.17	207.16	3.56	26.98	3.21
.32	67.297	292.36	187.46	3.84	28.28	3.43
.48	70.363	206.75	115.73	5.10	32.56	4.31
.71	66.984	132.41	58.236	6.62	34.39	5.13
1.00	59.437	85.066	26.532	8.06	32.76	5.56
1.17	55.138	68.796	17.327	8.72	30.95	5.61
2.00	39.764	32.902	2.8081	10.70	21.40	4.91

1.3.4 Análisis de sensibilidad

La salida del análisis de sensibilidad corresponde a los rendimientos por recluta que dan distintos parámetros. Se imprime el factor, en este caso igual a 0.1. Esto significa que cada uno de los parámetros incluidos en el análisis se presenta en dos casos, por ejemplo, si consideramos la K a la que se ha dado un valor original de 0.3, en un caso (indicado con -) valdrá 0.27 y en el otro (indicado con +) tendrá un valor de 0.33. El símbolo (•) significa que el parámetro tiene el valor original.

A continuación se presentan los resultados del rendimiento por recluta (sin indicar el factor g , que siempre es 1). Al principio de cada línea hay once campos correspondientes cada uno a los 11 posibles parámetros a analizar (separados por grupos para facilitar la lectura). La primera fila corresponde al rendimiento por recluta original (sin alterar ningún parámetro). A continuación aparecen todas las combinaciones posibles de los dos valores de los parámetros considerados.

Si se trabaja con todos los parámetros (11) se tendrían 2048 análisis.

ANALISI DE SENSIBILITAT

Factor	.10	paràmetres = 2 4 5 6						
parametres	Y/R	Biomassa	SSB	Y/R per art				
••• •• •••	59.44	85.07	26.53	8.06	32.76	5.56	13	
••• -- •• •••	32.32	52.28	14.54	4.95	17.74	3.11	6	
••• -- + •••	28.22	49.93	13.84	4.32	15.49	2.72	5	
••• -+ •• •••	89.26	138.56	47.20	10.72	49.21	8.07	21	
••• -+ + •••	77.94	132.14	44.88	9.35	42.97	7.04	18	
••• +- •• •••	39.50	63.90	17.78	6.05	21.68	3.80	7	
••• +- + •••	34.49	61.03	16.91	5.29	18.93	3.32	6	
••• ++ •• •••	109.10	169.35	57.69	13.10	60.14	9.86	26	
••• ++ + •••	95.26	161.50	54.85	11.43	52.51	8.61	22	
•+ •• •• •••	35.87	44.71	12.67	5.50	19.68	3.45	7	
•+ -- + •••	32.28	43.13	12.20	4.95	17.72	3.11	6	
•+ -+ •• •••	99.06	118.90	41.23	11.89	54.60	8.95	23	
•+ -+ + •••	89.16	114.59	39.69	10.70	49.15	8.06	21	
•+ +- •• •••	43.84	54.65	15.49	6.72	24.06	4.22	8	
•+ +- + •••	39.46	52.72	14.92	6.05	21.65	3.80	7	
•+ ++ •• •••	121.07	145.32	50.40	14.54	66.74	10.94	28	
•+ ++ + •••	108.98	140.05	48.51	13.08	60.07	9.85	25	

1.3.5 Análisis de transición

Esta es la única parte del VIT que realiza un análisis fuera del equilibrio. Solamente es posible con datos organizados en edades, por lo que el ejemplo que ponemos no se corresponde con los anteriores (que provenían de tallas). Se presentan las

salidas de rendimiento por recluta, en primer lugar (ini) en equilibrio en las condiciones iniciales, luego tantas filas como años se ha solicitado simular (fuera del equilibrio), y en último lugar (fin) la situación de equilibrio en las condiciones finales. Se considera reclutamiento constante.

Si el cambio de condiciones es repentino y se mantiene (Mode B = 1, véase capítulo 1.2.5.5) se alcanza el equilibrio en tantos años como edades estén consideradas.

En el ejemplo presente se ha hecho un cambio proporcional en las mortalidades por pesca (Mode A = 2) (respectivamente 0.1, 1, 1 y 5 por arte) y Mode B = 1.

ANALISI DE LA TRANSICIO

any	Y/R	biomassa	S.S.B.	Y/R segons arts			
ini	52.928	79.384	22.954	7.481	28.763	4.948	11
1	74.080	61.078	8.718	.777	21.786	4.386	47
2	55.339	63.398	3.678	.882	23.378	5.242	25
3	62.708	68.520	4.334	.904	26.400	5.816	29
4	65.734	69.667	4.951	.905	27.067	5.849	31
5	66.020	69.724	5.008	.905	27.078	5.849	32
6	66.021	69.724	5.008	.905	27.079	5.849	32
7	66.021	69.724	5.008	.905	27.079	5.849	32
8	66.021	69.724	5.008	.905	27.079	5.849	32
9	66.021	69.724	5.008	.905	27.079	5.849	32
10	66.021	69.724	5.008	.905	27.079	5.849	32
fin	66.021	69.724	5.008	.905	27.079	5.849	32

2 Especificaciones técnicas

El programa está escrito en lenguaje FORTRAN 77, y es ejecutable en ordenadores PC-compatibles, con o sin coprocesador matemático.

2.1 Programa principal y subrutinas

VIT es modular y consta de un programa principal y 50 subrutinas de estructura sencilla y objetivo único. La mayoría de estas subrutinas comparten entre ellas y con el programa principal un COMMON que contiene todas las variables importantes para los procesos y evita, por lo tanto, el exceso de parámetros en las llamadas.

2.2 Ficheros

La configuración mínima es de dos ficheros: uno que contiene los datos y cuyo nombre es elegido por el usuario, y uno de salida de resultados que se llama VIT.PRN que es machacado al ejecutar de nuevo el programa.

Otros dos ficheros pueden intervenir en el proceso: uno de salida de la opción de paso de tallas a edades, que contendrá datos por edades dispuestos para ser leídos y analizados por el programa VIT, y otro de lectura para ciertas opciones del análisis de transición (capítulo 1.3.5). En ambos casos el usuario decide el nombre del fichero.

2.3 Nomenclatura

El COMMON es el siguiente:

```
common p0,p1,p2,p3,p4,p5,p6,p7,p8,p9
common n,m,iet,clas1,xincr,x(100,4),xeps,ivc
common xlinf,xk,t0,a,b,xm,fterm,prop(4),fecun(100)
common talla(101),edat(101),c(100,4),ctot(100),w(101)
common dt(100),tmitj(100),wmitj(100),emitj(100)
common xn(100),fg(100,4),ftot(100),z(100)
common cfin(4),ffin(4)
```

Las variables son enteras o reales según las normas FORTRAN sin modificar con las sentencias REAL o INTEGER. En la siguiente lista se presentan las principales variables usadas, en la nomenclatura del programa (1) y en nomenclatura estándar para las explicaciones posteriores. Su significado es el siguiente (las variables o vectores marcados con * se leen como parámetros en el fichero de datos y no pueden ser modificados por el programa):

1 ---	2 ---	Significado -----
p0-p9	-	Diez variables sin contenido por si se necesitan en el futuro
* n	n	Número de clases de edad o de talla (Max 100)
* m	m	Número de artes (Max 4)
* iet	-	Parámetro de control: si los datos son edades = 1 si los datos son tallas = 2
* clas1	-	Valor del límite inferior de la primera clase.
* xincr	-	Incremento de clases (para edades debe ser 1)
* x(100,4)	-	Matriz de datos
* xeps	-	Precisión de cálculo de VPA. Originalmente vale 1.E-12, pero cuando hay problemas de convergencia es modificado automáticamente. Puede ser también modificado manualmente (ver subrutinas VPA, SECANT y MENU).

ivc	-	Variable de control. Si vale 1, realiza el VPA estándar, si vale 2, realiza el análisis de cohortes de Pope
xlinf	L_0	Parámetro de la ecuación de crecimiento de von Bertalanffy
xk	K	Parámetro de la ecuación de crecimiento de von Bertalanffy
t0	t_0	Parámetro de la ecuación de crecimiento de von Bertalanffy
a	a	Parámetro de la relación talla-peso $w=a \cdot l^b$. Su valor se determina las unidades de peso que se utilizarán a lo largo de todo el programa.
b	b	Parámetro de alometría de la relación talla-peso
xm	M	Mortalidad natural (constante para todas las clases)
fterm	F(n)	Mortalidad por pesca terminal (conjunta para todos los artes)
prop(4)	-	Factores de proporción de cada arte (véase capítulo <u>1.2.2 Proporciones</u>)
fecun(100)	pm(i)	Proporción de individuos maduros de cada clase (0 todos inmaduros, .3 el 30% de maduros, 1 todos maduros, etc.)
talla(101)	l(i)	Talla inferior de cada clase.
edat(101)	t(i)	Edad inferior de cada clase.
c(100,4)	C(i,j)	Matriz de frecuencias por arte y clase una vez han sido transformados los datos (x(100,4)) según las proporciones, y por lo tanto, datos a analizar.
ctot(100)	C(i)	Captura total (para todos los artes) según clase. Transformada ya según las proporciones.
w(101)	w(i)	Peso inferior de cada clase (en las unidades de <u>a</u>)
dt(100)	$\delta t(i)$	Incrementos de tiempo (edad) entre límites de clase. Para edades todos los valores son 1.
tmitj(100)	$\bar{l}(i)$	Talla media de cada clase
wmitj(100)	$\bar{w}(i)$	Peso medio de cada clase

emitj(100) $\bar{t}(i)$ Edad media de cada clase

xn(100) N(i) Número de individuos de la población en el límite inferior de cada clase

fg(100,4) F(i,j) Matriz de mortalidades por pesca según clase y arte

ftot(100) F(i) Mortalidad por pesca global (para todos los artes) según clase

z(100) Z(i) Mortalidad total según clase

cfin(4) CT(j) Captura total por arte (en número de individuos)

ffin(4) - Mortalidad por pesca global (para todas las clases) según arte.

3 Organización del programa y metodología

Dados unos datos, unos parámetros y un modelo de poblaciones virtuales,

A continuación se da la lista del programa principal y (por orden alfabético) de las subrutinas, con una pequeña descripción de su funcionamiento: se indica el Grupo a que pertenece (significa dentro de que tipo de análisis trabaja), parámetros de llamada (solamente se pone cuando existen), si no tiene common se indica, si tiene, no se indica; programa o subrutina que la llama y subrutinas a las que llama (si no lo hace no se indica).

VIT Programa principal

Llama a creix2, tp2, mnat2, fter2, pro2, fecu2, vpa, menu, param, imp1, imp2, ypr200, canvi, sensib, te

Lee el fichero de datos

Abre el fichero vit.prn para imprimir los resultados

Por medio de la llamada a menu llama las subrutinas principales de análisis

calcu2

Grupo vpa

Es llamada por **vpa**

Llama a **simp1**

Calcula las medias de talla, edad y peso para cada clase.

La edad media se calcula:

$$\bar{t}(i) = \frac{\int_{t(i)}^{t(i+1)} t \, dN}{\int_{t(i)}^{t(i+1)} dN} =$$

$$= \frac{\exp\{-Z(i)t(i)\} \{t(i) + 1/Z(i)\} - \exp\{-Z(i)t(i+1)\} \{t(i+1) + 1/Z(i)\}}{\exp\{-Z(i)t(i)\} - \exp\{-Z(i)t(i+1)\}}$$

La talla media se calcula:

$$\bar{l}(i) = \frac{\int_{t(i)}^{t(i+1)} l \, dN}{\int_{t(i)}^{t(i+1)} dN} = \frac{L_{\infty}}{\exp\{-Z(i)t(i)\} - \exp\{-Z(i)t(i+1)\}}$$

$$\left[\exp\{-Z(i)t(i)\} \frac{1 - Z(i) \exp[-K(t(i)-t_0)]}{Z(i) + K} - \exp\{-Z(i)t(i+1)\} \frac{1 - Z(i) \exp[-K(t(i+1)-t_0)]}{Z(i) + K} \right]$$

El peso medio se calcula:

$$\bar{w}(i) = \frac{\int_{t(i)}^{t(i+1)} w \, dN}{\int_{t(i)}^{t(i+1)} dN} = \int_{t(i)}^{t(i+1)} \exp(-Z(i) t \{1 - \exp[-K(t-t_0)]\})^b dt$$

dado que no se ha podido dar con una solución analítica se resuelve por el método numérico de Simpson (véase subrutina **simp1** y **functw**).

calcul

Grupo **vpa**

Es llamada por **vpa**

Calcula límites inferiores de edad, talla y peso

En el caso de que las clases de lectura sean tallas

$$t(i) = t_0 - \ln(1 - l(i)/L_\infty)/K$$

Si son edades

$$l(i) = L_\infty (1 - \exp(-K(t(i)-t_0)))$$

Calcula los pesos:

$$w(i) = a l(i)^b$$

Calcula los incrementos de tiempo:

$$\delta t(i) = t(i+1) - t(i)$$

Calcula también las capturas transformadas según las proporciones

canvi

Grupo **VIT** subrutina principal para el análisis de transición

Es llamada por **VIT**

Llama a **ypr**, **llemor**, **llepomor**, **llepro**, **yprn**, **vectn**, **vpa**

Realiza el análisis de la transición

creix

Grupo **param**

Es llamada por **param**

Llama a **creix1**, **creix2**

Subrutina de trabajo para los parámetros de crecimiento

creix1

Grupo **param**

Parámetros: (**iban**)

iban = 0 no lee nuevos parámetros

= 1 lee nuevos parámetros

Es llamada por **creix**

Llama a **sino**

Presenta los parámetros de crecimiento

creix2

Grupo param

Es llamada por VIT, creix

Lee los parámetros de crecimiento. En el caso que el parámetro \underline{L} leído sea inferior a $\underline{l}(n)$ da aviso de error y vuelve a pedirlo.

derypr

Grupo ypr

Parámetros: (n,q,xm,ftot,wmitj,dt,d)

q = factor del rendimiento por recluta

d = derivada

Es llamada por ypr200

Calcula la derivada del rendimiento por recluta respecto al factor q, que multiplica la mortalidad por pesca:

$$\frac{dY}{dq} = \sum_{i=1}^n \left[\frac{F(i) w(i)}{Z(i)} \left[\{N(i)-N(i+1)\} - \left[1 - q \frac{F(i)}{Z(i)} - q \sum_{j=1}^{i-1} \{F(j) \delta t(j)\} \right] + q F(i) \delta t(i) N(i+1) \right] \right]$$

distrib

Grupo te

Parámetros: (n,mi,ma,x,talled)

Es llamada por matrius

A partir de talled(100,50) genera la composición por edades de una clase de talla

fecu

Grupo param

Es llamada por param

Llama a fecu1, fecu2

Subrutina de trabajo para lectura de proporción de maduros

fecu1

Grupo param

Parámetros: (iban)

iban = 0 no lee nuevos parámetros

= 1 lee nuevos parámetros

Es llamada por fecu

Llama a sino

Presenta los valores actuales de proporción de maduros

fecu2

Grupo param

Es llamada por VIT, fecu

Lee las proporciones de maduros

fter

Grupo param

Es llamada por param

Llama a fter1, fter2

Subrutina de trabajo para la lectura de la F terminal

fter1

Grupo param

Parámetros: (iban)

iban = 0 no lee nuevos parámetros

= 1 lee nuevos parámetros

Es llamada por fter

Llama a sino

Presenta la F terminal actual

fter2

Grupo param

Es llamada por VIT, fter

Lee la F terminal

funct

Grupo vpa

Parámetros: (f, fun, xm, xn, c, t)

No tiene common

Es llamada por secant

Contiene la ecuación de captura del VPA

functw

Grupo vpa

Subrutina function

Parámetros: (i, edad)

No tiene common

Es llamada por simp1

Contiene la integral para cálculo de pesos medios

imp1

Grupo **VIT**: subrutina de impresión
Es llamada por **VIT**
Llama a **imp11**, **imp12** e **imp13**

Subrutina de trabajo de impresión

imp11

Grupo **VIT**: subrutina de impresión
Es llamada por **imp1**

Imprime la cabecera

imp12

Grupo **VIT**: subrutina de impresión
Es llamada por **imp1**

Imprime el cuadro de datos generales por números

imp13

Grupo **VIT**: subrutina de impresión
Es llamada por **imp1**

Imprime los datos generales por pesos

imp2

Grupo **VIT**: subrutina de impresión
Es llamada por **VIT**

Imprime resultados del VPA

llemor

Grupo **transición**
Es llamada por **canvi**

Lee de fichero una matriz de mortalidades por pesca

llepmor

Grupo **transición**
Parámetros: (mode)
mode=1 lee de consola
mode=2 lee de fichero

Es llamada por **canvi**

Lee de fichero o consola (según el parámetro mode) un vector de factores (tantos como artes) para multiplicar por las mortalidades por pesca

llepro

Grupo **transición**

Parámetros: (mode)

mode=1 lee de consola

mode=2 lee de fichero

Es llamada por **canvi**

Llama a **vpa, pro**

Lee de fichero o consola (según el parámetro mode) un vector de factores de proporción para las capturas según arte.

matrius

Grupo **te**

Parámetros: (n,m,mi,ma,x,talled)

No tiene **common**

Es llamada por **te**

Llama a **distrib**

A partir de la matriz **talled(100,50)**, construye la composición por edades a partir de la composición por tallas

menu

Grupo **VIT**

Parámetros: (men)

men = opción elegida

Es llamada por **VIT**

Presenta el menú de opciones (de 1 a 7) principales. Existe una opción 8, no presentada, que permite cambiar la sensibilidad para la convergencia del VPA.

meps

Grupo **vpa**

Es llamada por **menu**

Llama a **sino**

Permite cambiar la precisión para la convergencia del VPA

mnat

Grupo **param**

Es llamada por **param**

Llama a **mnat1** y **mnat2**

Subrutina de trabajo para la lectura de M

mnat1

Grupo param

Parámetros: (iban)

iban = 0 no lee nuevos parámetros
= 1 lee nuevos parámetros

Es llamada por mnat

Llama a sino

Presenta la mortalidad natural actual

mnat2

Grupo param

Es llamada por mnat

Lee la mortalidad natural

param

Grupo VIT

Es llamada por VIT

Llama a creix, tp, mnat, fter, pro, fecu

Presenta el menú de cambio de parámetros

pro

Grupo param

Es llamada por param

Llama a pro1, pro2

Subrutina de trabajo para la lectura de las proporciones

pro1

Grupo param

Parámetros: (iban)

iban = 0 no lee nuevos parámetros
= 1 lee nuevos parámetros

Es llamada por pro

Llama a sino

Presenta las proporciones actuales

pro2

Grupo param

Es llamada por pro

Llama a vpa

Lee las proporciones

secant

Grupo vpa
Parámetros: (c,x,xm,xn,t,xeps)
No tiene common
Es llamada por vpa1
Llama a funct1

Subrutina de cálculo aproximado por el método de la secante

sensib

Grupo VIT. Subrutina principal para análisis de sensibilidad
Es llamada por VIT
Llama a sino, ypr, vpa

Realiza el análisis de sensibilidad

simp1

Grupo vpa
Parámetros: (i)
No tiene common
Es llamada por calcu2
Llama a functw

Calcula una integral numéricamente por el método de Simpson

sino

Grupo: subrutina de caracter general
Parámetros: (iban)
 iban=0 si ha leído S, s, Y o y
 iban=1 si ha leído N o n
Es llamada por sensib, meps, creix1, fecul, fter1, mnat1,
 pro1, tpl

Subrutina para interpretar la lectura de "si" (S o s) o "yes" (Y o y) y "no" (N o n). Otro caracter no es aceptado.

te

Grupo VIT: Subrutina principal de paso de tallas a edades
Parámetros: (titol)
Es llamada por VIT
Llama a vpedat, matrius

Pasa de tallas a edades. Abre el fichero (de nombre decidido por el usuario) conteniendo los datos organizados por clases de edad y legibles directamente por VIT

tp

Grupo **param**
Es llamada por **param**
Llama a **tp1**, **tp2**

Subrutina de trabajo para la lectura de los parámetros de la relación talla-peso

tp1

Grupo **param**
Parámetros: (**iban**)
 iban = 0 no lee nuevos parámetros
 = 1 lee nuevos parámetros
Es llamada por **tp**
Llama a **sino**

Presenta los parámetros actuales de la relación talla-peso

tp2

Grupo **param**
Es llamada por **VIT**, **tp**

Lee los parámetros de la relación talla-peso

vectn

Grupo **transición**
Parámetros: (**xxn**)
 xxn es el vector de número de individuos por edad
 fuera del equilibrio
No tiene **common**
Es llamada por **canvi**

Construye un vector de números de individuos de una cohorte fuera del equilibrio

vpa

Grupo **VIT**, subrutina principal de **vpa** (usada también por los grupos de análisis de **transición** y **sensibilidad**)
Es llamada por **VIT**, **canvi**, **llepro**, **pro2**, **sensib**
Llama a **vpal**, **vpal2**, **calcul**, **calcu2**

Subrutina de trabajo para poblaciones virtuales

vpal

Grupo vpa

Es llamada por vpa

Llama a **secant**

Realiza el VPA estándar según la ecuación de captura

$$C(i) = \frac{F(i)}{Z(i)} N(i+1) \frac{1 - \exp\{-Z(i)\delta t(i)\}}{\exp\{-Z(i)\delta t(i)\}}$$

El algoritmo de solución de $F(i)$ utilizado es el de la secante (subrutina **secant**), con una sensibilidad, en principio de $1.E-12$, pero puede ser alterado automáticamente de forma temporal (cuando no se ha alcanzado en 50 iteraciones) o puede ser modificado de forma definitiva manualmente (ver **menu**).

vpa2

Grupo vpa

Es llamada por vpa

Realiza el análisis de cohortes de Pope según las ecuaciones:

$$N(n) = CT(n) Z(n)/F(n)$$

$$x1(i) = \left[\begin{array}{c} \frac{L^\infty - l(i)}{L^\infty - l(i+1)} \end{array} \right] \quad M/(2K)$$

$$N(i) = [N(i+1) \quad x1(i) \quad CT(i)] \quad x1(i)$$

$$F(i) = - M - \frac{\ln [N(i+1)/N(i)]}{\delta t(i)}$$

vpedat

Grupo te

Parámetros: (n,xincr,xlinf,xk,t0,mi,ma,talla,talled)

talled (100,50): matriz de probabilidades de la relación talla-edad

No tiene **common**

Es llamada por **te**

Construye una matriz de probabilidades para la relación talla-edad

ypr

Grupos **ypr**, **transición** y **sensibilidad**

Parámetros: (q,y,bio,ss,yy)

q	factor que multiplica a F
y	rendimiento por recluta total
bio	biomasa
ss	biomasa de reproductores
yy(j)	rendimiento por recluta por arte

Es llamada por **ypr200**, **sensib**, **canvi**

Calcula el rendimiento por recluta según:

$$Y = \sum_{i=1}^n \{q F(i) \bar{w}(i) \bar{N}(i)\}$$

donde $N(1) = 1$

$$\bar{N}(i) = N(i) \{1 - \exp(-Z(i) \delta t(i))\} / Z(i)$$

Asimismo calcula la biomasa ($\sum_{i=1}^n \{\bar{N}(i) \bar{w}(i)\}$) la biomasa de reproductores ($\sum_{i=1}^n \{\bar{N}(i) \bar{w}(i) pm(i)\}$) y el rendimiento por recluta de cada arte ($\sum_{i=1}^n \{q F(i,j) \bar{w}(i) \bar{N}(i)\}$).

ypr200

Grupo **VIT**. Subrutina general de rendimiento por recluta

Es llamada por **VIT**

Llama a **ypr**, **derypr**

Realiza el análisis de rendimiento por recluta en 200 puntos

yprn

Grupos **ypr**, **transición**

Parámetros: (y,bio,ss,yy,xxn)

Es llamada por **ypr200**, **canvi**

Calcula el rendimiento por recluta fuera del equilibrio

4 Listados del programa principal y las subrutinas

(las subrutinas son presentadas en orden alfabético)

VIT Programa principal

```
common p0,p1,p2,p3,p4,p5,p6,p7,p8,p9
common n,m,iet,clas1,xincr,x(100,4),xeps,ivc
common xlinf,xk,t0,a,b,xm,fterm,prop(4),fecun(100)
common talla(101),edat(101),c(100,4),ctot(100),w(101)
common dt(100),tmitj(100),wmitj(100),emitj(100)
common xn(100),fg(100,4),ftot(100),z(100)
common cfin(4),ffin(4)
character*40 fit1
character*80 titol
```

c
c
c

lectura de dades

```
xeps=1.e-12
write(*,*)'entra el nom del fitxer de dades'
read(*,100)fit1
100 format(a)
open(1,file=fit1,status='old')
read(1,100)titol
read(1,*)n,m,iet,clas1,xincr
do 1 i=1,n
zz=clas1+xincr*(i-1)
go to(2,3),iet
2 edat(i)=zz
go to 1
3 talla(i)=zz
1 read(1,*)(x(i,j),j=1,m)
go to(4,5),iet
4 edat(n+1)=edat(n)+xincr
go to 6
5 talla(n+1)=talla(n)+xincr
6 close(1)
open(2,file='vit.prn',status='unknown')
write(2,100)titol
```

c

```
write(*,*)'*****'
write(*,*)'*
write(*,*)'* 1 fa VPA estàndard
write(*,*)'*
write(*,*)'* 2 fa anàlisi de cohorts de Pope
write(*,*)'*
write(*,*)'*****'
write(*,*)'*
write(*,*)'* Entra la teva opció
write(*,*)'*
write(*,*)'*****'
read(*,*)ivc
```

c

```
call creix2
call tp2
call mnat2
```

```

    call fter2
    call pro2
    call fecu2
    call vpa
  9 call menu(men)
    go to (10,11,12,13,14,15,16),men
10 call param
    call vpa
    go to 9
11 call imp1
    call imp2
    go to 9
12 call ypr200
    go to 9
13 call canvi
    call vpa
    go to 9
14 call sensib
    call vpa
    go to 9
15 call te(titol)
    go to 9
16 stop
    end

```

```

subroutine calcu2
common p0,p1,p2,p3,p4,p5,p6,p7,p8,p9
common n,m,iet,clas1,xincr,x(100,4),xeps,ivc
common xlinf,xk,t0,a,b,xm,fterm,prop(4),fecun(100)
common talla(101),edat(101),c(100,4),ctot(100),w(101)
common dt(100),tmitj(100),wmitj(100),emitj(100)
common xn(100),fg(100,4),ftot(100),z(100)
common cfin(4),ffin(4)
do 10 i=1,n
  et1=exp(-z(i)*edat(i))
  et2=exp(-z(i)*edat(i+1))
  xnt=et2-et1
  til=et1*(1.-z(i)*exp(-xk*(edat(i)-t0))/(z(i)+xk))
  ti2=et2*(1.-z(i)*exp(-xk*(edat(i+1)-t0))/(z(i)+xk))
  tmitj(i)=xlinf*(ti2-til)/xnt
  til=et1*(edat(i)+1./z(i))
  ti2=et2*(edat(i+1)+1./z(i))
  emitj(i)=(ti2-til)/xnt
10 call simpl(i)
  return
end

```

```

subroutine calcul
common p0,p1,p2,p3,p4,p5,p6,p7,p8,p9
common n,m,iet,clas1,xincr,x(100,4),xeps,ivc
common xlinf,xk,t0,a,b,xm,fterm,prop(4),fecun(100)
common talla(101),edat(101),c(100,4),ctot(100),w(101)
common dt(100),tmitj(100),wmitj(100),emitj(100)
common xn(100),fg(100,4),ftot(100),z(100)
common cfin(4),ffin(4)
do 7 i=1,n+1
go to (8,9),iet
8 talla(i)=xlinf*(1.-exp(-xk*(edat(i)-t0)))
go to 7
9 edat(i)=t0-alog(1.-talla(i)/xlinf)/xk
7 w(i)=a*talla(i)**b
do 11 i=1,n
dt(i)=edat(i+1)-edat(i)
ctot(i)=0.
do 11 j=1,m
c(i,j)=x(i,j)*prop(j)
11 ctot(i)=ctot(i)+c(i,j)
return
end

```

```

subroutine canvi
dimension yy(4),xxn(100),p(4)
common p0,p1,p2,p3,p4,p5,p6,p7,p8,p9
common n,m,iet,clas1,xincr,x(100,4),xeps,ivc
common xlinf,xk,t0,a,b,xm,fterm,prop(4),fecun(100)
common talla(101),edat(101),c(100,4),ctot(100),w(101)
common dt(100),tmitj(100),wmitj(100),emitj(100)
common xn(100),fg(100,4),ftot(100),z(100)
common cfin(4),ffin(4)
character*12 fit
write(*,*)'*****'
write(*,*)'*
write(*,*)'*
write(*,*)'*                OPCIONS DE LA TRANSICIO
write(*,*)'*
write(*,*)'*****'
write(*,*)'*
write(*,*)'*    Mode A   1   canvi dels factors de proporció
write(*,*)'*                        2   alteració de F per un factor
write(*,*)'*                        3   canvi de mortalitats per pesca
write(*,*)'*
write(*,*)'*    Mode B   1   el canvi es produeix de cop
write(*,*)'*                        2   el canvi es produeix any per any
write(*,*)'*
write(*,*)'*****'
write(*,*)'*
write(*,*)'*    Solament les opcions Mode A=1 i 2 i Mode B = 1
write(*,*)'*    permeten entrada per teclat. Qualsevol altra com-
write(*,*)'*    binació necessita un fitxer de lectura.
write(*,*)'*
write(*,*)'*****'
write(*,*)'*
write(*,*)'*    Entra: anys per simular, Mode A, i Mode B
write(*,*)'*
write(*,*)'*****'
do 3 i=1,n
3  xxn(i)=xn(i)/xn(1)
  read(*,*)iter,modea,modeb
  call ypr(1.,y,bio,s,yy)
  write(2,101)
  write(2,102)y,bio,s,(yy(j),j=1,m)
  if (modea.eq.1.and.modeb.eq.1) go to 1
  if (modea.eq.2.and.modeb.eq.1) go to 20
  write(*,*)'entra el nom del fitxer amb els paràmetres del canvi
  read(*,100)fit
  open(4,file=fit,status='old')
  if(modea.eq.3.and.modeb.eq.1) go to 6
  go to 2
6  call llemor
  go to 2
  1  do 79 j=1,m
79  p(j)=prop(j)
  call llepro(1)
  go to 2
20  call llepmor(1)

```

```

c
c      Comença el procés iteratiu
c
  2 do 4 k=1,iter
    if(modeb.eq.1) go to 5
    if(modea.eq.1) go to 10
    call llemor
    go to 5
  10 call llepro(2)
  5  call yprn(y,bio,s,yy,xxn)
    call vectn(xxn)
  4  write(2,103)k,y,bio,s,(yy(j),j=1,m)
    call ypr(1.,y,bio,s,yy)
    write(2,104)y,bio,s,(yy(j),j=1,m)
    if(modea.eq.2.or.modea.eq.3) go to 80
    do 81 j=1,m
  81 prop(j)=p(j)
    call vpa
c
100 format(a)
101 format(///' ANALISI DE LA TRANSICIO')
102 format(/'any          Y/R  biomassa      S.S.B.          Y/R segons art
1//'ini',7f10.3/)
103 format(i3,7f10.3)
104 format(/'fin',7f10.3)
  80 return
    end

```

```

subroutine creix
common p0,p1,p2,p3,p4,p5,p6,p7,p8,p9
common n,m,iet,clas1,xincr,x(100,4),xeps,ivc
common xlinf,xk,t0,a,b,xm,fterm,prop(4),fecun(100)
common talla(101),edat(101),c(100,4),ctot(100),w(101)
common dt(100),tmitj(100),wmitj(100),emitj(100)
common xn(100),fg(100,4),ftot(100),z(100)
common cfin(4),ffin(4)
call creix1(iban)
if(iban) 1,1,2
  2 call creix2
  1 return
    end

```



```

subroutine creix1(iban)
common p0,p1,p2,p3,p4,p5,p6,p7,p8,p9
common n,m,iet,clas1,xincr,x(100,4),xeps,ivc
common xlinf,xk,t0,a,b,xm,fterm,prop(4),fecun(100)
common talla(101),edat(101),c(100,4),ctot(100),w(101)
common dt(100),tmitj(100),wmitj(100),emitj(100)
common xn(100),fg(100,4),ftot(100),z(100)
common cfin(4),ffin(4)
write(*,*)'els paràmetres de creixement són'
write(*,*)
write(*,*)'      L(inf) =',xlinf
write(*,*)'      K      =',xk
write(*,*)'      t(0)  =',t0
write(*,*)
write(*,*)'els vols mantenir? (s/n)'
call sino(iban)
return
end

```

```

subroutine creix2
common p0,p1,p2,p3,p4,p5,p6,p7,p8,p9
common n,m,iet,clas1,xincr,x(100,4),xeps,ivc
common xlinf,xk,t0,a,b,xm,fterm,prop(4),fecun(100)
common talla(101),edat(101),c(100,4),ctot(100),w(101)
common dt(100),tmitj(100),wmitj(100),emitj(100)
common xn(100),fg(100,4),ftot(100),z(100)
common cfin(4),ffin(4)
2 write(*,*)
write(*,*)'entra els paràmetres de creixement'
write(*,*)'L(inf) K t(0)'
read(*,*)xlinf,xk,t0
if(xlinf.le.talla(n+1)) go to 1
return
1 write(*,*)'L(inf) massa petit. No pot ser inferior a',talla(n+1)
go to 2
end

```

```

subroutine derypr(n,q,xm,ftot,w,dt,d)
dimension ftot(1),w(1),dt(1)
x1=1.
d=0.
s1=0.
do 6 i=1,n
f=ftot(i)
z=f*q+xm
x2=x1*exp(-z*dt(i))
dx=x1-x2
fz=f/z
fn=f*dt(i)*x2
d=d+fz*w(i)*(dx*(1.-q*(fz+s1))+fn*q)
x1=x2
s1=s1+f*dt(i)
6 continue
return
end

```

```

subroutine distrib(n,mi,ma,x,talled)
dimension x(1),y(50),talled(100,50)
do 2 k=1,ma-mi+1
2 y(k)=0.
do 1 i=1,n
do 1 k=1,ma-mi+1
1 y(k)=y(k)+talled(i,k)*x(i)
do 3 k=1,ma-mi+1
3 x(k)=y(k)
return
end

```

```

subroutine fecu
common p0,p1,p2,p3,p4,p5,p6,p7,p8,p9
common n,m,iet,clas1,xincr,x(100,4),xeps,ivc
common xlinf,xk,t0,a,b,xm,fterm,prop(4),fecun(100)
common talla(101),edat(101),c(100,4),ctot(100),w(101)
common dt(100),tmitj(100),wmitj(100),emitj(100)
common xn(100),fg(100,4),ftot(100),z(100)
common cfin(4),ffin(4)
call fecul(iban)
if(iban) 1,1,2
2 call fecu2
1 return
end

```

```

subroutine fecul(iban)
common p0,p1,p2,p3,p4,p5,p6,p7,p8,p9
common n,m,iet,clas1,xincr,x(100,4),xeps,ivc
common xlinf,xk,t0,a,b,xm,fterm,prop(4),fecun(100)
common talla(101),edat(101),c(100,4),ctot(100),w(101)
common dt(100),tmitj(100),wmitj(100),emitj(100)
common xn(100),fg(100,4),ftot(100),z(100)
common cfin(4),ffin(4)
write(*,*)'les proporcions de madurs son'
write(*,*)
do 1 i=1,n
go to (2,3),iet
2 write(*,*)' classe',i,' edat', edat(i),' proporció=',fecun(i)
go to 1
3 write(*,*)' classe',i,' talla', talla(i),' proporció=',fecun(i)
1 continue
write(*,*)
write(*,*)'les vols mantenir ? (s/n)'
call sino(iban)
return
end

```

```

subroutine fecu2
common p0,p1,p2,p3,p4,p5,p6,p7,p8,p9
common n,m,iet,clas1,xincr,x(100,4),xeps,ivc
common xlinf,xk,t0,a,b,xm,fterm,prop(4),fecun(100)
common talla(101),edat(101),c(100,4),ctot(100),w(101)
common dt(100),tmitj(100),wmitj(100),emitj(100)
common xn(100),fg(100,4),ftot(100),z(100)
common cfin(4),ffin(4)
write(*,*)
write(*,*)'entra les proporcions de madurs'
write(*,*)'quan ja tot siguin 1 entra -1'
write(*,*)
do 1 i=1,n
go to (5,6),iet
5 write(*,*)'classe',i,' edat',edat(i)
go to 7
6 write(*,*)'classe',i,' talla',talla(i)
7 read(*,*)zz
if(zz) 2,3,3
3 fecun(i)=zz
1 continue
2 do 4 k=i,n
4 fecun(k)=1.
return
end

```

```

subroutine fter
common p0,p1,p2,p3,p4,p5,p6,p7,p8,p9
common n,m,iet,clas1,xincr,x(100,4),xeps,ivc
common xlinf,xk,t0,a,b,xm,fterm,prop(4),fecun(100)
common talla(101),edat(101),c(100,4),ctot(100),w(101)
common dt(100),tmitj(100),wmitj(100),emitj(100)
common xn(100),fg(100,4),ftot(100),z(100)
common cfin(4),ffin(4)
call fter1(iban)
if(iban) 1,1,2
2 call fter2
1 return
end

```

```

subroutine fter1(iban)
common p0,p1,p2,p3,p4,p5,p6,p7,p8,p9
common n,m,iet,clas1,xincr,x(100,4),xeps,ivc
common xlinf,xk,t0,a,b,xm,fterm,prop(4),fecun(100)
common talla(101),edat(101),c(100,4),ctot(100),w(101)
common dt(100),tmitj(100),wmitj(100),emitj(100)
common xn(100),fg(100,4),ftot(100),z(100)
common cfin(4),ffin(4)
write(*,*)'la mortalitat natural es'
write(*,*)
write(*,*)'      F(n)  =',fterm
write(*,*)
write(*,*)'la vols mantenir ? (s/n)'
call sino(iban)
return
end

```

```

subroutine fter2
common p0,p1,p2,p3,p4,p5,p6,p7,p8,p9
common n,m,iet,clas1,xincr,x(100,4),xeps,ivc
common xlinf,xk,t0,a,b,xm,fterm,prop(4),fecun(100)
common talla(101),edat(101),c(100,4),ctot(100),w(101)
common dt(100),tmitj(100),wmitj(100),emitj(100)
common xn(100),fg(100,4),ftot(100),z(100)
common cfin(4),ffin(4)
write(*,*)
write(*,*)'entra la mortalitat per pesca terminal'
read(*,*)fterm
return
end

```

```

subroutine funct(f,fun,xm,xn,c,t)
z=xm+f
fun=(f/z)*xn*(1.-exp(-z*t))/(exp(-z*t))-c
return
end

```

```

function functw(i,e)
common p0,p1,p2,p3,p4,p5,p6,p7,p8,p9
common n,m,iet,clas1,xincr,x(100,4),xeps,ivc
common xlinf,xk,t0,a,b,xm,fterm,prop(4),fecun(100)
common talla(101),edat(101),c(100,4),ctot(100),w(101)
common dt(100),tmitj(100),wmitj(100),emitj(100)
common xn(100),fg(100,4),ftot(100),z(100)
common cfin(4),ffin(4)
functw=exp(-z(i)*e)*(1.-exp(-xk*(e-t0)))*b
return
end

```

```

subroutine impl
common p0,p1,p2,p3,p4,p5,p6,p7,p8,p9
common n,m,iet,clas1,xincr,x(100,4),xeps,ivc
common xlinf,xk,t0,a,b,xm,fterm,prop(4),fecun(100)
common talla(101),edat(101),c(100,4),ctot(100),w(101)
common dt(100),tmitj(100),wmitj(100),emitj(100)
common xn(100),fg(100,4),ftot(100),z(100)
common cfin(4),ffin(4)
call imp11
call imp12
call imp13
return
end

```

```

subroutine impl1
common p0,p1,p2,p3,p4,p5,p6,p7,p8,p9
common n,m,iet,clas1,xincr,x(100,4),xeps,ivc
common xlinf,xk,t0,a,b,xm,fterm,prop(4),fecun(100)
common talla(101),edat(101),c(100,4),ctot(100),w(101)
common dt(100),tmitj(100),wmitj(100),emitj(100)
common xn(100),fg(100,4),ftot(100),z(100)
common cfin(4),ffin(4)

```

c

```

write(2,100)
write(2,103)xlinf,xk,t0,a,b,xm
write(2,105)(prop(j),j=1,m)
return
100 format(///'*****')
103 format(/' L(inf) =',f10.2,' K =',f10.2,' t(0) =',f10.2//
1' a =',e10.3,' b =',f10.2//' M =',f10.2)
105 format(///' factors de proporció ',4f9.4)
end

```

```

subroutine impl2
dimension ctg(4),emg(4),tmg(4),ctotw(100),cw(100,4)
common p0,p1,p2,p3,p4,p5,p6,p7,p8,p9
common n,m,iet,clas1,xincr,x(100,4),xeps,ivc
common xlinf,xk,t0,a,b,xm,fterm,prop(4),fecun(100)
common talla(101),edat(101),c(100,4),ctot(100),w(101)
common dt(100),tmitj(100),wmitj(100),emitj(100)
common xn(100),fg(100,4),ftot(100),z(100)
common cfin(4),ffin(4)

c
write(2,108)
write(2,104)

c
do 1 i=1,n
1 write(2,101)i,edat(i),emitj(i),talla(i),tmitj(i),w(i),wmitj(i)
write(2,112)edat(n+1),talla(n+1),w(n+1)
write(2,102)
do 2 i=1,n
2 write(2,106)i,ctot(i),(c(i,j),j=1,m)

c
ct=0.
em=0.
tm=0.
do 3 j=1,m
ctg(j)=0.
tmg(j)=0.
3 emg(j)=0.

c
do 4 i=1,n
ct=ct+ctot(i)
em=em+ctot(i)*emitj(i)
tm=tm+ctot(i)*tmitj(i)
do 4 j=1,m
ctg(j)=ctg(j)+c(i,j)
tmg(j)=tmg(j)+c(i,j)*tmitj(i)
4 emg(j)=emg(j)+c(i,j)*emitj(i)
em=em/ct
tm=tm/ct
do 5 j=1,m
tmg(j)=tmg(j)/ctg(j)
5 emg(j)=emg(j)/ctg(j)

c
write(2,107)ct,(ctg(j),j=1,m)
write(2,109)em,(emg(j),j=1,m)
write(2,110)tm,(tmg(j),j=1,m)
return
101 format(i4,4f11.4,2g14.4)
102 format(//' Captures en nombres'//'clas          Captura total
lures per art'//)
104 format('/'clas          ed inf          ed mitj          tall inf          tall mitj
linf          pes mitj'//)
106 format(i4,f18.4,4f13.4)
107 format('/' Sum',f18.4,4f13.4)
108 format(//' Dades generals')
109 format('/' edat mitjana',f9.4,4f13.4)
110 format('/'talla mitjana',f9.4,4f13.4)
112 format(4x,2(f11.4,11x),g14.4)
end

```

```

subroutine impl3
dimension ctg(4), emg(4), tmg(4), ctotw(100), cw(100,4)
common p0,p1,p2,p3,p4,p5,p6,p7,p8,p9
common n,m,iet,clas1,xincr,x(100,4),xeps,ivc
common xlinf,xk,t0,a,b,xm,fterm,prop(4),fecun(100)
common talla(101),edat(101),c(100,4),ctot(100),w(101)
common dt(100),tmitj(100),wmitj(100),emitj(100)
common xn(100),fg(100,4),ftot(100),z(100)
common cfin(4),ffin(4)

c
write(2,111)
do 20 i=1,n
ctotw(i)=ctot(i)*wmitj(i)
do 21 j=1,m
21 cw(i,j)=c(i,j)*wmitj(i)
20 write(2,106) i,ctotw(i),(cw(i,j),j=1,m)

c
ct=0.
do 30 j=1,m
30 ctg(j)=0.

c
do 40 i=1,n
ct=ct+ctotw(i)
do 40 j=1,m
40 ctg(j)=ctg(j)+cw(i,j)

c
write(2,107) ct,(ctg(j),j=1,m)

c
do 50 j=1,m
50 ctg(j)=ctg(j)*100./ct
write(2,100) (ctg(j),j=1,m)

c
100 format(/' Percentatges',8x,4f13.4)
106 format(i4,f18.4,4f13.4)
107 format(/' Sum',f18.4,4f13.4)
111 format(//' Captures en pesos '//'clas          Captura total  Captu
1 per art'//)
return
end

```

```

subroutine imp2
common p0,p1,p2,p3,p4,p5,p6,p7,p8,p9
common n,m,iet,clas1,xincr,x(100,4),xeps,ivc
common xlinf,xk,t0,a,b,xm,fterm,prop(4),fecun(100)
common talla(101),edat(101),c(100,4),ctot(100),w(101)
common dt(100),tmitj(100),wmitj(100),emitj(100)
common xn(100),fg(100,4),ftot(100),z(100),cfin(4),ffin(4)
write(2,100)
xnfin=0.
ftotfin=0.
zfin=0.
spim=0.
sumt=0.
sume=0.
do 70 j=1,m
70 ffin(j)=0.
write(2,114)
do 14 i=1,n
xnfin=xnfin+xn(i)
ftotfin=ftotfin+ftot(i)*dt(i)
zfin=zfin+z(i)*dt(i)
pim=(xn(i)-xn(i)*exp(-z(i)*dt(i)))/z(i)
spim=spim+pim
sumt=sumt+pim*tmitj(i)
sume=sume+pim*emitj(i)
do 71 j=1,m
71 ffin(j)=ffin(j)+fg(i,j)*dt(i)
14 write(2,111)i,xn(i),pim,z(i),ftot(i),(fg(i,j),j=1,m)
ftotfin=ftotfin/(edat(n+1)-edat(1))
zfin=zfin/(edat(n+1)-edat(1))
sume=sume/spim
sumt=sumt/spim
do 72 j=1,m
72 ffin(j)=ffin(j)/(edat(n+1)-edat(1))
write(2,113)xnfin,spim
write(2,101)zfin,ftotfin,(ffin(j),j=1,m)
write(2,603)sume,sumt
spim=0.
xnfin=0.
write(2,124)
do 140 i=1,n
xn1=xn(i)*w(i)
xnfin=xnfin+xn1
pim=(xn(i)-xn(i)*exp(-z(i)*dt(i)))/(z(i))*wmitj(i)
spim=spim+pim
140 write(2,111)i,xn1,pim
write(2,113)xnfin,spim
100 format(///'          RESULTATS DE LA VPA')
101 format('/' mortalitats mitjanes           ',6f7.4)
111 format(i4,2f14.0,6f7.4)
113 format('/' Tot'2f14.0)
114 format(///' nombres i mortalitats'//
1'clas          Nombre      Nombre mitj          z   f(tot)      f per art'//)
124 format(///' pesos'//
1'clas          Pes          Pes mitj'//)
603 format('/' població: edat mitjana',f7.2,10x,'talla mitjana',f7.2
return
end

```



```

subroutine llemor
common p0,p1,p2,p3,p4,p5,p6,p7,p8,p9
common n,m,iet,clas1,xincr,x(100,4),xeps,ivc
common xlinf,xk,t0,a,b,xm,fterm,prop(4),fecun(100)
common talla(101),edat(101),c(100,4),ctot(100),w(101)
common dt(100),tmitj(100),wmitj(100),emitj(100)
common xn(100),fg(100,4),ftot(100),z(100)
common cfin(4),ffin(4)

```

c
c
c

Llegeix una matriu de mortalitats

```

do 7 i=1,n
ftot(i)=0.
read(4,*)(fg(i,j),j=1,m)
do 6 j=1,m
6 ftot(i)=ftot(i)+fg(i,j)
7 z(i)=ftot(i)+xm
return
end

```

```

subroutine llepmor(mode)
dimension pnou(4)
common p0,p1,p2,p3,p4,p5,p6,p7,p8,p9
common n,m,iet,clas1,xincr,x(100,4),xeps,ivc
common xlinf,xk,t0,a,b,xm,fterm,prop(4),fecun(100)
common talla(101),edat(101),c(100,4),ctot(100),w(101)
common dt(100),tmitj(100),wmitj(100),emitj(100)
common xn(100),fg(100,4),ftot(100),z(100)
common cfin(4),ffin(4)

```

c
c
c

Llegeix un vector de proporcions

```

go to(1,2),mode
1 write(*,*)'entra',m,' factors que han de multiplicar les F's'
read(*,*)(pnou(j),j=1,m)
6 do 3 i=1,n
ftot(i)=0.
do 5 j=1,m
fg(i,j)=fg(i,j)*pnou(j)
5 ftot(i)=ftot(i)+fg(i,j)
3 z(i)=xm+ftot(i)
return
2 read(4,*)(pnou(j),j=1,m)
go to 6
end

```

```

subroutine llepro(mode)
dimension pnou(4)
common p0,p1,p2,p3,p4,p5,p6,p7,p8,p9
common n,m,iet,clas1,xincr,x(100,4),xeps,ivc
common xlinf,xk,t0,a,b,xm,fterm,prop(4),fecun(100)
common talla(101),edat(101),c(100,4),ctot(100),w(101)
common dt(100),tmitj(100),wmitj(100),emitj(100)
common xn(100),fg(100,4),ftot(100),z(100)
common cfin(4),ffin(4)

```

c
c
c

Llegeix un vector de proporcions

```

go to(1,2),mode
1 call pro
go to 4
2 read(4,*)(prop(j),j=1,m)
4 call vpa
return
end

```

```

subroutine matrius(n,m,mi,ma,x,talled)
dimension x(100,4),talled(100,50),treball(100)
do 300 j=1,m
do 301 i=1,n
301 treball(i)=x(i,j)
call distrib(n,mi,ma,treball,talled)
do 302 k=1,ma-mi+1
302 x(k,j)=treball(k)
300 continue
return
end

```



```

subroutine mnat
common p0,p1,p2,p3,p4,p5,p6,p7,p8,p9
common n,m,iet,clas1,xincr,x(100,4),xeps,ivc
common xlinf,xk,t0,a,b,xm,fterm,prop(4),fecun(100)
common talla(101),edat(101),c(100,4),ctot(100),w(101)
common dt(100),tmitj(100),wmitj(100),emitj(100)
common xn(100),fg(100,4),ftot(100),z(100)
common cfin(4),ffin(4)
call mnat1(iban)
if(iban) 1,1,2
2 call mnat2
1 return
end

```

```

subroutine mnat1(iban)
common p0,p1,p2,p3,p4,p5,p6,p7,p8,p9
common n,m,iet,clas1,xincr,x(100,4),xeps,ivc
common xlinf,xk,t0,a,b,xm,fterm,prop(4),fecun(100)
common talla(101),edat(101),c(100,4),ctot(100),w(101)
common dt(100),tmitj(100),wmitj(100),emitj(100)
common xn(100),fg(100,4),ftot(100),z(100)
common cfin(4),ffin(4)
write(*,*)'la mortalitat natural es'
write(*,*)
write(*,*)'          M   =',xm
write(*,*)
write(*,*)'la vols mantenir ? (s/n)'
call sino(iban)
return
end

```

```

subroutine mnat2
common p0,p1,p2,p3,p4,p5,p6,p7,p8,p9
common n,m,iet,clas1,xincr,x(100,4),xeps,ivc
common xlinf,xk,t0,a,b,xm,fterm,prop(4),fecun(100)
common talla(101),edat(101),c(100,4),ctot(100),w(101)
common dt(100),tmitj(100),wmitj(100),emitj(100)
common xn(100),fg(100,4),ftot(100),z(100)
common cfin(4),ffin(4)
write(*,*)
write(*,*)'entra la mortalitat natural'
read(*,*)xm
return
end

```

```

subroutine param
common p0,p1,p2,p3,p4,p5,p6,p7,p8,p9
common n,m,iet,clas1,xincr,x(100,4),xeps,ivc
common xlinf,xk,t0,a,b,xm,fterm,prop(4),fecun(100)
common talla(101),edat(101),c(100,4),ctot(100),w(101)
common dt(100),tmitj(100),wmitj(100),emitj(100)
common xn(100),fg(100,4),ftot(100),z(100)
common cfin(4),ffin(4)
10 write(*,*)'*****'
write(*,*)'*
write(*,*)'*
MODIFICACIO DE PARAMETRES
write(*,*)'*
write(*,*)'*
write(*,*)'*
1 Creixement
write(*,*)'*
write(*,*)'*
2 Relació talla-pes
write(*,*)'*
write(*,*)'*
3 Mortalitat natural
write(*,*)'*
write(*,*)'*
4 F terminal
write(*,*)'*
write(*,*)'*
5 Factors de proporció
write(*,*)'*
write(*,*)'*
6 Proporció de madurs
write(*,*)'*
write(*,*)'*
7 Torna al Menú principal
write(*,*)'*
read(*,*)kpar
go to (1,2,3,4,5,6,7),kpar
1 call creix
go to 10
2 call tp
go to 10
3 call mnat
go to 10
4 call fter
go to 10
5 call pro
go to 10
6 call fecu
go to 10
7 return
end

```

```

subroutine pro
common p0,p1,p2,p3,p4,p5,p6,p7,p8,p9
common n,m,iet,clas1,xincr,x(100,4),xeps,ivc
common xlinf,xk,t0,a,b,xm,fterm,prop(4),fecun(100)
common talla(101),edat(101),c(100,4),ctot(100),w(101)
common dt(100),tmitj(100),wmitj(100),emitj(100)
common xn(100),fg(100,4),ftot(100),z(100)
common cfin(4),ffin(4)
call prol(iban)
if(iban) 1,1,2
2 call pro2
1 return
end

```

```

subroutine prol(iban)
common p0,p1,p2,p3,p4,p5,p6,p7,p8,p9
common n,m,iet,clas1,xincr,x(100,4),xeps,ivc
common xlinf,xk,t0,a,b,xm,fterm,prop(4),fecun(100)
common talla(101),edat(101),c(100,4),ctot(100),w(101)
common dt(100),tmitj(100),wmitj(100),emitj(100)
common xn(100),fg(100,4),ftot(100),z(100)
common cfin(4),ffin(4)
write(*,*)'els factors de proporció son'
write(*,*)
do 1 j=1,m
1 write(*,*)'    art',j,' factor de proporció =',prop(j)
write(*,*)
write(*,*)'els vols mantenir ? (s/n)'
call sino(iban)
return
end

```

```

subroutine pro2
dimension percen(4),wtart(4)
common p0,p1,p2,p3,p4,p5,p6,p7,p8,p9
common n,m,iet,clas1,xincr,x(100,4),xeps,ivc
common xlinf,xk,t0,a,b,xm,fterm,prop(4),fecun(100)
common talla(101),edat(101),c(100,4),ctot(100),w(101)
common dt(100),tmitj(100),wmitj(100),emitj(100)
common xn(100),fg(100,4),ftot(100),z(100)
common cfin(4),ffin(4)
write(*,*)
3 write(*,*)'      TRANSFORMACIO A LA CAPTURA TOTAL'
write(*,*)
write(*,*)'0   per transformar segons nombres d''exemplars'
write(*,*)'1   per transformar segons pesos'
read(*,*)iban
if(iban) 52,52,81
52 write(*,*)
write(*,*)'entra proporcions dels',m,' arts'
read(*,*)(prop(j),j=1,m)
1 return
81 if(xn(1)) 10,10,2
10 do 11 j=1,m
11 prop(j)=1.
call vpa
2 write(*,*)'entra pes total'
read(*,*)ptotal
91 write(*,*)'entra percentatges dels',m,' arts'
read(*,*)(percen(j),j=1,m)
sum=0.
do 90 j=1,m
90 sum=sum+percen(j)
if(sum.ne.100) go to 91
55 do 40 j=1,m
40 wtart(j)=0.
do 41 i=1,n
do 41 j=1,m
41 wtart(j)=wtart(j)+x(i,j)*wmitj(i)
do 92 j=1,m
92 prop(j)=ptotal*percen(j)/(100.*wtart(j))
call vpa
cw=0.
do 50 i=1,n
do 50 j=1,m
50 cw=cw+x(i,j)*prop(j)*wmitj(i)
if(abs(cw-ptotal).gt.ptotal*1.e-6) go to 55
return
end

```

```

subroutine secant(c,x,xm,xn,t,xeps)
eps=xeps
3 x0=x
kont=0
delta=x/10
x1=x0+delta
call funct(x0,f0,xm,xn,c,t)
1 call funct(x1,f1,xm,xn,c,t)
kont=kont+1
if(kont.gt.50) go to 4
dif=f0-f1
if(abs(dif).le.eps) go to 2
x2=x1-f1*((x1-x0)/(f1-f0))
x0=x1
f0=f1
x1=x2
go to 1
2 x=x1
return
4 eps=eps*10
write(*,*)'precisió de la VPA disminuda a',eps
go to 3
end

```



```

        go to (102,103),ni(2)
103 xk=pa(2)+pa(2)*prct*(-1)**i2
    marca(2)=mmp(i2)
102 do 1 i3=1,ni(3)
    go to (104,105),ni(3)
105 t0=pa(3)+pa(3)*prct*(-1)**i3
    marca(3)=mmp(i3)
104 do 1 i4=1,ni(4)
    go to (106,107),ni(4)
107 a=pa(4)+pa(4)*prct*(-1)**i4
    marca(4)=mmp(i4)
106 do 1 i5=1,ni(5)
    go to (108,109),ni(5)
109 b=pa(5)+pa(5)*prct*(-1)**i5
    marca(5)=mmp(i5)
108 do 1 i6=1,ni(6)
    go to (110,111),ni(6)
111 xm=pa(6)+pa(6)*prct*(-1)**i6
    marca(6)=mmp(i6)
110 do 1 i7=1,ni(7)
    go to (112,113),ni(7)
113 fterm=pa(7)+pa(7)*prct*(-1)**i7
    marca(7)=mmp(i7)
112 do 1 i8=1,ni(8)
    go to (114,115),ni(8)
115 prop(1)=pa(8)+pa(8)*prct*(-1)**i8
    marca(8)=mmp(i8)
114 do 1 i9=1,ni(9)
    go to (116,117),ni(9)
117 prop(2)=pa(9)+pa(9)*prct*(-1)**i9
    marca(9)=mmp(i9)
116 do 1 i10=1,ni(10)
    go to (118,119),ni(10)
119 prop(3)=pa(10)+pa(10)*prct*(-1)**i10
    marca(10)=mmp(i10)
118 do 1 i11=1,ni(11)
    go to (120,121),ni(11)
121 prop(4)=pa(11)+pa(11)*prct*(-1)**i11
    marca(11)=mmp(i11)
120 call calcul
    call vpa
    call ypr(1.,y,bio,s,yy)
    write(2,122)(marca(j),j=1,11),y,bio,s,(yy(j),j=1,m)
1 continue
    xlinf=pa(1)
    xk=pa(2)
    t0=pa(3)
    a=pa(4)
    b=pa(5)
    xm=pa(6)
    fterm=pa(7)
    prop(1)=pa(8)
    prop(2)=pa(9)
    prop(3)=pa(10)
    prop(4)=pa(11)
    return
122 format(3a1,1x,2a1,1x,2a1,1x,4a1,3f10.2,4f7.2)
150 format(///' ANALISI DE SENSIBILITAT'///' Factor',f6.2,'

```

```

1parametres =',11i3)
151 format(//' parametres          Y/R      Biomassa      SSB          Y/R
1 art'/)
end

```

```

subroutine simpl(i)
common p0,p1,p2,p3,p4,p5,p6,p7,p8,p9
common n,m,iet,clas1,xincr,x(100,4),xeps,ivc
common xlinf,xk,t0,a,b,xm,fterm,prop(4),fecun(100)
common talla(101),edat(101),c(100,4),ctot(100),w(101)
common dt(100),tmitj(100),wmitj(100),emitj(100)
common xn(100),fg(100,4),ftot(100),z(100)
common cfin(4),ffin(4)
et1=exp(-z(i)*edat(i))
et2=exp(-z(i)*edat(i+1))
xnt=xn(i)*(et2-et1)
aa=edat(i)
bb=edat(i+1)
nn=20
h=(bb-aa)/nn
wmitj(i)=functw(i,aa)+functw(i,bb)
do 1 k=1,nn-1,2
1 wmitj(i)=wmitj(i)+functw(i,aa+h*k)*4
do 2 k=2,nn-2,2
2 wmitj(i)=wmitj(i)+functw(i,aa+h*k)*2
wmitj(i)=-wmitj(i)*a*xn(i)*z(i)*xlinf**b*h/(xnt*3)
return
end

```

```

subroutine sino(i)
data in,jn/'N','n'/
data is,js/'S','s'/
data iy,jy/'Y','y'/
1 read(*,2)j
if(j.eq.in.or.j.eq.jn) go to 3
if(j.ne.is.and.j.ne.js.and.j.ne.iy.and.j.ne.jy) go to 1
i=0
return
3 i=1
return
2 format(a1)
end

```

```

subroutine te(titol)
dimension talled(100,50),xx(100,4)
common p0,p1,p2,p3,p4,p5,p6,p7,p8,p9
common n,m,iet,clas1,xincr,x(100,4),xeps,ivc
common xlinf,xk,t0,a,b,xm,fterm,prop(4),fecun(100)
common talla(101),edat(101),c(100,4),ctot(100),w(101)
common dt(100),tmitj(100),wmitj(100),emitj(100)
common xn(100),fg(100,4),ftot(100),z(100)
common cfin(4),ffin(4)
character*12 fit3
character*80 titol
100 format(a)
c
c programa de transformació de talles a edats
c
kkk=1
1 write(*,*)'1 el fa de les dades originals'
write(*,*)'2 el fa de les dades afectades per les proporcions'
write(*,*)'3 el fa de les dades ajustades per la VPA'
read(*,*)iban
if(iban.eq.3) go to 2009
call vpedat(n,xincr,xlinf,xk,t0,mi,ma,talla,talled)
write(*,*)'entra el nom del fitxer de sortida'
read(*,100)fit3
open(3,file=fit3,status='unknown')
write(3,100)titol
nedat=ma-mi+1
write(3,*)nedat,m,kkk,mi,kkk
go to (2003,2004,2009),iban
2003 do 2005 i=1,n
do 2005 j=1,m
2005 xx(i,j)=x(i,j)
go to 2006
2004 do 2007 i=1,n
do 2007 j=1,m
2007 xx(i,j)=c(i,j)
2006 call matrius(n,m,mi,ma,xx,talled)
do 2008 k=1,nedat
2008 write(3,*)(xx(k,j),j=1,m)
close(3)
return
2009 write(*,*)'disculpa: aquesta opció encara no és a punt'
go to 1
end

```

```

subroutine tp
common p0,p1,p2,p3,p4,p5,p6,p7,p8,p9
common n,m,iet,clas1,xincr,x(100,4),xeps,ivc
common xlinf,xk,t0,a,b,xm,fterm,prop(4),fecun(100)
common talla(101),edat(101),c(100,4),ctot(100),w(101)
common dt(100),tmitj(100),wmitj(100),emitj(100)
common xn(100),fg(100,4),ftot(100),z(100)
common cfin(4),ffin(4)
call tp1(iban)
if(iban) 1,1,2
2 call tp2
1 return
end

```

```

subroutine tp1(iban)
common p0,p1,p2,p3,p4,p5,p6,p7,p8,p9
common n,m,iet,clas1,xincr,x(100,4),xeps,ivc
common xlinf,xk,t0,a,b,xm,fterm,prop(4),fecun(100)
common talla(101),edat(101),c(100,4),ctot(100),w(101)
common dt(100),tmitj(100),wmitj(100),emitj(100)
common xn(100),fg(100,4),ftot(100),z(100)
common cfin(4),ffin(4)
write(*,*)'els paràmetres de la relació talla-pes son'
write(*,*)
write(*,*)'          a   =' ,a
write(*,*)'          b   =' ,b
write(*,*)
write(*,*)'els vols mantenir ? (s/n) '
call sino(iban)
return
end

```

```

subroutine tp2
common p0,p1,p2,p3,p4,p5,p6,p7,p8,p9
common n,m,iet,clas1,xincr,x(100,4),xeps,ivc
common xlinf,xk,t0,a,b,xm,fterm,prop(4),fecun(100)
common talla(101),edat(101),c(100,4),ctot(100),w(101)
common dt(100),tmitj(100),wmitj(100),emitj(100)
common xn(100),fg(100,4),ftot(100),z(100)
common cfin(4),ffin(4)
write(*,*)
write(*,*)'entra els paràmetres de la relació talla-pes'
write(*,*)' a   b'
read(*,*)a,b
return
end

```

```

subroutine vectn(xxn)
dimension kkk(1),ppp(1),treb(100),xxn(100)
common p0,p1,p2,p3,p4,p5,p6,p7,p8,p9
common n,m,iet,clas1,xincr,x(100,4),xeps,ivc
common xlinf,xk,t0,a,b,xm,fterm,prop(4),fecun(100)
common talla(101),edat(101),c(100,4),ctot(100),w(101)
common dt(100),tmitj(100),wmitj(100),emitj(100)
common xn(100),fg(100,4),ftot(100),z(100)
common cfin(4),ffin(4)
treb(1)=1.
do 3 i=2,n
3  treb(i)=xxn(i-1)*exp(-z(i-1))
do 8 i=1,n
8  xxn(i)=treb(i)
return
end

```

```

subroutine vpa
common p0,p1,p2,p3,p4,p5,p6,p7,p8,p9
common n,m,iet,clas1,xincr,x(100,4),xeps,ivc
common xlinf,xk,t0,a,b,xm,fterm,prop(4),fecun(100)
common talla(101),edat(101),c(100,4),ctot(100),w(101)
common dt(100),tmitj(100),wmitj(100),emitj(100)
common xn(100),fg(100,4),ftot(100),z(100)
common cfin(4),ffin(4)

```

```

c
call calcul
c
go to (1,2),ivc
1 call vpa1
go to 3
2 call vpa2
c
3 call calcul2
c
return
end

```

```

subroutine vpa1
common p0,p1,p2,p3,p4,p5,p6,p7,p8,p9
common n,m,iet,clas1,xincr,x(100,4),xeps,ivc
common xlinf,xk,t0,a,b,xm,fterm,prop(4),fecun(100)
common talla(101),edat(101),c(100,4),ctot(100),w(101)
common dt(100),tmitj(100),wmitj(100),emitj(100)
common xn(100),fg(100,4),ftot(100),z(100)
common cfin(4),ffin(4)

```

```

c
ftot(n)=fterm
z(n)=xm+ftot(n)
do 7 j=1,m
7 fg(n,j)=ftot(n)*c(n,j)/ctot(n)
xn(n)=ctot(n)*z(n)/(ftot(n)*(1.-exp(-z(n)*dt(n))))
c
do 10 ii=2,n
i=n-ii+1
ftot(i)=ftot(i+1)
call secant(ctot(i),ftot(i),xm,xn(i+1),dt(i),xeps)
z(i)=xm+ftot(i)
xn(i)=xn(i+1)*exp(z(i)*dt(i))
do 13 j=1,m
13 fg(i,j)=ftot(i)*c(i,j)/ctot(i)
10 continue
return
end

```

```

subroutine vpa2
common p0,p1,p2,p3,p4,p5,p6,p7,p8,p9
common n,m,iet,clas1,xincr,x(100,4),xeps,ivc
common xlinf,xk,t0,a,b,xm,fterm,prop(4),fecun(100)
common talla(101),edat(101),c(100,4),ctot(100),w(101)
common dt(100),tmitj(100),wmitj(100),emitj(100)
common xn(100),fg(100,4),ftot(100),z(100)
common cfin(4),ffin(4)

```

c

```

ftot(n)=fterm
z(n)=ftot(n)+xm
xn(n)=ctot(n)*z(n)/ftot(n)
do 1 ii=2,n
i=n-ii+1
xl=((xlinf-talla(i))/(xlinf-talla(i+1)))**(xm/(2.*xk))
xn(i)=(xn(i+1)*xl+ctot(i))*xl
ftot(i)=-xm-alog(xn(i+1)/xn(i))/dt(i)
z(i)=ftot(i)+xm
do 1 j=1,m
1 fg(i,j)=ftot(i)*c(i,j)/ctot(i)
return
end

```

```

subroutine vpedat(n,xincr,xlinf,xk,t0,mi,ma,talla,talled)
dimension talla(1),talled(100,50)
mi=ifix(t0-alog(1.-talla(1)/xlinf)/xk)
ma=ifix(t0-alog(1.-(talla(n)+xincr)/xlinf)/xk)
do 300 kk=1,ma-mi+1
k=mi+kk-1
ta=xlinf*(1.-exp(xk*(t0-k)))
tb=xlinf*(1.-exp(xk*(t0-(k+1))))
do 302 i=1,n
t1=talla(i)
t2=talla(i)+xincr
if(ta.le.t1.and.tb.gt.t2) go to 301
if(tb.lt.t1.or.ta.gt.t2) go to 303
if(ta.le.t1.and.tb.lt.t2) go to 304
if(ta.ge.t1.and.tb.gt.t2) go to 305
talled(i,kk)=(tb-ta)/(t2-t1)
go to 302
305 talled(i,kk)=(t2-ta)/(t2-t1)
go to 302
304 talled(i,kk)=(tb-t1)/(t2-t1)
go to 302
303 talled(i,kk)=0.
go to 302
301 talled(i,kk)=1.
302 continue
300 continue
return
end

```

```

subroutine ypr(q,y,bio,s,yy)
dimension yy(4)
common p0,p1,p2,p3,p4,p5,p6,p7,p8,p9
common n,m,iet,clas1,xincr,x(100,4),xeps,ivc
common xlinf,xk,t0,a,b,xm,fterm,prop(4),fecun(100)
common talla(101),edat(101),c(100,4),ctot(100),w(101)
common dt(100),tmitj(100),wmitj(100),emitj(100)
common xn(100),fg(100,4),ftot(100),z(100)
common cfin(4),ffin(4)
xx=1.
y=0.
bio=0.
s=0.
do 21 j=1,m
21 yy(j)=0.
do 11 i=1,n
f=ftot(i)*q
zz=f+xm
cc=xx*(1.-exp(-zz*dt(i)))*wmitj(i)/zz
y=y+cc*f
bio=bio+cc
s=s+cc*fecun(i)
do 20 j=1,m
20 yy(j)=yy(j)+cc*q*fg(i,j)
11 xx=xx*exp(-zz*dt(i))
return
end

```



```

subroutine ypr200
dimension yy(4),ym(5),iy(5),yyy(5)
common p0,p1,p2,p3,p4,p5,p6,p7,p8,p9
common n,m,iet,clas1,xincr,x(100,4),xeps,ivc
common xlinf,xk,t0,a,b,xm,fterm,prop(4),fecun(100)
common talla(101),edat(101),c(100,4),ctot(100),w(101)
common dt(100),tmitj(100),wmitj(100),emitj(100)
common xn(100),fg(100,4),ftot(100),z(100)
common cfin(4),ffin(4)

c
write(2,100)
write(2,101)(i,fecun(i),i=1,n)

c
q=0.
call derypr(n,q,xm,ftot,wmitj,dt,d)
write(2,105)d

c
write(*,*)'*****'
write(*,*)'*
write(*,*)'*
write(*,*)'*
write(*,*)'*****'
write(*,*)'*
write(*,*)'*
write(*,*)'*
write(*,*)'*
write(*,*)'*
write(*,*)'*****'
read(*,*)iban
xin=0.01
k=200
do 10 j=1,m+1
iy(j)=0
10 ym(j)=0.
iban01=0
do 200 kq=1,k+1
call ypr(q,y,bio,s,yy)
go to (43,41),iban
43 if(iban01) 40,40,41
40 call derypr(n,q,xm,ftot,wmitj,dt,dd)
if(dd.gt.d/10) go to 41
iban01=1
write(2,*)
write(2,120)qq,y,bio,s,(yy(j),j=1,m)
write(2,*)
41 qq=q
go to (1,2),iban
1 if(q.eq.0.00) go to 2
if(q.gt.0.999.and.q.lt.1.001) go to 2
if(q.gt.1.999.and.q.lt.2.001) go to 2
YYY(m+1)=y
do 5 j=1,m
5 YYY(j)=yy(j)
do 3 j=1,m+1
if(iy(j)) 4,4,3
4 if(YYY(j).gt.ym(j)) go to 30
iy(j)=1
qq=q-xin

```

```

call ypr(qq,y,bio,s,YY)
go to 2
30 ym(j)=yyy(j)
3 continue
go to 200
2 write(2,120) qq,y,bio,s,(yy(j),j=1,m)
200 q=q+xin

```

c

```

return
100 format(///// 'ANALISI DE RENDIMENT PER RECLUTA')
101 format(/// 'Proporcions de madurs'//9(i3,f5.2))
105 format(/// 'Pendent en l''origen =',g15.6///' Factor Y/R
liomassa S.S.B Y/R segons art'/)
120 format(f7.2,3g13.5,4f7.2)
end

```

```

subroutine yprn(y,bio,s,yy,xxn)
dimension yy(4),xxn(100)
common p0,p1,p2,p3,p4,p5,p6,p7,p8,p9
common n,m,iet,clas1,xincr,x(100,4),xeps,ivc
common xlinf,xk,t0,a,b,xm,fterm,prop(4),fecun(100)
common talla(101),edat(101),c(100,4),ctot(100),w(101)
common dt(100),tmitj(100),wmitj(100),emitj(100)
common xn(100),fg(100,4),ftot(100),z(100)
common cfin(4),ffin(4)
y=0.
bio=0.
s=0.
do 21 j=1,m
21 yy(j)=0.
do 11 i=1,n
f=ftot(i)
zz=f+xm
cc=xxn(i)*(1.-exp(-zz*dt(i)))*wmitj(i)/zz
y=y+cc*f
bio=bio+cc
s=s+cc*fecun(i)
do 20 j=1,m
20 yy(j)=yy(j)+cc*fg(i,j)
11 continue
return
end

```