

BIBLIOTECA DE PROGRAMAS DE ANALIS
SIS ESTADISTICO.

MANUAL DEL USUARIO

BIBLIOTECA DE PROGRAMAS DE
ANALISIS ESTADISTICO.

MANUAL DEL USUARIO *

* La información de este manual se ha adaptado del BMD, Biomedical Computer Programs, W.J. Dixon Editor, University of - California Press, 1974, 773 pp.

I N D I C E

	<u>Pág.</u>
I. INTRODUCCION	4
II. BREVE DESCRIPCION DE LOS PROGRAMAS BMD	5
III. PREPARACION DE LOS DATOS	13
IV. PREPARACION DE LAS FICHAS DE CONTROL DE PRO GRAMA	21
V. PREPARACION DE LAS FICHAS DE CONTROL DEL - SISTEMA	34
VI. BIBLIOGRAFIA	38
VII. APENDICE. DESCRIPCION DEL MODULO BMDASM	40

I. INTRODUCCION

Muchos problemas en la investigación geológica y minera requieren extensos análisis de grandes cantidades de datos. El proceso de manipulación de datos debe ser, tanto como sea posible, - automático y rápido. El investigador debe tener las herramientas apropiadas para efectuar los tipos de análisis que su investigación requiere.

La inclusión de programas en este manual, y en sus formas específicas, ha sido guiada por la demanda dentro del IGME de procedimientos estadísticos y matemáticos que asistan en una gran variedad de problemas de investigación.

Los programas están ordenados en seis clases desde el punto de vista de la conveniencia de organización de este manual. Debido a la naturaleza de la estadística, esta clasificación no es única. Así, los programas de análisis multivariante de la varianza que se han incluido en la clase V, podrían también haberlo sido en la M.

Las clases son:

- D Descripción y tabulación
- M Análisis multivariante
- R Análisis de la regresión
- S Programas especiales
- T Análisis de series de tiempo
- V Análisis de la varianza

Cada programa está diseñado para realizar un cierto tipo de tarea. Sin embargo, un resultado particular puede obtenerse a través de más de un programa. En el párrafo siguiente se presenta una visión general de los distintos programas disponibles y los análisis que realizan.

II. BREVE DESCRIPCION DE LOS PROGRAMAS BMD

Clase D - Descripción y Tabulación

BMD01D: Descripción de Datos

Calcula promedios simples y medidas de dispersión, eliminando de los cálculos los valores que el usuario desee. - Puede eliminar blancos o valores específicos. O ambos o ninguno.

BMD02D: Correlación con Transgeneración

Calcula coeficientes de correlación simple, promedios y medidas de dispersión de las variables de entrada y/o de las transgeneradas de los casos seleccionados cuyos valores para las variables especificadas tienen una determinada relación lógica definida por una expresión Booleana. - Se incluye en la salida gráficos de cada par de variables.

BMD03D: Correlación con Eliminación de Item.

Calcula una matriz de correlación simple omitiendo los valores de las variables que indique el usuario.

Se puede especificar, para su eliminación, cualquier código incluido el blanco.

BMD04D: Contador de Frecuencia de Alfanuméricos

Calcula frecuencias de caracteres válidos que ocupen una columna. Se cuenta cualquier carácter válido numérico, alfabético o especial. La entrada puede ser por ficha, por cinta código EBCDIC o BCD; en cualquier caso los caracteres se imprimen en código EBCDIC.

BMD05D: Gráficos incluyendo Histogramas

Produce gráficos de nubes de puntos o histogramas. Se utilizan dos métodos para el dibujo de nubes de puntos.

El primer método genera un gráfico en una página con 100 unidades horizontales y 500 verticales. Automáticamente los puntos se escalan de acuerdo con estas dimensiones.

El segundo método genera un gráfico en varias páginas con tantas unidades verticales como valores haya de la variable base. La variable base no se escala, y sus valores - (casos) deben estar ordenados consecutivamente.

También se puede generar un histograma, que saldrá en una sola página, con un máximo de 34 intervalos. Tiene que especificarse el tamaño del intervalo.

BMD06D: Descripción de Estratos.

Se separan los casos en grupos de acuerdo con los intervalos de una variable, la condicionante. Para estos grupos se calculan frecuencias, medias, varianzas, desviaciones típicas, errores típicos de las medias y coeficientes de correlación.

BMD07D: Descripción de Estratos con Histogramas.

Agrupar los datos en un número de grupos especificado de acuerdo con el orden de entrada de los datos o bien en grupos cuyos valores para una variable base están dentro de intervalos cuyos límites fija el usuario. Para estos grupos, se imprimen histogramas para cada variable. El número de clases o categorías de los histogramas pueden especificarse o bien ser calculados por el programa. Para cada grupo se calculan las medias, desviaciones típicas y los coeficientes de correlación. También para los grupos combinados de una variable se calculan las medias y las desviaciones típicas. Así mismo se calcula una tabla de análisis de varianza. Se pueden especificar para todas las variables excepto para las variables base, valores especiales con el fin de eliminar de los cálculos ciertos valores o códigos. El programa distingue entre blancos y ceros a efectos de eliminación de valores.

BMD08D: Tabulación Cruzada con Agrupamiento de Variables.

Calcula tablas de frecuencia de doble entrada para rangos especificados de las variables originales, de las variables después de la transgeneración, de las variables agrupadas o de cualquier combinación de las anteriores. Los -

datos pueden ser positivos o negativos pero siempre enteros. No se admiten datos en cuyos campos haya perforado un punto decimal. Véase 2S.

BMD09D: Tabulación Cruzada, Datos Incompletos.

Genera tablas cruzadas de los datos de entrada, eliminando de ellas los valores o códigos especiales que se hayan especificado para designar valores falsos. El programa distingue entre blancos y ceros a efectos de eliminación. Pueden usarse distintos valores especiales o códigos para la misma variable. En la salida quedan reflejados los valores o códigos eliminados. Se utilizan los códigos -999 y -999.00999 para indicar en la salida los valores falsos de una variable transgenerada que han sido producidos por la transgeneración de otra variable que contenía un valor falso para este caso. Véase 2S.

BMD10D: Configuraciones de datos para Dicotomía.

Halla frecuencias y configuraciones de cualquier código determinado que se defina en los datos de entrada. Un código usado frecuentemente es el que representa valores falsos. Imprime ceros para designar el código o los valores falsos y unos para los demás. Imprime así mismo una matriz de datos de ceros y unos. Se calculan las frecuencias del código o del valor falso. Los casos que tienen el código especificado se identifican por números de ítem que corresponden al orden en que entran los datos. Si se desea se pueden obtener configuraciones de datos después de la eliminación.

BMD11D: Configuración de datos para Policotomías.

Imprime configuraciones de datos de una columna con números de ítem o de casos para poder identificar los casos que contienen esta configuración. Si se desea, se pueden recodificar los datos originales antes de imprimir las configuraciones.

BMD12D: Correlación Asimétrica con Datos Falsos.

Calcula grandes matrices de correlación o partes de las mismas de datos que contienen datos falsos. Las correlaciones se calculan usando todos los pares posibles, señalando con blancos los valores falsos.

BMD13D: Programa t.

Calcula estadísticos t y niveles de probabilidad asociados para la igualdad de las medias de dos grupos, mediante estimadores de la varianza agrupados o separados. También se calcula el estadístico F y el nivel de probabilidad asociado para la igualdad de varianzas de grupo.

Clase M - Análisis Multivariante

BMD01M: Análisis de Componente Principal.

Calcula las componentes principales de datos estandarizados y ordena cada uno de los casos según el tamaño de cada componente principal por separado. La salida incluye coeficientes de correlación, autovalores y ordenación de cada caso estandarizado.

BMD02M: Regresión sobre Componentes Principales.

Calcula los componentes principales de los datos estandarizados y ordena cada caso de acuerdo con el tamaño de cada componente por separado. Cada variable dependiente se somete a regresión sobre el primero, los dos primeros, los tres primeros, y todos los componentes principales una vez que cada componente está expresado por medio de datos estandarizados.

BMD04M: Análisis Discriminante para Dos Grupos

Calcula una función lineal de p variables medidas sobre cada uno de los individuos de dos grupos. Esta función puede servir como índice para la discriminación entre grupos y se determina de manera que se maximice la dife

rencia entre los índices promedio para los dos grupos, dividida por la desviación típica combinada de los índices. Véase 7M.

BMD05M: Análisis Discriminante para Varios Grupos.

Calcula un conjunto de funciones lineales para clasificar un individuo en uno o varios grupos. La entrada se compone de un conjunto de observaciones dentro de cada uno de los grupos; cada observación es un valor de una de las variables.

El sistema seguido para la asignación de grupo se deriva de un modelo de distribución normal multivariante de observaciones dentro de los grupos, tal que la matriz de covarianza es la misma para todos los grupos. Un individuo queda clasificado dentro del grupo para el cual la densidad de probabilidad estimada es la mayor. El procedimiento de cálculo equivalente evalúa la función lineal correspondiente a cada uno de los grupos y asigna un individuo al grupo para el cual este valor es más alto. Véase 7M.

BMD07M: Análisis discriminante paso a paso.

Realiza un análisis discriminante múltiple. Se calcula un conjunto de funciones de clasificación lineales mediante la elección de las variables independientes según un procedimiento de paso a paso. La variable introducida en cada paso se selecciona mediante uno de entre cuatro criterios disponibles, y una variable se elimina cuando su valor de F se hace demasiado bajo. Utilizando estas funciones y las probabilidades a priori, se determinan las probabilidades a posteriori de que cada caso pertenezca a cada uno de los grupos. El programa calcula también los coeficientes para las variables canónicas y proyecta las dos primeras con el objeto de dar un gráfico bidimensional de la separación entre los grupos.

BMD08M: Análisis factorial

Realiza un análisis factorial de hasta 198 variables. La factorización puede efectuarse a partir de la matriz de covarianzas o de la matriz de correlaciones. Las estimaciones iniciales de las comunalidades pueden ser las correlaciones múltiples al cuadrado, varianzas de la regresión, máximos valores absolutos de filas, o pueden ser especificados por el usuario. Si se desea, el programa realizará iteraciones sobre las estimaciones iniciales de las comunalidades. Son posibles tres tipos de rotación, todas basadas en el criterio oblimin. En la primera los factores son obligados a ser ortogonales, lo que produce, entre otras, rotaciones quartimax y varimax. En la segunda, el criterio se aplica a la estructura factorial, permitiéndose a los factores ser oblicuos, lo que produce rotaciones simples de pesos (ref. 1). Pueden estimarse las coordenadas de los casos en el espacio de los factores (scores). La entrada puede ser en forma de datos originales, matriz de correlaciones, matriz de covarianzas, o matriz factorial (matriz de pesos).

BMD09M: Correlación canónica

Calcula correlaciones canónicas, coeficientes canónicos y variables canónicas, correspondientes a dos conjuntos de variables. Las variables canónicas pueden alternativamente escribirse sobre una cinta de salida para posterior proceso. Los cálculos están basados sobre las matrices de correlaciones o covarianzas, tanto alrededor de la media como del origen.

BMD10M: Identificación de Valores Extremos

Examina datos multivariantes para la identificación de valores extremos calculando la distancia de Mahalanobis de cada caso respecto al centro de distribución de los restantes. Si la probabilidad del estadístico F correspondiente a la distancia mayor es menor que un valor dado, el caso correspondiente queda eliminado.

Clase R - Análisis de Regresión

BMD01R: Regresión lineal simple (Análisis de covarianza de dirección única).

Efectúa un análisis de regresión lineal simple sobre grupos de tratamiento simples o combinados con tamaños de muestra desiguales. (El término "grupos de tratamiento" se emplea aquí para describir categorías). Se calculan las sumas de productos cruzados "dentro de los grupos" y los coeficientes; de esta forma, se proporciona también en los resultados la información del análisis de covarianza.

BMD02R: Regresión por pasos

Calcula una secuencia de ecuaciones de regresión lineal múltiple por pasos. En cada paso se añade una variable a la ecuación de regresión. La variable añadida es la que provoca una mayor reducción en el error suma de cuadrados, la que tiene la mayor correlación parcial con la variable dependiente calculada con las variables que ya han sido añadidas y, además, la que de ser añadida, haría máximo el valor F. Por otra parte, las variables pueden ser incluidas en la ecuación de regresión. Las variables no incluidas se eliminan automáticamente cuando su valor F es demasiado bajo. Pueden seleccionarse también ecuaciones de regresión con o sin ordenada al origen.

BMD03R: Regresión múltiple con combinaciones de casos.

Realiza la regresión múltiple y el análisis de correlación de los datos dentro de cada selección de submuestras pertenecientes a la misma población. Una selección puede ser cualquier conjunto especificado de submuestras.

BMD04R: Regresión periódica y análisis armónico

Realiza un análisis de la regresión periódica o armónica, utilizando una función de la forma:

$$Y_t = a_0 + \sum_{i=1}^n a_i \cos(2\pi it/k) + b_i \sin(2\pi it/k)$$

Las regresiones periódicas se computan para valores sucesivos de n hasta un máximo especificado por el usuario.

BMD05R: Regresión Polinómica

Calcula una regresión polinómica de la forma

$$y = \beta_0 + \beta_1 x + \beta_2 x^2 + \dots + \beta_k x^k + e$$

siendo k entero y positivo.

BMD06R: Regresión asintótica

Efectúa un análisis de regresión asintótica usando una función exponencial modificada de la forma

$$y = \alpha + \beta \rho^x$$

mediante transgeneración (una o varias sucesivas) de la variable dependiente y pueden usarse otras ecuaciones.

BMD07R: Mínimos cuadrados no lineal

Obtiene un ajuste ponderado de mínimos cuadrados

$$y = f(x_1, \dots, x_t; \theta_1, \dots, \theta_p) + e$$

de una función especificada por el usuario para valores x_1, \dots, x_t , por medio de iteraciones de Gauss-Newton sobre los parámetros $\theta_1, \dots, \theta_p$. Dentro de cada iteración se seleccionan los parámetros en una forma paso a paso. El parámetro seleccionado en cada paso es aquél que, diferencialmente al menos, produce la máxima reducción en el error suma de cuadrados. Si es necesario para evitar problemas de singularidad, solamente se modifica un subconjunto de los parámetros para una iteración dada. Además, parámetros con valores límite, cuya modificación en una iteración dada daría lugar a violaciones de algún límite, tampoco se modifican. La iteración se realiza "sobre un límite" cuando el mejor ajuste se presenta fuera del rango del parámetro.

Clase S: Programas Especiales

BMD02S: Análisis de Tablas de Contingencia

Calcula tablas de frecuencia y porcentajes de doble entrada, χ^2 , coeficientes de contingencia y razones de máxima verosimilitud. Cada variable puede ser categorizada mediante diversos procedimientos. Cada uno de estos procedimientos se llama una categorización.

BMD09S: Transgeneración

Efectúa determinadas transgeneraciones en las variables que entran como datos. Se puede efectuar cualquier transgeneración cuyo código figure en la lista de la Introducción. La entrada puede ser por ficha perforada, por cinta código BCD o por cinta código binario.

BMD12S: Transgeneración abierta

Este programa tiene por objeto la transgeneración de datos que más tarde serán procesados por otros programas - BMD. La transgeneración se efectúa por medio de sentencias FORTRAN que proporciona el usuario. Para uso standard sólo se requieren las sentencias aritméticas aunque se puede utilizar todo el lenguaje FORTRAN. Las medias y las desviaciones típicas de las variables originales pueden calcularse previamente si el usuario las necesita para su transgeneración. Las selecciones de caso y variables se especifican más tarde.

Clase T - Análisis de Series Temporales

BMD02T: Autocovarianza y Análisis Espectral de Potencia

Calcula autocovarianza, espectro de potencia, covarianza cruzada, espectro cruzado, función de transferencia y - función de coherencia de series temporales. Se incluyen gráficos de todo lo anterior.

BMD03T: Estimación de espectros de series temporales

Estima autoespectros, espectros cruzados y coherencias

de series temporales estacionarias. Cada serie se descompone en sus componentes de frecuencia mediante una transformación finita de Fourier y los estimadores necesarios se obtienen sumando los productos de las series transformadas. La deriva lineal de cada serie se elimina antes de la transformación. Si se desea, se puede prefiltrar y reducir la escala de la serie antes de la eliminación de deriva. Se pueden obtener matrices de espectros cruzados en forma compleja que más tarde pueden servir de entrada para el programa BMD04T. Es posible tratar un gran número de bandas de frecuencia gracias a la rapidez del algoritmo que se utiliza (ver bibliografía). Siempre se usan espectros dobles lo que hace que se obtengan estimadores que valen la mitad de los que se obtendrían con programas que sólo utilizan espectros simples.

BMD04T: Análisis espectral múltiple de series temporales

A partir de una secuencia de matrices de valores complejos de espectros y espectros cruzados, normalmente generadas por el programa BMD03T, se calculan las coherencias múltiples y las matrices de frecuencia de respuesta correspondientes a un conjunto de series de entrada y salida. Para la ganancia y fase de cada componente de las matrices de frecuencia de respuesta se calculan también los intervalos de confianza. Así mismo se obtienen estimadores de los espectros y espectros cruzados de las series de salida que se generan a partir de parte de las series de entrada (también llamada matriz espectral condicional).

Clase V - Análisis de Varianza

BMD03V: Análisis de la covarianza para diseño factorial.

Este programa realiza un análisis factorial completo de la covarianza.

BMD04V: Análisis de la covarianza con covariantes múltiples

Este programa está diseñado para tratar la información del análisis de la covarianza para el análisis de la varianza de una variable con múltiples covariantes y tratamiento - desigual de tamaños de grupos. Los casos pueden ser especificados por el usuario como en los tratamientos de grupos, o bien se pueden colocar por el programa en el tratamiento de grupos de acuerdo a una expresión Booleana especificada.

BMD05V: Hipótesis lineal general

Realiza los cálculos necesarios para el modelo de hipótesis lineal general. Las variables independientes son de dos tipos generales

- (1) Variables utilizadas para especificar las clasificaciones del análisis de la varianza.
- (2) Variables utilizadas como covariantes.

Utilizando estas variables, el programa sirve para el análisis de la varianza equilibrado o desequilibrado y para el problema de valores ausentes.

BMD06V: Hipótesis lineal general con contrastes

Este programa se parece al BMD05V en que está diseñado para estimar y obtener el significado estadístico de los parámetros que intervienen en el modelo de hipótesis lineal general. Es más general en cuanto que es capaz de obtener el significado estadístico de cualquier función lineal de valor real de los parámetros y es menos general porque no puede hacer test simultáneos de orden superior. Este programa puede hacer el test de la hipótesis $\beta_1 = \beta_2 = 0$, y admite transgeneraciones de los datos de entrada de códigos del 1 al 14.

BMD07V: Tests de rango múltiple

Realiza tests de rango múltiple para dos tipos de datos de entrada:

1) Datos brutos respecto a una variable de clasificación con tamaños de grupos desiguales, 2) tamaños de muestra, medias de grupos, desviaciones típicas y etiquetas de grupos. Si el usuario suministra ficha(s) de rangos, el test se efectúa según esta(s) ficha(s), en otro caso se utilizará el test de rango múltiple de Duncan. El test de rango múltiple de Duncan se utiliza también en los valores de comparación, si el usuario suministra sus diseños de contraste.

BMD08V: Análisis de la varianza

Realiza el análisis de la varianza para cualquier diseño jerárquico con celdas de tamaños iguales. Estos comprenden los diseños incluidos, parcialmente incluidos, parcialmente cruzados y completamente cruzados. El modelo queda especificado al indicar las relaciones de inclusión de los índices (ver 3.d.). Se puede ejecutar un análisis por separado de algunas variables dependientes en forma simultánea. Cada tabla de análisis de la varianza incluye una media cuadrática esperada en términos de los componentes de la varianza. Para estos cálculos se debe especificar cada índice como fijo o aleatorio, tomado de una población finita o infinita.

BMD09V: Análisis de la covarianza

Realiza el análisis de la covarianza utilizando una o más covariantes. Los tamaños de los grupos pueden ser desiguales y se puede realizar un análisis paralelo con varias variables dependientes. El programa comprueba la igualdad de pendiente de grupo en grupo y calcula el estadístico t para contrastes de medias de grupos ajustados, incluyendo en particular, todas las diferencias de pares posibles. El elemento de un grupo puede ser especificado mediante una variable de clasificación o mediante un "caso en blanco", indicador de fin de grupo.

BMD10V: Hipótesis lineal general (n° 2)

Estima los parámetros y comprueba las hipótesis respecto - del modelo lineal general. Las variables ficticias que especifican el análisis de la varianza de la parte del modelo, son generadas por el programa y comprobados de manera automática. Se pueden comprobar hipótesis adicionales expresadas en la forma de restricciones lineales sobre los - coeficientes de regresión. El programa no requiere modelos equilibrados ni modelos de rango total.

BMD12V: Multivariante de la varianza y de la covarianza

Realiza el Modelo I de análisis univariante o multivariante de la varianza o la covarianza para cualquier diseño jerárquico con celdas de tamaños iguales. Esto comprende los diseños incluidos, parcialmente incluidos, parcialmente cruzados y totalmente cruzados. El modelo queda especificado al indicar las relaciones de inclusión de los índices. Se pueden efectuar varios análisis para cada problema, especificando las diferentes variables dependientes o las covariantes en fichas de subproblema separadas. El análisis univariante de la varianza se puede realizar con el programa BMD08V.

III. PREPARACION DE DATOS

Esta Sección explica la forma standard de entrada de datos a los programas BMD. Estos pueden entrar por ficha, cinta o disco.

A. Entrada de datos standard

La forma de entrada de datos standard en los programas BMD es como se indica.

		Variables (columnas)				
		x_1	x_2	x_3	x_p
1		x_{11}	x_{12}	x_{13}	x_{1p}
2		x_{21}	x_{22}	x_{23}	x_{2p}
3	
4	
.						
.						
.						
.						
.						
n		x_{n1}	x_{n2}	x_{n3}	x_{np}

La cabecera representa variables, p.ej: ley, densidad, etc. Cada fila representa un conjunto de valores de estas variables, o sea p.ej. la ley, densidad, etc. de una muestra determinada. Los valores x_{ij} se llaman datos, el conjunto de estos valores numéricos se llama matriz de datos, cada fila es un caso y cada columna una variable.

Los datos se perforan caso a caso, o sea que todos los valores del primer caso se perforan en orden en una o más fichas. El segundo caso debe comenzar en una nueva ficha y así sucesivamente. Cada caso debe entrar con el mismo formato. Esto hace que de un caso al siguiente, cada valor de variable debe ocupar el mismo lugar en cada ficha perforada.

En general, no todos los campos de una ficha son datos; por ejemplo hay campos que sirven para identificación, etc. Los campos - que se quiere que ocupen los datos se seleccionan por medio de - la Ficha de Formato Variable.

Entrada por cinta

Todos los programas BMD están preparados para leer los datos de cinta o disco. El uso de cintas o discos para entrada tiene la - ventaja de que el almacenamiento es muy compacto. Un fichero de datos puede releerse varias veces para realizar varios análisis de los mismos datos, simplemente rebobinando la cinta.

Codificación y perforación

La ficha perforada es un medio importante de comunicación con - las computadoras. La información se registra por medio de peque- ños agujeros rectangulares perforados en determinados lugares de una ficha de tamaño standard. La ficha está constituída por 80 - columnas verticales, cada una de las cuales tiene doce posicio- - nes de perforación. Estas doce posiciones forman doce filas hori- zontales en la ficha. Un carácter se representa por una o más - perforaciones en una columna. Existen dos códigos de perforación: el BCD que consta de 47 caracteres y el EBCDIC que consta de 62.

La codificación de la información (es decir, la asignación de có- dígos numéricos para representar diferentes tratamientos, grupos de edad, etc) no sólo debe atender a que los caracteres usados - sean legales, sino también al tipo de análisis de esos códigos - que se va a efectuar. Por ejemplo, si se desea efectuar una cla- sificación, los códigos deben ser únicamente numéricos. Si un có- digo tiene más de 10 categorías (0, 1, 2, ..., 9) deberán usarse dos columnas para representarlo, y si tiene más de 100, tres. No se recomienda el uso del signo - como 11 y el + como 12 porque - puede dar lugar a complicaciones.

Un conjunto de columnas adyacentes que contienen un item de in- formación, p. ej. la edad, peso, altura, nombre, etc se llama un campo. Si un campo contiene únicamente caracteres numéricos con

signo o sin él, con punto decimal o sin él, se llama campo numérico. Si contiene otros caracteres se llama alfanumérico o alfabético.

Puesto que los expertos en computadoras han desarrollado un lenguaje técnico propio, es necesario aprender algo de este lenguaje para poder entenderse. Este manual incluye algunas definiciones, pero resulta imposible que sea completo. Hay sin embargo numerosos manuales editados por los fabricantes de equipos de tratamiento de datos a los que remite al lector.

IV. PREPARACION DE LAS FICHAS DE CONTROL DE PROGRAMA

Introducción

Los programas BMD están escritos de modo tan general que cada programa tiene posibilidad de manejar numerosas opciones. Es misión del usuario el especificar valores de parámetros, cálculos a efectuar, salidas que desea, forma de entrada de los datos, etc.

En esta sección vamos a describir las fichas de control que utilizan casi todos los programas BMD. No todas estas fichas se necesitan para todos los programas e igualmente hay fichas especiales que son necesarias para determinados programas y que no se describen aquí. El usuario debe consultar la descripción del programa que vaya a utilizar para saber cuáles de estas fichas debe codificar, la codificación y el orden en que deben entrar en el programa. Las fichas de control que son opcionales tienen la referencia entre paréntesis en la sección "Organización de datos". Las demás deben siempre codificarse.

Las fichas de control necesarias para la ejecución de un programa BMD siempre empiezan con una ficha problema y acaban en una ficha final. La ficha problema debe llevar en las columnas 1 a 6 las letras PROBLM y la final, en las mismas columnas FINISH. Entre ambas irán las fichas de formato variable y, opcionalmente, fichas de etiquetas, fichas de transgeneración y las fichas especiales que requiera cada programa en particular.

Ficha Problema: Como se ha dicho anteriormente, la ficha problema tiene en col. 1 a 6 las letras PROBLM. En las restantes columnas lleva el número de fichas de transgeneración, número de fichas de formato variable y la información especial de cada programa. El formato y los parámetros de esta ficha se describen en la documentación de cada programa.

Ficha(s) de Formato Variable de Entrada: Al menos una ficha de formato variable debe figurar en cada programa BMD. Sirve para especificar la disposición de los datos en las fichas o cinta de datos y permite al usuario variar dicha disposición de un -

proyecto a otro. El formato debe encontrarse siempre entre paréntesis. Luego se describirá esta ficha con más detalle.

Ficha(s) de Formato Variable de Salida: Algunos programas permiten al usuario especificar la disposición de los resultados que vayan a salir en impresora, perforadora de fichas o en cinta. Las diferencias entre los formatos de salida y de entrada se describirán más adelante.

Fichas de Etiqueta: El usuario puede, mediante las fichas de etiqueta, asociar nombres alfanuméricos a los números de variables. Cada ficha de etiqueta debe llevar las letras LABELS en las columnas 1 a 6. A continuación pueden figurar hasta 7 números de variables con su etiqueta asociada correspondiente. Las fichas de etiquetas sólo se pueden usar en los programas en cuya descripción se cita expresamente esta posibilidad.

Fichas de Transgeneración: Mediante estas fichas se pueden crear nuevas variables por la aplicación de transformaciones algebraicas, trigonométricas, exponenciales y logarítmicas a las variables originales. Estas nuevas variables pueden reemplazar a las antiguas o añadirse a las mismas con nuevos números. La discusión completa del concepto de transgeneración se hace más adelante.

Nota: A no ser que se diga lo contrario, los campos numéricos de las fichas de control deben perforarse sin punto decimal. Este se supone que se encuentra en la posición extrema derecha del campo. Las columnas en blanco se toman como ceros. En general, los números de la ficha de control deben situarse de modo que terminen en la posición extrema derecha del campo correspondiente. Esto se llama ajuste a la derecha.

A. Fichas de Etiqueta

Permite sustituir los índices numéricos de las variables que aparecen en la salida, por nombres alfanuméricos.

Preparación de la ficha

Col	1- 6	LABELS (obligatorio)
	7-10	Número de la variable que se va a etiquetar (ajustado a la derecha)
	11-16	Etiqueta alfanumérica correspondiente
	17-20	Número de otra variable
	21-26	Etiqueta alfanumérica
	⋮	
	67-70	Número de otra variable
	71-76	Etiqueta correspondiente (hasta 7 por ficha)

Aunque en cada ficha caben hasta 7 pares de números y nombres, puede especificarse sólo un par por ficha. No obstante, en la ficha Problema o Subproblema debe especificarse un número de etiquetas igual a las que aparecen en las fichas de Etiquetas. No es necesario etiquetar todas las variables.

B. Fichas de Transgeneración

El término "transgeneración" se usa para indicar la transformación de variables de entrada y la creación de nuevas variables antes de que empiecen los cálculos de algunos programas.

Muchos programas BMD incluyen la opción de transgeneración y la lista de transgeneraciones que pueden hacerse con esta opción figura en las "Limitaciones" que se incluyen en las descripciones de programas. Para los programas que no tienen esta opción, puede prepararse una transgeneración previa de los datos mediante otros programas BMD como por ejemplo el BMD09S.

A continuación vamos a poner un ejemplo en el cual el símbolo x_i se utiliza tanto para nombrar a la variable i -ésima como al valor de la misma. En él describimos una serie de sucesivas transformaciones para reemplazar la variable x_5 por el valor -

$$\sqrt{x_1^2 + x_3^2}.$$

TransformaciónVARIABLES ALMACENADAS EN CADA PASO

	x_1	x_2	x_3	x_4	x_5
$x_1^2 \rightarrow x_1$	x_1^2	x_2	x_3	x_4	x_5
$x_3^2 \rightarrow x_3$	x_1^2	x_2	x_3^2	x_4	x_5
$x_1 + x_3 \rightarrow x_5$	x_1^2	x_2	x_3^2	x_4	$x_1^2 + x_3^2$
$x_5 \rightarrow x_5$	x_1^2	x_2	x_3^2	x_4	$\sqrt{x_1^2 + x_3^2}$

En este ejemplo se ve que el valor original de x_5 no tiene importancia. Incluso en este momento, la variable x_5 puede ser una variable ficticia que únicamente se utilice para sucesivas transformaciones.

LISTA DE TRANSGENERACIONES

Notaciones

i, j, k : índices de variables (no necesariamente distintos)

c : constante

a_1, a_2, a_3 : constantes

n : número de casos o tamaño de la muestra

$$\text{media } \bar{x}_i = \frac{1}{n} \sum_{j=1}^n k_{ji}$$

$$\text{desviación típica: } \frac{1}{n-1} \sum_{j=1}^n (x_{ji} - \bar{x}_i)^2 \quad 1/2$$

<u>Código</u>	<u>Transgeneración</u>	<u>Restricción</u>
01	$\sqrt{x}_i \rightarrow x_k$	$x_i \geq 0$
02	$\sqrt{x}_i + \sqrt{x_i+1} \rightarrow x_k$	$x_i \geq 0$
03	$\log_{10} x_i \rightarrow x_k$	$x_i > 0$
04	$e^{x_i} \rightarrow x_k$	-
05	$\arcsen \sqrt{x}_i \rightarrow x_k$	$0 \leq x_i \leq 1$
06	$\arcsen \sqrt{x_i/(n+1)} + \arcsen \sqrt{(x_i+1)/(n+1)} \rightarrow x_k$	$0 \leq (x_i/n) \leq 1$
07	$1/x_i \rightarrow x_k$	$x_i \neq 0$
08	$x_i + c \rightarrow x_k$	-
09	$x_i^c \rightarrow x_k$	-
10	$x_i^c \rightarrow x_k$	$x_i \geq 0$
11	$x_i + x_j \rightarrow x_k$	-
12	$x_i - x_j \rightarrow x_k$	-
13	$x_i x_j \rightarrow x_k$	-

<u>Código</u>	<u>Transgeneración</u>	<u>Restricción</u>
14	$x_i/x_j \rightarrow x_k$	$x_j \neq 0$
15	si $x_i > c$, $1 \rightarrow x_k$; en caso contrario $0 \rightarrow x_k$;	-
16	si $x_i > x_j$, $1 \rightarrow x_k$; en caso contrario, $0 \rightarrow x_k$	-
17	$\log_e x_i \rightarrow x_k$	$x_i > 0$
18	$x_i - \bar{x}_i \rightarrow x_k$	-
19	$x_i/s_i \rightarrow x_k$	-
20	$\text{sen } x_i \rightarrow x_k$	-
21	$\text{cos } x_i \rightarrow x_k$	-
22	$\text{arctan } x_i \rightarrow x_k$	-
23	$x_i^{x_j} \rightarrow x_k$	$x_i > 0$
24	$c^{x_i} \rightarrow x_k$	$c > 0$
25	$x_i \rightarrow x_k$	-
26	$c \rightarrow x_k$	-
27-39	No se usan	
40	Si $x_i = a_1$ o a_2 o a_3, \dots, a_7 , entonces $c \rightarrow x_k$; en los demás casos x_k no cambia	
41	Si x_i es blanco entonces $c \rightarrow x_k$; en los demás casos x_k no cambia	$x_i \neq -0$

<u>Código</u>	<u>Transgeneración</u>	<u>Restricción</u>
42	si $x_i = a_1$ o a_2 o a_3, \dots o a_7 $x_j \rightarrow x_k$ en los demás casos x_k no cambia	
43	si x_i es blanco, $x_j \rightarrow x_k$ si no, x_k no cambia	$x_i \neq 0$

Cuando existe una violación de las restricciones que figuran en el cuadro anterior durante la transgeneración, el programa imprimirá un mensaje de diagnóstico. Unos programas saltarán a leer el siguiente problema si existe, otros no tendrán en cuenta la transgeneración en la que ha ocurrido el error y continuarán con los cálculos y finalmente, unos terceros continuarán leyendo los datos de entrada, por si hubiera otro error, antes de saltar al siguiente problema si existe.

1. Fichas de Transgeneración Standard

Se usan con los programas que utilizan la entrada de datos standard. Sea p el n° de variables de la matriz de datos y m el máximo número de variables que puede manejar el programa. Se pueden usar para transgeneración cualquiera de las variables $x_1 \dots x_m$. Los valores iniciales de las primeras p variables se leen del fichero de datos de entrada (en ficha o en cinta). Los valores iniciales de las restantes $m-p$ variables no hacen falta para los cálculos previos. Después de la transgeneración de un caso particular, los valores de las $p+q$ primeras variables para este caso se usan como valores de variables transgeneradas. Si las $p+q$ variables necesarias para el cálculo no son las primeras $p+q$, deben trasladarse a las $p+q$ primeras. Esto puede suceder cuando se usa la transgeneración de código 25. Los números p y q (q puede ser mayor, menor o igual a 0) se especifican en la ficha Problema. Los índices i, j y k de la lista de transgeneración, pueden ser mayores que p y que $p+q$ pero nunca mayores que m .

Preparación de la ficha

Col 1- 6 TRNGEN (obligatorio)
 7- 9 Índice de variable k
 10-11 Código de Transgeneración
 12-14 Índice de variable i
 15-20 Índice de variable j o constante C
 21-25 Blancos
 26 Número de constantes a_i para el código 40 ó 42
 27-32 Valor a_1
 33-38 Valor a_2

 63-68 Valor a_7

Las constantes c , a_1 , a_2 ... a_7 se perforarán con punto decimal si se usan con variables de tipo de formato F y sin él si el tipo de formato es I.

Las fichas de transgeneración para el ejemplo que se ha puesto antes serían:

```
TRNGEN001100012.0000
TRNGEN003100032.0000
TRNGEN00511001000003
TRNGEN00501005000000
```

2. Ficha especial de Transgeneración

Se usa solamente en programas que requieren transformaciones de la forma $f(y) \rightarrow Y$. La ficha especial puede describir transformaciones sucesivas.

Preparación de la ficha

Col 1- 6 SPECTG (obligatorio)
 7 Número de transformaciones (≤ 8 por ficha)
 8- 9 Código* para la 1^a transformación
 10-15 Constante** para la 1^a transformación (si no hay, dejar en blanco)

Col 16-17 Código* para la 2^a transformación

⋮

64-65 Código* para la 8^a transformación

66-71 Constante** para la 8^a transformación

Por ejemplo, si el usuario desea efectuar la transformación -

$$(Y^2 - 3 + 100)^2 \rightarrow Y$$

deberá escribir la siguiente ficha especial

SPECTG5102.000008-3.0000100000008100.00102.0000

C. Fichas de Formato Variable de Entrada

La palabra "formato" significa la disposición de la información perforada en una ficha. El formato de una ficha de datos es una secuencia de campos (variables) cada uno de los cuales puede ocupar una o varias columnas. Para los programas de ordenador un formato es un conjunto de especificaciones, de acuerdo con las cuales el programa lee la información de las fichas perforadas. Estas especificaciones indican el programa qué partes (o columnas) de la ficha debe saltar, cuáles debe considerar como un solo número y cuáles como varios en una sola fila. Por ejemplo, el formato puede indicar el programa que una ficha que tiene perforadas las cifras "345890021" se debe leer como "34.5, 890.0, .021" o bien "34., 9002.1" ó "3, 4, 58, 90, 0, 2, 1" etc

También sirve el formato para indicar al programa cómo debe leer un cierto conjunto de fichas cuando hace falta más de una para contener todos los datos de un caso. Cada caso sucesivo se supone que tiene el mismo formato.

* El código debe ser uno de los que se especifica como utilizables en la descripción particular del programa

** Perforar el punto decimal. No es preciso perforarlo si la constante es un entero ajustado a la derecha.

Además de proveer un método económico de preparación de datos de entrada por ficha (definiendo campos lo más pequeños posible o empaquetando los datos) el formato variable permite considerable libertad para controlar la entrada de datos.

Por ejemplo, permite al usuario:

- Seleccionar para cada caso solamente aquellas tarjetas que tienen campos que interesen.
- Seleccionar estos campos de interés, entre todos los que figuran en una ficha.
- Escalar los datos de entrada (es decir, correr el punto decimal).

La descripción completa de los formatos puede encontrarse en los manuales de programación FORTRAN. A continuación se describen los formatos que normalmente hacen falta para los programas BMD.

D. Fichas de Formato Variable de Salida

Algunos programas BMD requieren el uso de las fichas de formato variable para describir la salida de resultados, sea por impresora, por ficha perforada o por cinta. La función de la ficha de formato es la misma para entrada que para salida, es decir, es una descripción de los datos en un medio externo al ordenador.

Los formatos de entrada y salida son idénticos a excepción de las pequeñas diferencias que se describen a continuación.

- Los formatos de tipo F para salida siempre tienen el punto decimal y por tanto hay que reservar un sitio para él.
- Si se usa la especificación de factor de escala de la forma "sPnFw.d" la representación externa del número es 10^s veces la interna. O sea que si el número interno es -15.9357 la especificación 2PF9.1 hará que en el medio externo que sea (impresora, ficha o cinta) el número quede representado por "~~15~~-1593.6".

- Cada línea de entrada o salida en fichas perforadas no puede exceder de 80 caracteres, y si es en impresora o cinta, de 132.
- La posición 1 de la línea de impresión se usa para control del carro de la impresora, es decir, para que empiece página (si en esa posición hay un 1), para que escriba a doble espacio (si hay un 0), para que escriba a espacio simple (si hay un blanco) o para que no salte (si hay un +). Este carácter no ocupa sitio en la línea de impresión, o sea que se puede describir otras 132 posiciones además de la 1^a.

1. Formato variable tipo F

Es el usado con más frecuencia por los programas BMD. Es necesario siempre que el punto decimal esté perforado o que deba ser colocado por el programa. Todos los datos que entren con este formato deben ser números con o sin signo y con o sin punto decimal.

Especificaciones

(a) "nFw.d" F es el indicador de punto flotante (decimal).

n es el número de campos de w posiciones (incluyendo signo y punto decimal si está perforado), y d el número de dígitos a la derecha del punto decimal si éste no está perforado ($0 \leq d \leq w$). Si el punto decimal está perforado d no se tiene en cuenta. Si no se especifica n se supone que vale 1.

(b) "sPnFw.d" P es el indicador de escala.

s es el factor de escala (que se ha explicado antes); n, w y d se han explicado en (a).

(c) "mX" X es el indicador de salto y m el número de columnas que deben saltarse.

(d) "/" indica "saltar una ficha de datos"

"//" indica "saltar dos fichas de datos"

El formato se perfora empezando por un paréntesis izquierdo se guido por la secuencia de especificaciones y cerrado por un pa réntesis derecho. Las especificaciones (a), (b) y (c) deben ir seguidas por una coma, excepto cuando preceden a un "/" o a un paréntesis derecho. Las columnas que figuren en blanco, se ignoran.

Ejemplo: (5x, 2F6.0, F1.0, 3x, F5.0/5x,F6.0)

- (1) Salta cinco columnas
- (2) Lee dos campos de 6 dígitos (col 6-11 y 12-17)
- (3) Lee un campo de un dígito (col 18)
- (4) Salta tres columnas
- (5) Lee un campo de cinco dígitos (col 22-26)
- (6) Salta a la siguiente ficha
- (7) Salta 5 columnas
- (8) Lee un campo de 6 dígitos (col 6 a 11) de la segunda ficha
- (9) Repite lo anterior para cada dos fichas.

Ejemplo de Salida

(1H0,13F10.4/1x,14F9.4)

- (1) Salta dos líneas antes de imprimir
- (2) Escribe 13 números en campos de 10 posiciones de las cuales las cuatro últimas son decimales y la anterior está ocupada por el punto decimal.
- (3) Salta a la siguiente línea
- (4) Escribe 14 campos de 9 dígitos con punto decimal y cuatro dígitos decimales.

Especificación "sPnFw.d"

Ya se explicó anteriormente esta especificación. Pondremos simplemente un ejemplo.

2PF6.0 especificará que el n° leído haya que dividirlo por 10^2 .
-2PF6.0 que hay que dividirlo por 0.01.

es decir: Si hay punto decimal perforado

número almacenado = número leído/ 10^s $-8 < s < 8$

Si no hay punto decimal en el campo

número almacenado = número leído/ 10^{s+d}

Hay que tener en cuenta que en un formato que describa varios campos, si uno de ellos tiene factor de escala, se supone que los siguientes tienen el mismo si no se dice lo contrario. Por ejemplo

(F4.2, -6PF3.0, F4.0)

El programa interpretará que el último campo es -6PF4.0.

2. Formato Variable tipo I

Sirve para procesar únicamente valores enteros. La especificación es "nIw" siendo w el tamaño del campo y n el número de campos (se supone 1 si no se pone n).

Todos los datos deben ser números con o sin signo y sin punto decimal. Por ejemplo

(12I3, I4, 11I2)

3. Formato Variable tipo A

Este formato se usa en los programas que pueden procesar datos con caracteres alfabéticos, numéricos, especiales o cualquier combinación de los anteriores. La especificación es "nAw" siendo w el tamaño del campo, $1 \leq w \leq 4$ y n el número de campos. Cada especificación de campo ocupa en el ordenador una palabra que consiste en cuatro caracteres. Si $w < 4$ los caracteres se posicionan a la izquierda del campo rellenando los restantes espacios con blancos (Ø). Por ejemplo

Dato: AGE	Formato: A4	Se almacena: AGEØ
\$	A1	\$ØØØ

V. PREPARACION DE LAS FICHAS DE CONTROL DEL SISTEMA

La utilización de una computadora en la resolución de problemas requiere, aparte de la confección de los programas, una serie de instrucciones que hagan conocer a la computadora lo que el usuario desea hacer con las fichas que le suministra; estas instrucciones constituyen el lenguaje de control del Sistema. Cada computadora, e incluso dentro de la misma computadora, cada Sistema Operativo, utiliza un lenguaje que normalmente es completamente distinto del de los demás. Como es inútil intentar describir todos ellos, el lenguaje de control y la confección de las fichas correspondientes que vamos a comentar a continuación se referirá a las características físicas y técnicas siguientes

Sistema IBM 370

Sistema Operativo DOS/VS (Disk Operating System/Virtual Storage)

Concepción del Sistema BMD: El conjunto de programas que constituyen el Sistema BMD, aunque todos útiles, se diferencian bastante en su frecuencia de uso, al menos desde nuestro punto de vista; por tanto no se ha considerado necesario su almacenamiento permanente en la Biblioteca de Programas de una computadora. En consecuencia, cada vez que sea necesario utilizar un programa, será preciso compilarlo y linkeditarlo (es decir, traducir el lenguaje del programa a un lenguaje inteligible por la computadora).

Las fichas necesarias para estas operaciones y su ejecución son las siguientes:

1° Ficha JOB

//~~JOBB~~ comentario (∅ = blanco)

Comienza en la columna 1 y el comentario dependerá de la costumbre de los usuarios del Ordenador en que se efectúe el trabajo o de sus programas de contabilidad.

2° Ficha OPTION

//~~OPTION~~LINK,NOLIST

Comienza en la columna 1 e indica que se desea la compilación y linkedición del programa sin que salga listado del mismo - por la impresora.

3° Ficha de llamada al Compilador FORTRAN

```
//EXEC FFORTRAN
```

También comienza en col. 1

4° Ficha de previsión por si el programa contuviera algún carácter BCD. En columna 1.

```
//FTCBCD
```

5° Fichas del programa BMD

6° Fichas de las subrutinas que dicho programa utiliza.

7° Fin de Fichero; en col. 1 y 2.

```
/*
```

8° Dado que los programas BMD utilizan como dirección lógica de la lectora de fichas la 5 y de la impresora la 6 y no se ha querido alterarlas, y como el IBM 370 usa con este fin las direcciones lógicas 2 y 3 respectivamente, es necesario incluir la ficha siguiente (que como excepción empieza en cualquier columna excepto en la 1):

```
INCLUDEILFGHTAB
```

9° Orden de linkeditar. En col. 1.

```
//EXECLNKEDT seguida de
```

```
/*
```

A continuación se encuentran ya las fichas necesarias para - ejecución propiamente dicha.

Si el programa utiliza alguna unidad de cinta, deberán ponerse tantas fichas de asignación como unidades de cinta vayan a usarse. La ficha, comenzando en col. 1 deberá escribirse

```
//ASSGNSYSnnn,X'UUU' en el que:
```

nnn = es un número de tres cifras que se obtendrá restando a la dirección lógica de la cinta tres unidades; es decir, que si la dirección de unidad de cinta es 9, quedará -SYS006.

UUU = es la unidad física en donde vaya a montarse la cinta magnética y cuyo número dependerá de la instalación que se utilice.

Detrás de la(s) ficha(s) ASSGN se colocará la orden de ejecución.

//EXEC

luego las fichas de datos, y detrás las fichas de fin de fichero y fin de JOB

/*

/&

Organización de los programas BMD abiertos (open-ended)

En algunos programas BMD, se da ocasión al usuario de que intervenga a su conveniencia en la redacción de los mismos. Aunque esta intervención requiere un conocimiento al menos elemental del lenguaje FORTRAN, describiremos someramente las fichas que deben siempre figurar.

La intervención del usuario debe constituir una subrutina completa. Las fichas de esta subrutina deberán figurar junto con las del programa y las demás subrutinas antes del //EXEC LNKEDT. Esta subrutina comenzará siempre con las sentencias:

SUBROUTINE Nombre (parámetro, parámetro, etc)

y acabarán con la sentencia

RETURN

todas ellas empezando en col. 7.

Entre la sentencia SUBROUTINE y la RETURN figurarán las que - haya escrito el usuario.

Nombre: es el nombre de la subrutina, y los parámetros aquellas variables que sirven para intercambiar información entre el programa principal y la subrutina.

No nos extendemos en mayores explicaciones, puesto que, como hemos dicho antes, la utilización de programas BMD abiertos requiere algunos conocimientos de FORTRAN y, por tanto, es inútil repetirlos aquí.

B I B L I O G R A F I A

Textos Generales:

- Chacón, E, Curso de Estadística, Publicaciones de la Universidad de Deusto, 3 vols, 1955.
- Dixon, W.J. and Massey, F.J. - Introduction to Statistical Analysis, McGraw-Hill, 1969.
- Mood, A.M. and Graybill, F - Introducción a la Teoría de la Estadística, Aguilar, 1969.
- Ríos, S Métodos Estadísticos, Ediciones del Castillo, 1967.

Análisis Multivariante:

- Anderson, T.W. An Introduction to Multivariate Statistical Analysis, Wiley, 1958.
- Cooley, W.W. and Lohnes, P.R. - Multivariate Procedures for the Behavioral Sciences, Wiley, 1962.
- Morrison, D.F. Multivariate Statistical Methods, McGraw-Hill, 1967.
- Rao, C.R. Linear Statistical Inference and Its Applications, Wiley, 1965

Aplicación de Computadoras:

- Afifi, A.A. and Azen, S.P. - Statistical Analysis - A computer Oriented Approach, Academic Press, 1972.
- Enslein, K, Ralston A, and Wilf, H - Statistical methods for digital computers, Vol. 3 of Mathematical Methods for Digital Computers, John Wiley, 1977

Aplicación a las Ciencias de la Tierra:

Davis, J.C Statistics and Data Analysis in Geology,
John Wiley & Sons, 1973.

Krumbein, W.C. and Grayvill, F.A. - An Introduction to Statis-
tical Models ins Geology, McGraw-Hill, 1965.

Koch, G.S. and Link, R.F - Statistical Analysis of Geological
Data, John Wiley & Sons, 2 vols, 1971.

Miller, R.L. and Kahn, J.S. - Statistical Analysis in the Geo-
logical Sciences, John Wiley & Sons, 1962.

VII. APENDICE. DESCRIPCION DEL MODULO BMDASM

Este módulo está escrito en ENSAMBLADOR IBM/370, y ha sido probado bajo el sistema operativo DOS/VS, siendo totalmente compatible con el sistema operativo DOS/VS, por utilizar instrucciones de ensamblador comunes.

Deberá ensamblarse BMDASM conjuntamente con la compilación de cada programa BMD, que utilice alguna de las subrutinas o funciones resueltas por BMDASM; o catalogar este módulo en alguna biblioteca rubicable, e incluirlo al linkeditar alguno de los programas BMD que utilice alguna subrutina o función resuelta por BMDASM.

- Ejemplo para DOS/VS

Compilación conjunta de un programa BMD y BMDASM	Suponiendo que BMDASM está catalogado en la RL
<pre>// JOB ... // OPTION LINK // EXEC FFORTRAN : Programa principal : Y : subrutinas BMDXXX /* // EXEC ASSEMBLY : Módulo : BMDASM : /* // EXEC LNKEDT // EXEC : Fichas de datos : del programa BMDXXX /* /&</pre>	<pre>// JOB // OPTION LINK // EXEC FFORTRAN : Programa principal : Y : subrutinas BMDXXX /* INCLUDE BMDASM // EXEC LNKEDT // EXEC LNKEDT // EXEC : : Ficha de datos del : programa BMDXXX /* /&</pre>

MODULO BMDASM

El Módulo BMDASM, contiene una serie de subrutinas y funciones utilizadas por los programas:

BMD02R, BMD07V, BMD08V, BMD12V, BMD08M, BMD07R v BMD14S.

Subrutinas y Funciones que incluye este Módulo:

- GETCHR (1)
- PUTCHR (1)
- LAND (2)
- LOR (2)
- LCOMPL (2)
- AND (2)
- OR (2)
- COMPL (2)
- DSHIFT (2)
- SHIFT (2)
- LSHIFT (2)

(1 = SUBROUTINA, 2 = FUNCION)

Descripción:

GETCHR: (obtener carácter de tabla de caracteres)

CALL GETCHR(A,N,CH)

A Variable entera o real, o Matriz de cualquier dimensión,
entera o real.

N Variable o constante INTEGER*2

CH ... Variable REAL*4 o INTEGER*4

GETCHR, mueve el carácter N-SIMO de A, al byte más significativo
de la variable CH borrando los otros tres bytes a blan--
cos (exadecimal 40)

PUTCHR: (colocar carácter en tabla de caracteres)

CALL PUTCHR (A,N,C)

A Variable entera o real, o Matriz de cualquier dimensión,
entera o real.

N Variable o constante INTEGER*4

C Variable REAL*4 ó INTEGER*4

PUTCHR, Mueve el carácter que ocupa el byte más significativo de
la variable C, en la posición N-SIMA de A.

Los otros tres bytes de la variable C se ignoran.

Sólo modifica 1 byte de A.

AND y LAND:

Funciones lógicas AND, para aplicar a variables reales y enteras respectivamente.

```
..... AND (A,B) .....
..... LAND (I,J) .....
```

Al utilizar alguna de estas expresiones se sustituirá por el valor calculado.

A y B variables REAL*4

I y J variables INTEGER*4

El valor calculado es el que resulta de aplicar la operación lógica AND entre los 32 bits de A o (I) y los 32 bits de B o (J).

COMPL y LCOMPL (funciones lógicas)

Calculan el complemento a uno del valor en binario de una variable real o entera respectivamente.

```
..... COMPL (A) .....
..... LCOMPL (I) .....
```

El valor calculado es el que resulta de sustituir todos los bits 1 por 0 y viceversa en los 32 bits de la variable A o I.

OR y LOR

Funciones lógicas OR, para aplicar a variables reales y enteras respectivamente.

..... OR (A,B)

..... LOR (I,J)

Al utilizar alguna de estas expresiones, se sustituye por el valor calculado

A y B variables REAL*4

I y J variables INTEGER*4

El valor calculado es el que resulta de aplicar la operación lógica OR entre los 32 bytes de A o I y los 32 bytes de B o J.

SHIFT, LSHIFT, DSHIFT

Funciones lógicas para desplazar a derecha o izquierda un número determinado de bits, una variable REAL*4, INTEGER*4 y REAL*8 respectivamente.

..... SHIFT(A,K)

..... LSHIFT(I,K)

..... DSHIFT(D,K)

A variable REAL*4

I y K . variables INTEGER*4

D variable REAL*8 (doble precisión)

El valor calculado es el que resulta de desplazar el primer parámetro a la DERECHA o IZQUIERDA, según que el segundo sea NEGATIVO o POSITIVO, tantos bits, como el valor absoluto del segundo parámetro.

EXTERNAL SYMBOL DICTIONARY

PAGE 1

SYMBOL	TYPE	ID	ADDR	LENGTH	LC-ID
BMDASM	SD (GSECT)	C1	CCCCC0	CO013F	
PJCHR	LD (ENTRY)		000000		01
GECHR	LD (ENTRY)		000022		01
BOJLEA	LD (ENTRY)		00004A		01
LAND	LD (ENTRY)		00004A		01
AND	LD (ENTRY)		00004A		01
LCR	LD (ENTRY)		000076		01
OR	LD (ENTRY)		000076		01
LCOMPL	LD (ENTRY)		0000A2		01
COMPL	LD (ENTRY)		0000A2		01
DSHIFT	LD (ENTRY)		0000CA		01
LSHIFT	LD (ENTRY)		000112		01
SHIFT	LD (ENTRY)		000112		01

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	DOSKVS ASSEMBLER REL 32.0 19.52 78-03-20
				2 *	BMDASM	
				3 *	ESCRITO EN OCT-1977. CENTRO DE CALCULO E.T.S.I. DE MINAS DE MADRID.	
				4 *	EN ASSEMBLER IBM-DOSKVS, REL 32.0.	
				5 *	ESTE MODULO CONTIENE SUBROUTINAS UTILIZADAS POR	
				6 *	BMD02R, BMD07V, BMD0EV, BMD12V, BMD06M, BMD01R Y BMD14S.	
				7 *	ENSAMBLAR ESTE MODULO E INCLUIRLO CUANDO SE LINKEDITE	
				8 *	ALGUNO DE LOS ANTERIORES PROGRAMAS.	
				9 *		
CCCCC				10	BMDASM CSECT	
				11	ENTRY GETCHR, PUTCHR, ECLLEAN, LAND, LCR, LCCMPL, AND, CR, COMPL	
				12	ENTRY DSHIFT, SHIFT, LSHIFT	

Zubiri Zabala

Zubiri Zabala

Zabala

LCC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT
				0004A	55 EQU	LEAVE ECU *
				0004A	56 DCCL	ECU * 'L' AND 'Y' AND 'A' LOGIC FUNCTION AND DATA VALUES
				0004A	57 LAND	ECU * ENTIRE AND REAL RESPECTIVELY.
				0004A	58 AND	ECU * AND(A,B) AND LAND(I,J)
				0004A	59	USING *,15
CCCC4A	50EC D00C			0000C	60 STM	14,12,12(13) SALVA REGISTROS GENERALES
CC004E	5821 0000			00000	61 L	2,0(1) CARCA DIRECCION DEL PRIMER PARAMETRO
CC0052	5822 C000			00000	62 L	2,0(2) CARCA CONTENIDO DEL PRIMER PARAMETRO
CC0056	5831 0004			00004	63 L	3,4(1) CARCA DIRECCION DEL SEGUNDO PARAMETRO
CC005A	5423 0000			00000	64 N	2,0(3) AND LOGIC
CC005E	1802			00065	65 LR	0,2 CARCA RESULTADO EN EL REG. O PUNTO FIJO
CC0060	5020 F00E		0012E	66 ST	2,TRCCL	ALMACENA EL RESULTADO
CC0064	7800 F00E		0012E	67 LE	0,TRCCL	CARCA RESULTADO EN EL REG. C PUNTO FLCT.
CC006E	98EF D00C		0000C	68 LM	14,15,12(13)	RESTAURA REGISTROS GENERALES 14-15
00006C	92FF D00C	0000C		69 MVI	12(13),X'FF'	RETURN CORRECTO
CC0070	981C D018		00018	70 LM	1,12,24(13)	RESTAURA REGISTROS GENERALES 1-12
CC0074	07FE			71 BR	14	RETURN A QUIEN LLAMA.
				73 *		'L' AND 'R' LOGIC FUNCTION FOR VALUES
				74 *		ENTIRE AND REAL RESPECTIVELY.
				00076	75 LCR	ECU * CR(A,B) AND LCR(I,J)
				CC076	76 CR	ECU *
				CC076	77	USING *,15
000074	50EC D00C			0000C	78 STM	14,12,12(13) SALVA REGISTROS GENERALES
CC007A	5821 0000			00000	79 L	2,0(1) CARCA DIRECCION DEL PRIMER PARAMETRO
CC007E	5822 C000			00000	80 L	2,0(2) CARCA CONTENIDO DEL PRIMER PARAMETRO
CC0082	5831 0004			00004	81 L	3,4(1) CARCA DIRECCION DEL SEGUNDO PARAMETRO
CC0086	5823 0000			00000	82 O	2,0(3) CR LOGIC
CC008A	1802			00083	83 LR	0,2 CARCA RESULTADO EN EL REG. O PUNTO FIJO
CC008C	5020 F012		0012E	84 ST	2,TRCCL	ALMACENA EL RESULTADO
CC0090	7800 F082		0012E	85 LE	0,TRCCL	CARCA RESULTADO EN EL REG. C PUNTO FLCT.
CC0094	98EF D00C		0000C	86 LM	14,15,12(13)	RESTAURA REGISTROS GENERALES 14-15
CC0098	92FF D00C	0000C		87 MVI	12(13),X'FF'	RETURN CORRECTO
CC009C	981C D018		00018	88 LM	1,12,24(13)	RESTAURA REGISTROS GENERALES 1-12
CC00A0	07FE			89 BR	14	RETURN A QUIEN LLAMA.

Zubini Paramo

Zubini Paramo

LOC OBJECT CODE ADDR1 ADDR2 STMT SOURCE STATEMENT DOSRVS ASSEMBLER REL 32.0 19.52 78-03-30

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	DOSRVS ASSEMBLER REL 32.0 19.52 78-03-30
				91 *	'LCCMPL' Y	'COMPL' CAMEIA LOS 32 BITS DE UNA
				92 *		VARIABLE - CEROS POR UNOS Y
	000A2			93 L	COMPL EQU *	UNOS POR CEROS.
	000A2			94 C	COMPL EQU *	VARIABLE ENTERA- LCCMPL(I)
				95 *		VARIABLE REAL - COMPL(A)
	000A2			96	USING *,15	
0000A2	90EC	D00C		0000C	97	STM 14,12,12(13) SALVA REGISTROS GENERALES
0000A6	5821	0000		00000	98	L 2,0(1) CARGA DIRECCION DEL PRIMER PARAMETRO
0000AA	5822	0000		00000	99	L 2,0(2) CARGA CONTENIDO DEL PRIMER PARAMETRO
0000AE	5720	F0EE		00130	100	X 2,XFF CR EXCLUSIVO LOGICO CON VARIABLE X'FF----
0000B2	1802			101	LR 0,2	CARGA RESULTADO EN EL REG. 0 PUNTO FIJO
0000B4	502C	F0EE		00128	102	ST 2,1000 ALMACENA EL RESULTADO
0000B8	730C	F0EE		0012E	103	LE 0,1000 CAMBIA RESULTADO EN EL REG. 0 PUNTO FIJO.
0000BC	730E	0000		00000	104	LM 14,15,12(13)
0000C0	521F	0000		0000C	105	MVI 12(13),X'FF' RETURN CORRECTO
0000C4	9810	D018		0001E	106	LM 1,12,24(13) RESTAURA REGISTROS GENERALES 1 - 12
0000C8	07FE			107	BF 14	RETURN A QUIEN LLAME

Zubiri Pineda

Zubiri Pineda

Zubiri Pineda

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	COSRVS ASSEMBLER REL 32. C 15.52 78-03-30	
				109 *	FUNCIONES CSPIFT SHIFT LSHIFT		
				110 *	CALCULAN EL VALOR DE DESPLAZAR EL PRIMER PARAMETRO		
				111 *	A LA DERECHA O A LA IZQUIERDA, SEGUN QUE EL SEGUNDO		
				112 *	PARAMETRO SEA NEGATIVO O POSITIVO, TANTOS BITS COMO		
				113 *	EL VALOR QUE CONTIENE LA VARIABLE ENTERA DEL SEGUNDO		
				114 *	PARAMETRO.		
				115 *	DSHIFT(C,J) DOBLE PRECISION		
				116 *	SHIFT(A,K) SIMPLE PRECISION		
				117 *	LSHIFT(I,M) VARIABLE ENTERA		
		000CA		118 DSHIFT	ECU		
		CCCCA		119	USING *,15		
0000CA	90EC	000C		120	STM 14,12,12(13)	SALVA REGISTROS GENERALES	
CCCCCE	5821	0000		121	L 2,0(1)	CARGA DIRECCION DEL PRIMER PARAMETRO	
0000E2	6002	0000		122	LD 0,0(2)	CARGA VALOR DEL PRIMER PARAMETRO	
0000D6	600C	FC5E		00126	STC 0,TB00L	ALMACENA EL VALOR DEL PRIMER PARAMETRO	
CCCCCA	5823	F05E		0012E	LM 2,3,TB00L		
0000CE	5841	00J4		CC004	L 4,4(1)	CARGA DIRECCION DEL SEGUNDO PARAMETRO	
0000E2	5844	CC0C		0000C	L 4,0(4)	CARGA CONTENIDO DEL SEGUNDO PARAMETRO	
0000E6	1244			127	LTP 4,4	PRUEBA SI ES POSITIVO	
0000E8	4740	FC3E	00108	128	BM RIGHT	DIFERENCIA SI NEGATIVO	
CC00EC	3020	4000		129	SLDL 2,0(4)	DESPLAZA A LA IZQUIERDA	
0000F0	1832			130 RET	LR 0,2	CARGA RESULTADO DE LA FUNCION ENTERA	
CC00F2	5023	FC5E	00128	131	STM 2,3,TB00L	ALMACENA RESULTADO	
CC00F6	6800	F05E	00128	132	LC 0,TB00L	CARGA RESULTADO DE LA FUNCION REAL	
CC00FA	78EF	D00C	CCCCC	133	LM 14,15,12(13)	RESTAURA REGISTROS GENERALES 14,15	
CC00FE	52FF	D00C	000CC	134	MVT 12(13),X'FFF'	RETORNA CORRECTO	
CC0102	9810	D010	CCCCE	135	LM 1,12,24(13)	RESTAURA REGISTROS GENERALES 1 A 12	
CC0106	07FE			136	BR 14	RETORNA A QUIEN LLAMA	
CC010E	1044			137 RIGHT	LPR 4,4	CARGA VALOR POSITIVO DEL SEG. PARAM.	
CC011A	8020	4000	CCCCC	138	SRCL 2,0(4)	DESPLAZA A LA DERECHA	
CC011E	47FC	F026	000F0	139	E RET	DIFERENCIA A COMPLETAR LA FUNCION	
				00112	ECU *		
				00112	SHIFT	ECU *	
				00112	142	USING *,15	
				0000C	143	STM 14,12,12(13)	ALMACENA REGISTROS GENERALES
CC0112	90EC	000C		144	SR 2,3	CREA LCS 32 BITS DE LA DERECHA PRIM. PAR.	
CC0116	1833			145	L 2,0(1)	CARGA DIRECCION DEL PRIMER PARAMETRO	
CC011E	5821	0000	00000	146	L 2,0(2)	CARGA A LA IZQUIERDA EL CONT DEL PRIM. PAR.	
CC011C	5822	0000	00138	147	L 15,=A(CSHIFT)	CARGA REGISTRO BASE FFFF DIRECCIONAR	
CC012C	58FC	F026	000CA	148	USING DSHIFT,15		
CC0124	47F0	F014	CC0CE	149	BC 15,15	DIFERENCIA A COMPLETAR LA FUNCION	
CC012E				150 TB00L	ES 0		
CC0130	FFFFFF			151 XFF	CC X'FFFFFF'		
CC0138				152	LTCR		
CC013A	000000CA			153	=A(DSHIFT)		
00013C	404040			154	=CL3'		
				155	END		

zabarr (Pascado)

zabarr (Pascado)

Pascado

CROSS-REFERENCE

PAGE 8

SYMBOL	LEN	ID	VALUE	DEFN	REFERENCES
AND	00001	01	00004A	CCC58	0011
BNORM	00001	01	000000	00010	
BCOL	00001	01	00004A	00054	
BCLEAN	00001	01	00004A	00055	0011
CCMPL	00001	01	0000A2	00094	0011
LSHIFT	00001	01	00000A	CC118	0148 0153
GETCHR	00001	01	CCCC22	0C035	0011
L/IC	00001	01	00004A	00057	0011
L/CMPL	00001	01	0000A2	CC093	0011
LCR	00001	01	000076	00075	0011
LSHIFT	00001	01	000112	00140	0011
OR	00001	01	000076	CCC76	0011
PUTCHR	00001	01	000000	00014	0011
PUT	00002	01	0000F0	00130	0135
RIGHT	00002	01	000108	00137	0128
RI	00004	01	0000DE	00125	0149
SHIFT	00001	01	000112	CC141	0012
TBCCL	00008	01	CCCC28	00150	0066 0067 0064 0065 0102 0103 0123 0124 0121 0132
XFF	00004	01	000130	00151	0100
=A (DSHIFT)	00004	01	000138	00153	0147
=CL3	00003	01	00013C	00154	CC45

78-03-30

zubini Database

zubini Database

zubini Database

DIAGNOSTICS AND STATISTICS

PAGE 5

78-03-30

NO ERRORS FOUND

OPTIONS FOR THIS ASSEMBLY - ALIGN, LIST, XREF, NCLINK, NOCHECK, NOEDEC

THE ASSEMBLER WAS RUN IN 215016 BYTES
END OF ASSEMBLY

zabiri Baacki

zabiri Baacki

zabiri Baacki