

ORIGINAL

ESCUELA UNIVERSITARIA POLITECNICA DE LAS PALMAS DE GRAN CANARIA

TITULO : TECNICAS DE DEPURADO DE SOFTWARE. EL PROGRAMA ARCHALUM

Autor:

Tutor:

Carlos García Martí

Sebastián Suarez Gil

I N D I C E

PARTE I: TECNICAS DE DEPURADO DE SOFTWARE

	PAGINA
1. ERRORES DE SOFTWARE.....	001
1.1 Introducción.....	001
1.2 Errores y faltas de software.....	001
1.3 Errores nuevos y primitivos.....	002
1.4 Causas que generan error.....	002
1.4.1 Hipótesis 1.....	003
1.4.2 Hipótesis 2.....	003
1.4.3 Hipótesis 3.....	004
1.4.4 Hipótesis 4.....	004
1.5 Clasificación de errores.....	005
2. CONSIDERACIONES GENERALES SOBRE EL TESTEO.....	006
2.1 Introducción.....	006
2.1.1 Definiciones.....	006
2.1.2 Interrelación entre testeo, diseño y fiabilidad.....	007
2.2 El testeo exhaustivo.....	009
2.3 Clasificación jerárquica de los testeos.....	010
2.3.1 Testeo tipo 0.....	010
2.3.2 Testeo tipo 1.....	011
2.3.3 Testeo tipo 2.....	012
2.3.4 Testeos tipo 3 y 4.....	012
3. TECNICAS DE TESTEO.....	013
3.1 Introducción.....	013
3.1.1 Diseño top-down.....	013
3.1.2 Diseño bottom-up.....	014
3.2 Testeo modular.....	014
3.3 Testeo por integración.....	015
3.4 Testeo top-down frente a testeo bottom-up...	016
3.5 Testeo mixto.....	017

	PAGINA
3.5.1 Solución big bang.....	017
3.5.2 Testeo top-down modificado.....	018
3.5.3 Testeo top-down-bottom-up.....	019
3.6 Testeo regresivo.....	019
4. DEPURADO DE SOFTWARE.....	020
4.1 Introducción.....	020
4.2 Tecnicas de depurado.....	020
4.3 Estrategias de depurado.....	022
4.3.1 Introducción.....	022
4.3.2 Aproximaciones sucesivas.....	022
4.3.3 Backtracking.....	022
4.3.4 Método inductivo.....	023
4.3.5 Método deductivo.....	025
5. COMANDOS DE DEPURADO DEL BASIC/3000.....	027
5.1 Comandos TRACE/UNTRACE.....	027
5.2 Comandos BREAK/UNBREAK.....	029
5.3 Comando ABORT.....	031
5.4 Comando GO.....	032
5.5 Comando SHOW.....	032
5.6 Comando SET.....	033

I N D I C E

PARTE II: EL PROGRAMA ARCHALUM

	PAGINA
1. ANALISIS DEL PROGRAMA ARCHALUM.....	037
1.1 Características generalaes.....	037
1.2 Módulo 1. Programa Menú.....	038
1.3 Módulo 2. Matriculación de alumnos.....	040
1.4 Módulo 3. Renuncia de convocatorias.....	049
1.5 Módulo 4. Cambio de convocatorias.....	054
1.6 Módulo 5. Introducción de notas.....	059
1.7 Módulo 6. Volcado de ficheros.....	064
1.8 Módulo 7. Listado por carrera.....	068
1.9 Módulo 8. Listado por curso.....	072
1.10 Módulo 9. Listado por asignatura.....	075
1.11 Módulo 10. Tabla por curso.....	079
1.12 Módulo 11. Asignaturas existentes.....	082
1.13 Módulo 12. Ficha del alumno.....	085
1.14 Módulo 13. Listado por nacionalidad.....	088
1.15 Módulo 14. Listado por carrera terminada...	092
2. FICHEROS UTILIZADOS POR EL PROGRAMA ARCHALUM....	095
2.1 Formato de los ficheros creados en BASIC/ 3000.....	095
2.2 Ficheros tipo "C".....	096
2.2.1 Características.....	096
2.2.2 Estructura.....	096
2.2.3 Formato de grabaciones del fichero tipo "C".....	098
2.3 Fichero de expedientes.....	099
2.3.1 Características.....	099
2.3.2 Estructura.....	099
2.3.3 Formato de grabaciones del fichero de expedientes.....	101

	PAGINA
2.4 Archivos auxiliares.....	102
2.4.1 Archivo HELP.....	102
2.4.1.1 Características.....	102
2.4.1.2 Estructura.....	102
2.4.1.3 Formato de grabaciones del archivo HELP.....	103
2.4.2 Archivo TEMP.....	104
2.4.2.1 Características.....	104
2.4.2.2 Estructura.....	105
2.4.2.3 Formato de grabaciones del archivo TEMP.....	106
2.5 Dimensión máxima de las variables que introducen datos en archivos.....	107
2.6 Interpretación de las codificaciones de los datos académicos.....	107
2.6.1 Codificación en archivos tipo "C" y HELP.....	107
2.6.2 Codificación en archivo de expedientes y archivo TEMP.....	113
3. LISTADO DEL PROGRAMA ARCHALUM.....	115

OBJETIVO

La finalidad de este proyecto ha sido la construcción de un programa de gestión de archivos aplicado a la EUP de Las Palmas de Gran Canaria, para controlar la gestión referente al alumnado. Dada la importancia de la fiabilidad de los datos, se acompaña de un estudio teórico de las técnicas de depurado existentes hasta hoy.

La gestión de archivos se realiza a través de una serie de ficheros, que al igual que el resto del programa están protegidos por una clave de entrada que evita el acceso a los datos por parte de personas no autorizadas. Cabe señalar asimismo que en el desarrollo del programa se han respetado fielmente las incompatibilidades de matriculación en vigor, por lo que el programa está no sólo preparado para manejar los datos de alumnado, sino para rechazar aquellos datos que sean incompatibles con las características estatutarias de matriculación. Sin embargo esto no es obstáculo para una fácil adaptación del programa para el uso de este con vistas a otras facultades de características diferenciadas, dada la estructura elástica de los datos susceptibles de ser introducidos como índices de los ficheros. Finalmente señalar que el programa ha sido intensamente depurado, por lo que ofrece una alta fiabilidad en su aplicación y manejo.

PARTE I

TECNICAS DE DEPURADO DE SOFTWARE

1. ERRORES DE SOFTWARE

1.1 Introducción

En esta sección se hace un estudio inicial de los errores que pueden aparecer en un programa y cuya eliminación constituye el objetivo de los procesos de testeo y depurado analizados en secciones posteriores. Para ello enunciamos una serie de definiciones de los tipos de error, sus causas y una enumeración de algunos tipos de error más frecuentes.

1.2 Errores y faltas de software

Debe diferenciarse primeramente entre error y falta de software. Un error de software es siempre atribuible a una falta. Además un error origina un fallo del sistema mientras que no siempre ocurre así con una falta. Por ejemplo, cuando uno de dos elementos redundantes de un sistema es defectuoso y el otro opera correctamente no aparece ningún error. Así pues, podemos distinguir tres tipos de errores. Los repetitivos (el mismo error se manifiesta en distintos puntos del programa), los errores múltiples causados por una sola falta, y el error simple causado por un conjunto de faltas.

No entran en la definición de errores de software aquellos atribuibles a un fallo del operador, si bien este error puede producir un error o una falta del sistema. La misión del programador consiste en parte en

evitar la comisión de faltas por parte del operador en situaciones cruciales.

1.3 Errores nuevos y primitivos

En el transcurso de un proceso de depuradoses útil saber si el error es nuevo o primitivo. Para ser más precisos podemos usar la terminología de "error previamente fijado" y "error generado". Definimos estos términos en la siguiente forma:

-Un error previamente fijado es aquel que se repite básicamente en la misma forma después que el programador ha terminado de trabajar en la modificación de la codificación creyendo que el error ha sido corregido. La conclusión de que un error es un "error previamente fijado" puede asumirse sólo después del estudio de la falta correspondiente.

-Un error generado es aquel que no existe hasta que es creado como consecuencia de una modificación hecha para corregir otro error. La mejor manera de diagnosticar un error generado es hallando la falta que lo causa. Sin embargo, a veces es posible clasificarlo como error generado debido a la naturaleza de este, por ejemplo si una variable recién creada aparece en un formato erróneo.

1.4 Causas que generan error

Existen varias hipótesis sobre los errores de programa, las cuales enumeramos en este apartado.

1.4.1 Hipótesis 1ª: El número de errores por línea es constante.

La codificación en lenguajes de alto nivel tiene un número de errores inicial menor. Para ello se argumentan dos motivos:

(1) En un lenguaje de alto nivel como el PL/1, por ejemplo, ciertas condiciones excepcionales como dividir por cero, overflow, etc. son tratadas automáticamente. Sin embargo, si se escriben en lenguaje máquina se debe tratar estos problemas cada vez y existe la posibilidad de error por omisión.

(2) Algunos autores proponen que el número de faltas creadas por línea de código es el mismo, tanto si se escribe en lenguaje de alto nivel como en lenguaje máquina. Cada línea de un lenguaje como el PL/1 abarca de 5 a 10 líneas de lenguaje máquina. Si este efecto es cierto debe haber una reducción de contenido de error correspondiente a un factor entre 5 y 10.

1.4.2 Hipótesis 2ª: La reducción de memoria favorece la aparición de errores.

El contenido de error inherente a algunos programas se atribuye a la escasez de contenido de memoria disponible. La teoría es que si la memoria es muy restringida, los programadores tendrán que recurrir a la técnica de segmentación de programas y otros "trucos" de codificación para comprimir la función deseada en el espa-

cio de memoria asignado. Se da por sentado que estos trucos introducen gran complejidad y son el origen de muchas faltas.

1.4.3 Hipótesis 3ª: Un sistema muy cargado origina errores.

Este efecto es muy difícil de demostrar ya que es difícil definir el concepto de carga de un computador. Está claramente relacionado con el número de usuarios de un sistema multiusuario o con el número de procesos que tienen lugar en un sistema multiproceso. También hay que tener en cuenta la complejidad de las tareas que son procesadas y la diversidad de recursos que se entremezclan durante la ejecución de aquellos.

1.4.4 Hipótesis 4ª: La puesta a punto disminuye la frecuencia de aparición de errores.

Es un hecho conocido que un sistema hardware/software puede ser afinado de tal modo que funcione correctamente. Esto no es la conocida optimización de los parámetros de un sistema sino que tiene que ver con la eliminación de errores conocidos para un determinado tipo de datos de entrada. Así, si el mismo software es sometido a un número significativo de conjuntos de entradas, aparecerá toda una serie de nuevos errores. En tal caso podríamos poner a punto el programa eliminando los nuevos errores, de tal forma que el programa funcionaría bien con las nuevas entradas.

1.5 Clasificación de errores

Quien haya hecho alguna vez un depurado extenso, sabrá que hay unos tipos de faltas que se repiten con tínuamente. Además algunas clases de faltas ocurren solamente al programar en lenguaje ensamblador, otras al programar en lenguaje de alto nivel y algunas en ambos casos. La siguiente lista enumera algunos de los tipos de error.

- 1.) Uso de variables no dimensionadas.
- 2.) Equivocarse por 1 al indexar o usar un contador.
- 3.) Bit de flag incorrecto.
- 4.) Problemas de complementación aritmética.
- 5.) Inicialización de variables.
- 6.) Problemas de puntero.
- 7.) Transferencia de control incorrecta.
- 8.) Problemas de direccionamiento indirecto.

Otro punto a tener en cuenta es que los diferentes estilos de programación pueden dar pie a distintos tipos de errores. Por ejemplo, habrá algunos tipos de faltas que aparecerán en una programación top-down y difícilmente en la bottom-up y viceversa. Análogamente las faltas que aparezcan en lenguajes de alto nivel, lenguaje ensamblador y en microprogramación no serán los mismos.

2. CONSIDERACIONES GENERALES SOBRE EL TESTEO

2.1 Introducción

El desarrollo de un sistema de software consta de tres fases principales: diseño, codificación y depurado. Esta última fase abarca aproximadamente de un 30% a un 50% de los esfuerzos del desarrollo, y consiste en la localización y corrección de errores. Hay que hacer notar que aunque las palabras testeo (búsqueda de errores) y depurado (localización y corrección de errores) suelen usarse indistintamente, son conceptos diferentes. Podríamos decir que aunque erróneamente, se entiende por depurado de un sistema su testeo unido al depurado propiamente dicho.

2.1.1 Definiciones

-Testeo: Es el proceso de ejecutar un programa (o parte de él) con la intención de encontrar errores.

-Verificación: Es el intento de encontrar errores emulando un programa.

-Validación: Es el intento de encontrar errores ejecutando un programa en un equipo determinado.

-Depurado: Es el proceso que implica el diagnóstico preciso de un error conocido y su corrección.

-Testeo modular: Es la verificación de un módulo del programa, normalmente en un entorno aislado.

-Testeo de integración: Es la verificación de las

interconexiones de un sistema (módulos y subsistemas).

-Testeo del sistema: Es la verificación y/o validación del sistema bajo el punto de vista de sus objetivos iniciales. El testeo del sistema es un proceso de verificación si se realiza en un entorno simulado; es un proceso de validación si se realiza en un entorno real.

-Testeo de aceptación: Es la validación del sistema o programa bajo el punto de vista de los requerimientos del usuario.

-Testeo de instalación: Es la validación de cada instalación particular del sistema con el intento de encontrar algún error cometido al instalar el sistema.

-Fiabilidad de un sistema: Es la probabilidad de que un determinado sistema opere durante cierto tiempo sin error de software sobre la máquina para la que fue diseñada, presuponiendo que es usado dentro de los límites de diseño.

2.1.2 Interrelación entre testeo, diseño y fiabilidad.

La evidencia de que el coste de corregir un error hallado al principio del desarrollo de un sistema frente a uno hallado más tarde, es menor, podría llevar a la conclusión de concentrar todo el esfuerzo de testeo al principio del desarrollo. Sin embargo este razonamiento no es válido, porque algunos errores se manifiestan por sí mismos, y sólo pueden ser detectados después de en-

samblar todos los módulos que constituyen el sistema.

Existe una clara interrelación entre diseño y testeo, ya que un buen número de errores se introducen en el propio diseño. Sin embargo, apesar de nuestros esfuerzos en optimizar el diseño, siempre será necesario un testeo posterior para descubrir los errores introducidos.

La interrelación entre fiabilidad y testeo se manifiesta de diferentes formas. En primer lugar el testeo elimina una serie de errores y determina la fiabilidad de un programa. Es una decisión técnica la que determina el testeo mínimo necesario que elimine los errores residuales suficientes para producir un sistema con una fiabilidad aceptable. Por otra parte, el supervisor del sistema debe aspirar a un sistema fiable, de alta calidad y que satisfaga plenamente al usuario. Esta es también una cuestión de costes, ya que una empresa puede abaratar los costes de un programa reduciendo el testeo, pero con el riesgo de perder credibilidad ante los clientes; o depurar el programa de tal manera que encarezca el producto obteniendo ganancias mínimas.

2.2 El testeo exhaustivo

Los diseñadores de software mantienen que un programa podría quedar libre de errores si pudiera ser testado exhaustivamente. En la práctica, sin embargo, esto resulta imposible. Veámoslo con un ejemplo:

Supongamos que el ensamblador es capaz de detectar todos los errores sintácticos del programa, con lo cual los posibles errores restantes serán algorítmicos o lógicos. Si elegimos como algoritmo el que se deduce de la fórmula

$$R_1, R_2 = \frac{-B \pm \sqrt{B-4AC}}{2A}$$

podemos demostrar que la fórmula es correcta (sólo habría que analizar el caso de las raíces complejas o de dividir por cero), y por tanto asumir fácilmente que el algoritmo es correcto también.

Debemos ahora considerar los errores lógicos. Un error lógico sería la transferencia errónea de control dentro del programa (IF "expresión" THEN DO WHILE "expresión"). Otra posibilidad de error lógico abarca el overflow de una palabra de longitud finita, diferencias entre el cero positivo y cero negativo, etc.

Suponiendo que hemos escrito el programa en lenguaje ensamblador, todos estos problemas deben ser considerados posibles causas de error. Esto significa que existe la posibilidad de que un programa funcione para cierto conjunto de datos, pero que para otros dé error.

Por este motivo, para que el testeo sea exhaustivo, de bemos probar todas las posibles combinaciones de A, B y C, y obtener las soluciones de R_1 y R_2 . Paralelamente debemos comprobar que se cumple $B/A = -(R_1 + R_2)$ y $C/A = R_1 - R_2$ para verificar el cálculo. Suponiendo que la palabra del computador sea de 12 bits habría como máximo $2^{12} = 4096$ palabras distintas para cada uno de los coeficientes A, B y C.

El testeo exhaustivo propuesto hasta aquí es imposible por las siguientes razones:

(1) Es imposible prácticamente, chequear las combinaciones creadas ($2^{12} * 2^{12} * 2^{12} = 64 * 10^9$).

(2) El tiempo de cálculo es prohibitivo. Supongamos que el tiempo que tarda en ejecutarse una vez el programa es 250 s. El tiempo de comprobar todas las posibilidades será pues:

$$64 * 10^9 \text{ combinaciones} * 250 * 10^{-6} \text{ s por combinación} = \\ = 16 * 10^6 \text{ s} * 5000 \text{ horas}$$

2.3 Clasificación jerárquica de los testeos

Ante la imposibilidad de realizar un testeo exhaustivo de un programa, se ha hecho una hecho una clasificación de los testeos por su nivel de dificultad creciente que va desde el nivel 0 al nivel 4

2.3.1 Testeo tipo 0

Este testeo se basa en probar por lo menos una vez

cada una de las instrucciones del programa. Esta tarea tan trabajosa se simplifica enormemente con el uso de compiladores modernos que indican el número de veces que se realiza cada instrucción.

Una forma común de realizar el test tipo 0 es ejercitar cada una de las funciones del programa por lo menos una vez y chequear la codificación del programa para comprobar que cada instrucción se ha ejecutado al menos una vez.

2.3.2 Testeo tipo 1

Uno de los problemas que se plantean al testear un programa a un nivel superior al tipo 0 es la dependencia entre los datos y los predicados de las tomas de decisión, los cuales dan la opción de ejecutar el programa por distintos caminos. El testeo tipo 1 se basa pues en lo que podríamos llamar ejecución forzada y que consiste en que el testeador solamente reconoce el hecho de que ha sido alcanzada una toma de decisión en el seguimiento del programa. Una vez alcanzada la toma de decisión, se fuerza la ejecución del programa para dos casos: uno donde la premisa es cierta, y otro donde la premisa es falsa. Así pues, la ejecución forzada implica el seguimiento de todos los caminos en que se ramifica un programa, incluyendo aquellos que nunca serán alcanzados debido a las características de la premisa.

2.3.3 Testeo tipo 2

Este testeo es similar al anterior, pero se basa en lo que llamamos ejecución natural de un programa. Consiste en que el testeador lee los predicados de las tomas de decisión, computa si son ciertos o falsos, basándose en el valor actual de las variables y sigue la ramificación correspondiente basándose en el resultado del cómputo.

2.3.4 Testeos tipo 3 y 4

Definimos el testeo tipo 3 como un testeo exhaustivo de un sistema en el que la secuencia de entradas y las condiciones de entrada son fijas. Análogamente el testeo tipo 4 es un testeo exhaustivo de un sistema en el que las condiciones iniciales o la secuencia de entradas, o bien ambas, pueden cambiar durante la ejecución del programa.

3. TECNICAS DE TESTEO

3.1 Introducción

En esta sección se compara una serie de diferentes técnicas de testeo que están íntimamente relacionadas con el proceso de diseño. Por ello defino primeramente algunos conceptos básicos de los tipos de diseño que servirán para comprender las técnicas de depurado expuestas posteriormente.

3.1.1 Diseño top-down

El diseño top-down es básicamente un proceso de descomposición que se enfoca sobre la estructura de control del programa. El primer paso es estudiar todos los aspectos del proyecto de que se dispone y de dividirlos en módulos que cumplan individualmente funciones determinadas. El segundo paso es dividir cada uno de estos módulos en submódulos independientes. Este proceso se repite hasta que se obtengan módulos que sean lo suficientemente reducidos para abarcarlos mentalmente y codificarlos de una forma sencilla.

Una característica importante del diseño top-down es que a cada nivel modular, los detalles del nivel inferior están ocultos. Si una estructura de datos está contenida íntegramente en un módulo de nivel inferior, no necesita ser definido hasta que el nivel sea alcanzado en el proceso de diseño. Sin embargo si los datos deben

ser compartidos por varios módulos de distinto nivel, la estructura de datos deberá ser definida antes de pasar a un nivel inferior.

3.1.2 Diseño bottom-up

En un diseño bottom-up se elige un sistema típico de diseño y se decide por intuición o por un análisis somero qué partes del diseño son más difíciles i limitadoras. Estas partes del diseño son analizadas en primer lugar y se toman las decisiones oportunas para el diseño. El resto del diseño se adapta luego al diseño inicial.

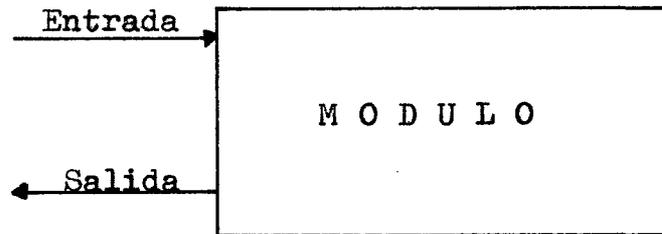
El diseño bottom-up se puede definir como una síntesis. Se dan las especificaciones para las partes clave y el diseño es formulado para esas partes. Si el diseño no es viable para las especificaciones dadas, éstas se cambian y el diseño continúa.

3.2 Testeo modular

Esta técnica es la más usada en la parte inicial del proceso de testeo. Consiste en ir probando los distintos módulos que abarquen una función determinada. Para que la eficacia de este tipo de testeo sea máxima los módulos deben estar claramente diferenciados (lo cual se consigue con una programación bien estructurada), de tal forma que el acceso y la salida de un determinado módulo se realice cada uno por un solo si-

sto,

tio, de tal manera que al testear sea fácil comparar el dato introducido con el dato obtenido.



3.3 Testeo por integración

Está claro que ciertos errores que se desprenden de la interacción de diferentes módulos de un programa no pueden ser hallados por testeo modular, sino sólo por un testeo interactivo. El proceso por el cual se unen los módulos de un programa para que éste realice todas sus funciones, se llama integración del sistema. Cuando se realizan testeos que comprueban la interacción entre los módulos de un programa, se denomina a este proceso como testeo de integración. El testeo de integración es seguido de un test de simulación de funcionamiento del programa, que constituye el último paso en el testeo de integración.

3.4 Testeo top-down frente a testeo bottom-up

En el testeo top-down empezamos con el programa de control y examinamos la forma en que recibe los datos, cómo pasa el control y los datos a varios módulos del programa, cómo los módulos devuelven datos al programa de control y cómo éste pasa los datos al periférico de salida. Obviamente el programa de control deberá estar codificado, antes de que podamos empezar el testeo top-down.

Si el sistema fuera codificado en forma bottom-up, el programa de control sería escrito el último, y si insistieramos en un testeo top-down, los módulos ya codificados quedarían sin testear hasta que el programa de control estuviera terminado. De este modo, si deseamos hacer un testeo top-down, es casi imprescindible que esté precedido por una codificación top-down. Sin embargo, un programa codificado y depurado en la forma top-down puede ser diseñado tanto en forma top-down como en forma bottom-up.

Durante la fase de testeo, la mayor diferencia entre testeo top-down y bottom-up es la necesidad de usar respectivamente "stubs" de testeo y "drivers" de módulos. Al testear por el método top-down un programa codificado en la misma forma, necesitamos como mínimo una instrucción de salida por cada módulo del programa para comprobar que el control del programa ha pasado por el módulo durante el testeo de la estructura del programa.

ma de control. Estas líneas falsas codificadas en el programa son llamados stubs de testeo. Estos stubs son eliminados o modificados en pasos subsiguientes, cuando se trata de codificar cada uno de los módulos.

Por contra, al testear por el método bottom-up un sistema codificado en la misma forma, deberemos construir drivers de testeo. El driver es un programa simulado que introduce datos en el módulo a testear y recibe los datos que éste ha procesado. La complejidad y costo del driver depende la importancia de la función que desempeña el módulo dentro del programa.

3.5 Testeo mixto

Hasta aquí se ha hablado solamente de estrategias de testeo top-down y bottom-up puras. Sin embargo es posible escoger estrategias intermedias, y de hecho muchas veces es preferible. La cuestión está en elegir una solución intermedia que contenga las mejores características de los dos métodos y que elimine o reduzca los inconvenientes que tengan ambos. En los siguientes subapartados se exponen las soluciones intermedias que conservan las características positivas de los métodos top-down y bottom-up.

3.5.1 Solución big bang

Este método consiste en codificar uno por uno los módulos que constituyen el sistema y testearlos indi-

vidulmente integrándolos todos de una vez. Esto se consigue con un sistema diseñado cuidadosamente por módulos, codificado por el sistema bottom-up y testeado modularmente. Con ello se reduce considerablemente la fase de integración definitiva, ya que el testeo final se minimiza. Sin embargo no es recomendable debido a la dificultad de diagnosticar los posibles errores que aparecen al ensamblar todos los módulos de una vez.

3.5.2 Testeo top-down modificado.

Debido a la características de diseño top-down, los módulos críticos no pueden ser testeados inmediatamente, sino que debe terminarse previamente la estructura de control. La solución es aplicar un diseño top-down modificado. De forma similar podemos hacer testeo top-down modificado. Esto significa que mientras el testeo por integración avanza un módulo crítico está siendo probado con un programa de testeo tipo "driver". Se presupone que todos los módulos han pasado previamente por un testeo modular independiente. La diferencia entre el testeo modular y el testeo con driver sobre un mismo módulo, estriba en que en el primer caso nos encontramos con las funciones internas del módulo, y en el segundo con su interrelación con el programa de control.

3.5.3 Testeo top-down-bottom-up.

Si hay más de un módulo crítico o si éste está muy interrelacionado con los demás, estamos prácticamente forzados a trabajar desde ambos extremos hacia la mitad. En este caso es muy importante elegir correctamente los módulos destinados a una integración tipo top-down y los destinados al tipo bottom-up. Como ya se ha constado, en los módulos críticos se opta por el bottom-up. Sin embargo hay que sopesar el coste adicional de construir drivers y stubs frente al testeo modular independiente.

3.6 Testeo regresivo.

Independientemente del compromiso elegido para testear, al descubrir un error éste se corrige y por lo tanto se modifica el software. El problema planteado entonces es decidir si hay que empezar de nuevo y repetir todos los tests. A esta repetición global o parcial se le denomina testeo regresivo. La repetición de todos los tests encarece el producto, pero es prudente repetir cierto número de ellos. En sistemas con una buena estructuración modular con poca interacción es posible repetir solamente los tests referidos a ese único módulo. En todo caso debe especificarse en el plan de testeo la normativa a seguir para realizar el testeo regresivo.

4. DEPURADO DE SOFTWARE

4.1 Introducción

Una vez que un error ha sido descubierto mediante el testeo, se inicia un proceso para localizarlo y de terminar su causa, al que llamamos depurado. Una vez analizado el fallo, se rediseña el programa y se repi ten algunos testeos para comprobar que el error ha de saparecido.

4.2 Técnicas de depurado

En la tabla adjunta se muestran la técnicas de de purado más conocidas. En general ninguna de dichas técnicas debe de ser aplicada sin un cuidadoso análi sis previo de los síntomas del error que conduzca a una hipótesis de las causas del error.

Vuelcos de memoria

Características: Obtención y listado de todos los registros y contenidos de memoria relevantes.

Ventajas: 1) En un instante crucial se obtiene el contenido de memoria completo, para su estudio. 2) Disminuye el esfuerzo de depurado si se usa para validar una hipótesis de error bien formulada.

Desventajas: 1) Requiere algún tiempo de CPU, un tiempo significativo de E/S, y mucho tiempo de análisis. 2) Es una pérdida de tiempo si se usa indiscriminadamente.

Instrucciones Print

Características: La instrucción Print estándar del lenguaje utilizado se introduce a lo largo del programa para conocer el valor de las variables deseadas.

Ventajas: 1) Es una forma sencilla de saber si una variable determinada varía de la forma adecuada tras determinado evento. 2) Una secuencia de instrucciones Print refleja el cambio dinámico de la variable estudiada.

Desventajas: 1) Son engorrosos de usar en programas largos. 2) Si se usan indiscriminadamente producen un gran número de datos a analizar, muchos de los cuales sobran.

4.3 Estrategias de depurado

4.3.1 Introducción

Como se acaba de constatar, la clave del depurado no son las herramientas disponibles, sino la estrategia usada para descubrir la causa del error.

4.3.2 Aproximaciones sucesivas

Una técnica obvia es la de aproximaciones sucesivas. El depurador observa los síntomas del error, hace un juicio inmediato de dónde puede estar el error y se lanza a corregir el programa con una o varias de las técnicas de depurado. Obviamente esta es una solu
ción lenta y costosa.

4.3.3 Backtracking

Otra técnica llamada "backtracking" consiste en examinar los síntomas del error para ver dónde fueron observados por primera vez. Se retrcede entonces en el flujo del programa hasta el punto en que los síntomas desaparecen. Generalmente, este proceso localiza la zona del programa que contiene el error. Un subsiguiente estudio cuidadoso de la zona afectada re
vela generalmente la causa del error. Otra variante del "backtracking" es el "forwardtracking", que consiste en avanzar desde determinado punto introduciendo

instrucciones de salida para observar en qué punto del programa el resultado empieza a ser incorrecto. Si asumimos que conocemos los valores correctos de las variables en determinados puntos claves, dentro de un programa podemos adoptar una estrategia de búsqueda de tipo dicotómico. Un conjunto de datos es introducido a la mitad del programa y se comprueban los datos de salida. Si estos son correctos, el error está en la primera parte del programa; si son incorrectos el error está en la segunda mitad. Este proceso se repite hasta que sea posible localizar la zona donde se produce el error para su análisis.

Dos soluciones más para el depurado son el método inductivo y el método deductivo.

4.3.4 Método inductivo

(1) Localizar los datos pertinentes.

El mayor error realizado al depurar un programa es no tener en cuenta todos los datos y síntomas disponibles acerca del problema. El primer paso es la enumeración de todo aquello que se sabe que el programa realiza correctamente y lo que realiza incorrectamente. Otras claves adicionales nos las dan otro tipo de testeo diferentes pero semejantes que no nos den error.

(2) Organizar los datos.

El segundo paso es estructurar los datos pertinentes para permitirnos observar semejanzas y los patrones que siguen.

(3) Construir una hipótesis.

El siguiente paso es estudiar las relaciones entre las pistas y las suposiciones usando los patrones que hayamos detectado en el paso anterior, hasta concluir una o varias hipótesis. Si no es posible construir ninguna teoría, es necesario recopilar más datos, que pueden obtenerse ejecutando tests adicionales. Si aparecen varias teorías se selecciona primero la más probable.

(4) Probar la hipótesis.

Uno de los mayores errores al depurar un programa consiste en saltarse este paso intentando resolver directamente el fallo del programa. Esto conduce con frecuencia a eliminar sólo un síntoma del problema o parte de él. La hipótesis se prueba comparándola a los datos o pistas originales, asegurándose de que explique completamente la existencia de las pistas. Si no es así, la hipótesis no es válida o está incompleta, o bien existe más de un error.

Para una más fácil comprensión obsérvese el diagrama adjunto.

4.3.5 Método deductivo

- (1) Enumerar las posibles causas o hipótesis.

El primer paso es desarrollar una lista de todas las causas de error posibles. No es necesario que sean explicaciones completas. Son meras teorías con las cuales podemos estructurar y analizar los datos disponibles.

- (2) Usar los datos para eliminar causas de error posible.

Mediante un análisis cuidadoso de los datos, especialmente buscando contradicciones, se intenta eliminar todas las causas posibles menos una. Si todas las causas quedan eliminadas es necesario recopilar más datos (por ejemplo con tests adicionales) para concebir nuevas teorías. Si queda más de una causa posible, se selecciona primero la más probable.

- (3) Refinar la hipótesis.

La posible causa del error puede ser correcta pero no lo suficientemente específica para determinar el error. El siguiente paso debe ser usar las pistas disponibles para refinar la teoría en algo más específico.

- (4) Probar la hipótesis.

La hipótesis se prueba comparándola a los datos o pistas originales, asegurándose de que explique completamente la existencia de las pistas. Si no es así la hipótesis no es válida o está incompleta, o

bien existe más de un error.

5. COMANDOS DE DEPURADO DEL BASIC/3000

El lenguaje BASIC/3000, por sus características peculiares al usar un intérprete, proporciona una serie de comandos que constituyen una herramienta muy valiosa de cara al depurado de programas. El estudio de estos comandos se realiza en esta sección.

5.1 comandos TRACE/UNTRACE

El comando TRACE se usa para activar el seguimiento de las variables seleccionadas, simples y matriciales, y de las instrucciones del programa en curso. El comando UNTRACE inhibe el seguimiento.

Formato:

TRACE (lista de elementos)

UNTRACE (lista de elementos)

La lista de elementos incluye variables, etiquetas, y rangos de etiquetas (etiqueta-etiqueta).

Las variables pueden ser simples, alfanuméricas, matriciales alfanuméricas y matriciales numéricas. Las variables matriciales se distinguen de las que no lo son añadiendo (*) o (*,*) a continuación de la matriz. Ejemplos de variables:

A, B(2)	Variable numérica
B\$, C\$(*)	Variable alfanumérica
C\$(*,*)	Matriz alfanumérica
A(*)	Matriz numérica de una dimensión
A(*,*)	Matriz numérica de dos dimensiones

Las etiquetas son etiquetas de instrucciones consistentes en enteros del 1 al 9999

El rango de etiquetas abarca todas las instrucciones intercaladas, incluyendo las instrucciones identificadas por las dos etiquetas. Por ejemplo:

80 Instrucción de la etiqueta 80
 30-70 Todas las instrucciones entre 30 y 70, ambas inclusive.

Explicación:

El intérprete del BASIC/3000 hace un rastreo de todos los datos especificados en el comando TRACE. Un rango de etiquetas implica el seguimiento de todas las instrucciones comprendidas en el rango. Cualquier cambio en el valor de una variable o la ejecución de una instrucción hace que se visualice (en pantalla o impresa) la información referente al seguimiento. Cualquier cambio en el valor de un elemento matricial conduce a la visualización de dicho elemento.

El comando TRACE, si no va acompañado de parámetros, produce el listado de los datos sometidos a seguimiento.

El comando UNTRACE inhibe el seguimiento de los datos que se especifiquen. Si no va acompañado de parámetros, detiene el seguimiento de todos los datos especificados en anteriores comandos TRACE.

El seguimiento especificado ocurre sólo cuando se ejecuta el programa. Cuando se produce la visualización esta tiene la siguiente forma:

@ etiqueta variable = valor

- TRACE etiqueta

dependiendo de si aparece o no una variable.

5.2 Comandos BREAK/UNBREAK

El comando BREAK permite al usuario especificar los puntos en que se desea que se interrumpa el programa. Un punto de interrupción es una etiqueta o un rango de etiquetas.

Formato:

BREAK

BREAK lista de puntos de interrupción

UNBREAK

UNBREAK lista de puntos de interrupción

Los datos en la lista de puntos de interrupción incluyen: etiqueta y etiqueta-etiqueta. Tienen las mismas características que para el comando TRACE.

Explicación:

Los comandos BREAK y UNBREAK pueden ser especificados antes de que el programa sea ejecutado o mientras está interrumpido. Cuando el programa está corriendo se interrumpe justo antes de ejecutar la instrucción cuya etiqueta consta en la lista de puntos de interrupción.

Cuando la ejecución se suspende debido a un punto de interrupción, la etiqueta de la instrucción a punto de ser ejecutada se imprime de la forma:

-BREAK etiqueta

Ante la continuación se imprime el signo > para indicar que puede introducirse un comando. Los comandos permitidos durante el período de interrupción se enumeran mas abajo

La ejecución se reanuda con el comando GO.

El comando BREAK, si no va acompañado de parametros, produce el listado de todos los puntos de interrupción en curso.

El comando UNBREAK si no va acompañado de parámetros elimina todos los puntos de interrupción en curso. Si incluye parámetros, el comando UNBREAK elimina todos los puntos de interrupción especificados por las etiquetas

de la lista de puntos de interrupción.

Comandos permitidos durante una interrupción:

ABORT
GO
SET
SHOW
BREAK/UNBREAK
TRACE/UNTRACE
LIST
DUMP
CREATE

5.3 Comando ABORT

El comando ABORT sólo está permitido en un período de interrupción. Aborta el programa suspendido y devuelve al usuario el control sobre todos los comandos.

Formato:

ABORT

Explicación:

Cuando se especifica el comando ABORT termina el período de interrupción y concluye la ejecución. El usuario puede ahora introducir cualquier comando legal durante la ejecución normal del programa BASIC.

5.4 Comando GO

El comando GO clausura el modo de depurado interactivo y reanuda el programa suspendido. Este comando sólo es válido en período de interrupción.

Formato:

GO

GO etiqueta

Explicación:

Un GO sin etiqueta reanuda el programa en la posición impresa cuando ocurrió la interrupción. Un GO con etiqueta reanuda la ejecución del programa en la dirección correspondiente, amén que dicha dirección transfiera el control fuera de la función en curso. El parámetro etiqueta no está permitido si la interrupción se produce a consecuencia de pulsar CTRL Y. En ese caso habrá que usar un GO sin etiqueta para reanudar la ejecución del programa.

5.5 Comando SHOW

El comando SHOW imprime los valores de los datos especificados; solamente está permitido en un período de interrupción.

Formato:

SHOW lista de datos

Esta lista puede incluir:

- Variables (numéricas o alfanuméricas)
- Elementos matriciales
- Matrices enteras (nombre(*)) para matrices unidimensionales; nombre(*,*) para matrices bidimensionales.

Explicación:

En el caso de una matriz se imprime el valor de todos sus elementos, excepto aquellos que no están definidos, en cuyo caso vienen señalados con la palabra UNDEFINED.

5.6 Comando SET

El comando SET permite al usuario asignarle a una variable un valor determinado. Este comando sólo es válido durante un período de interrupción.

Formato:

SET dato = constante

Los elementos a los que se les asigna un valor incluyen variables simples y matriciales, y sus características son las mismas que en el comando SHOW, con la excepción de que no pueden usarse asteriscos.

PARTE II

EL PROGRAMA ARCHALUM

1. ANALISIS DEL PROGRAMA "ARCHALUM"

En esta sección se estudia el funcionamiento del programa ARCHALUM, cuya estructuración en módulos in dependientes permite un cómodo análisis de cada una de sus funciones.

1.1 Características generales.

El programa ARCHALUM está concebido para gestionar el archivo de alumnos de una universidad, y aplicado concretamente a las características de la E.U.P. de Las Palmas. Se compone de un programa de control y trece módulos independientes entre si, que realizan las fun ciones de introducir, archivar, modificar, actualizar y listar datos sobre 45 ficheros indexados y un fiche ro principal, adaptándose a los requerimientos estatu tarios de matriculación. Tanto el programa de control como los distintos módulos están codificados en len guaje BASIC/3000, y debido a la longitud total del pro grama he optado por la práctica solución de concatenar módulos mediante el comando CHAIN, el cual permite lla mar sucesivamente a distintos módulos sin que los valo res asignados a las variables en un módulo se repitan en el siguiente.

Para proteger el acceso a los datos contra intrusos se ha asignado una clave a todos los módulos y fiche ros, que está contenida en la variable W\$ y que es fá- cilmente modificable usando el comando RENAME.

1.2 Módulo 1. Programa Menú.

Nombre del programa: ARCHALUM

Este módulo tiene la misión de seleccionar el comando deseado y transferir el control al módulo correspondiente, lo cual se realiza por concatenación, devolviéndose al final de cada módulo el control al programa menú. En éste se asigna a la variable W\$ el nombre de la clave de acceso a todos los módulos y ficheros del programa.

Explicación del organigrama.

-Bloque 1.

Selecciona el comando correspondiente al módulo que desea utilizarse.

Variables:

P\$: nº correspondiente al comando seleccionado.

-Bloque 2.

Explora si el comando seleccionado es el de salida. En caso afirmativo termina el programa. En caso negativo pasa al bloque 3.

-Bloque 3.

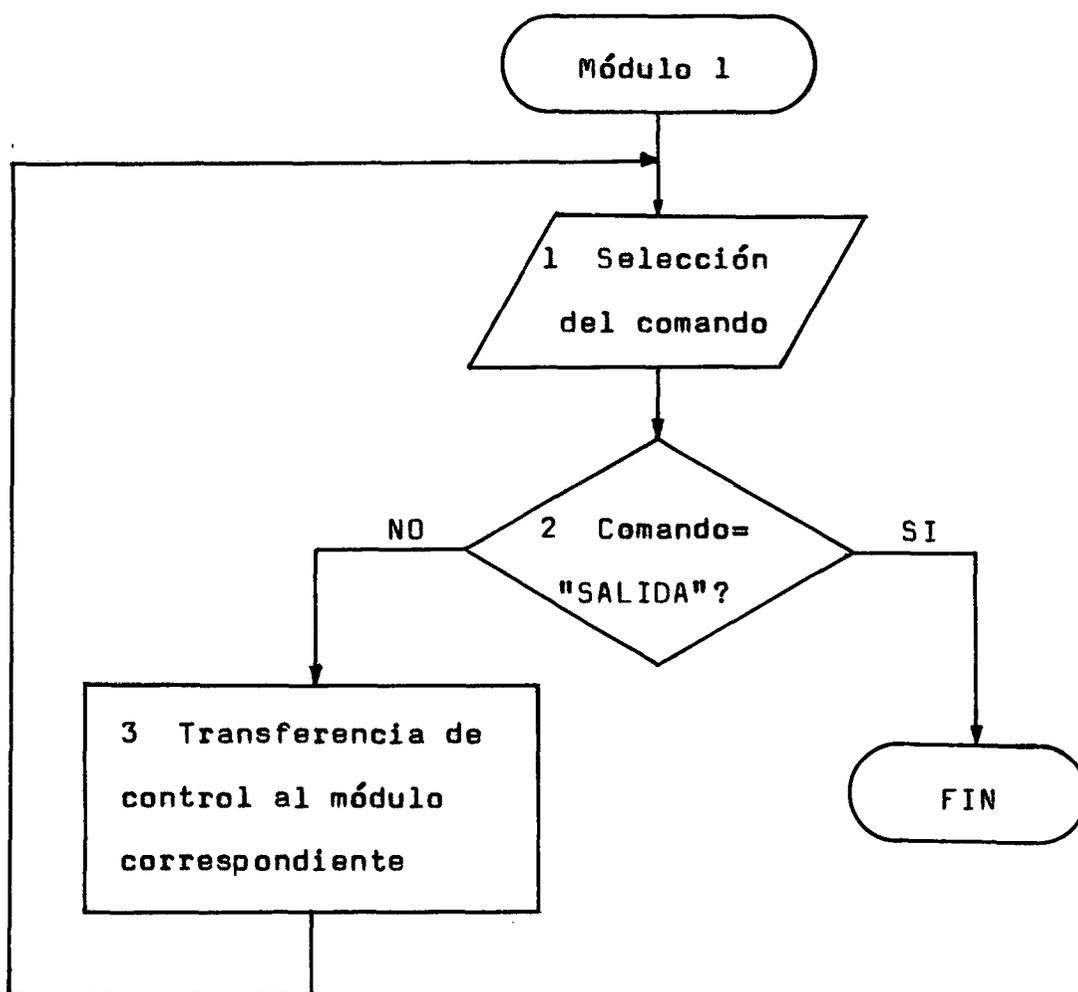
Transfiere el control al módulo seleccionado.

Variables:

P\$: nº del comando seleccionado.

W\$: clave de protección

-Organigrama.



1.3 Módulo 2. Matriculación de alumnos.

Nombre del programa: DELTA1

Este módulo permite la matriculación de alumnos en cualquiera de las carreras existentes.

Explicación del organigrama.

-Bloque 1.

Se selecciona la clave que indicará carrera, plan y curso o cursos a matricular.

Variables:

Z\$: contiene la clave de carrera curso y plan.

J : contiene el número de cursos +1.

-Bloque 2.

Se introducen los apellidos y el nombre del alumno a matricular.

Variables:

M\$(1) : primer apellido.

M\$(2) : segundo apellido.

M\$(3) : nombre.

-Bloque 3.

Explora la existencia del alumno en el fichero principal. En caso afirmativo pasa al bloque 6. En

caso negativo al bloque 4.

Variables:

M : indica el nº de grabación explorada en el fichero de expedientes.

W1 : W1=1 indica no localizado.

W1=0 indica localizado.

-Bloque 4.

Se introducen los datos personales del alumno estando la longitud de la variable de entrada determinada por el formato de pantalla.

Variables:

M\$(4) a M\$(22) : contienen los datos personales.

-Bloque 5.

Se localiza la posición del alumno en fichero temporal ("TEMP") y se inicializa la grabación corres-pondiente al orden alfabético que ocupe el alumno en el fichero, introduciendo los datos personales e inicializando los datos académicos.

Variables:

P1 : indica el nº de grabaciones existentes en el fichero temporal ("TEMP").

P2 : indica el nº que le corresponde al fichero temporal.

P3 : P3=1 indica desde dónde ha sido llamada la subrutina 1.

P4 : indica la primera grabación del fichero que contiene datos estandar de búsqueda.

P5 : indica el nº que le corresponde al fichero temporal.

R1 : R1=1 indicará a bloque posteriores que estamos trabajando con fichero temporal.

R0 : indica la posición del alumno dentro del fichero temporal.

-Bloque 6.

Se imprimen los datos personales del alumno.

Variables:

M\$(4) a M\$(22) : contienen los datos personales.

-Bloque 7.

Se explora la posibilidad de modificar datos personales. En caso negativo pasa al bloque 10. En caso afirmativo pasa al bloque 8.

Variable:

W1 : W1=2 indica petición de modificar datos.

-Bloque 8.

Se introducen los datos personales que se desean modificar usando las funciones UPS\$ y DEB\$ que convierten los datos en mayúsculas y eliminan los posibles espacios en blanco exteriores.

Variables:

M\$(4) a M\$(22) : contienen los datos personales modificados y no modificados.

-Bloque 9.

Introducción de los datos personales utilizados en el fichero de expedientes.

Variables:

M : indica la posición que ocupa el alumno en el fichero de expedientes.

M\$(1) a M\$(22) : datos del alumno que pueden contener modificación.

-Bloque 10.

Localiza la posición del alumno en fichero tipo "C". Se explora primeramente si el fichero tipo "C" a utilizar contiene grabaciones vacías. En caso negativo se realiza su ampliación en 25 grabaciones más. A continuación se realiza la localización.

Variables:

P1 : indica el nº de grabaciones existentes en fichero tipo "C".

P2 : indica el nº correspondiente al fichero tipo "C".

P3 : indica desde donde ha sido llamada la subrutina 1.

P4 : indica el primer record del fichero tipo "C" que contiene datos estándar de búsqueda.

P5 : indica el nº correspondiente al fichero tipo "C".

N2 : indica el nº de especialidades existentes en el curso.

N4 : indica el nº total de grabaciones del fichero tipo "C".

-Bloque 11.

Se seleccionan las asignaturas a matricular. El nº máximo de asignaturas es 15 por curso.

Variables:

W2 : nº total de asignaturas matriculadas.

W3(1) a W3(15) : números correspondientes a las asignaturas matriculadas.

-Bloque 12.

Reintroducción de los datos académicos (modificados o no) en los ficheros correspondientes.

Variables:

R0 : indica la posición que ocupa la grabación del alumno en el fichero tipo "C".

R2 : indica la posición que ocupa la grabación del alumno en el fichero temporal o en el fichero de expedientes.

P2 : indica el fichero tipo "C" que se está utilizando.

-Bloque 13.

Explora si existen más cursos por matricular. En caso afirmativo vuelve al bloque 10. En caso negativo pasa al bloque 14.

Variable:

K2 : indica el nº de cursos en los cuales se ha realizado matriculación.

-Bloque 14.

Explora si ha terminado el comando. En caso afirmativo pasa al bloque 17. En caso negativo sigue con el bloque 15.

Variable:

R\$: contiene la decisión de continuar o no.

-Bloque 15.

Explora si el nº de alumnos contenidos en el fichero temporal ("TEMP") es igual a 100. En caso negativo vuelve al bloque 1. En caso afirmativo pasa al bloque 16.

Variables:

N : nº de grabaciones en el fichero temporal.

N5 : nº de grabaciones utilizadas en el fichero de expedientes.

N6 : nº de grabaciones disponibles en el fichero de expedientes.

-Bloque 16.

Explora si las 100 grabaciones existentes en el fichero temporal caben en el fichero de expedientes. En caso de que no cupieran aumentaría la longitud del fichero de expedientes en 1000 grabaciones. A continuación se vuelcan los datos del fichero temporal en el fichero de expedientes y se vuelve al bloque 1.

Variables:

N : nº de grabaciones existentes en el fichero temporal.

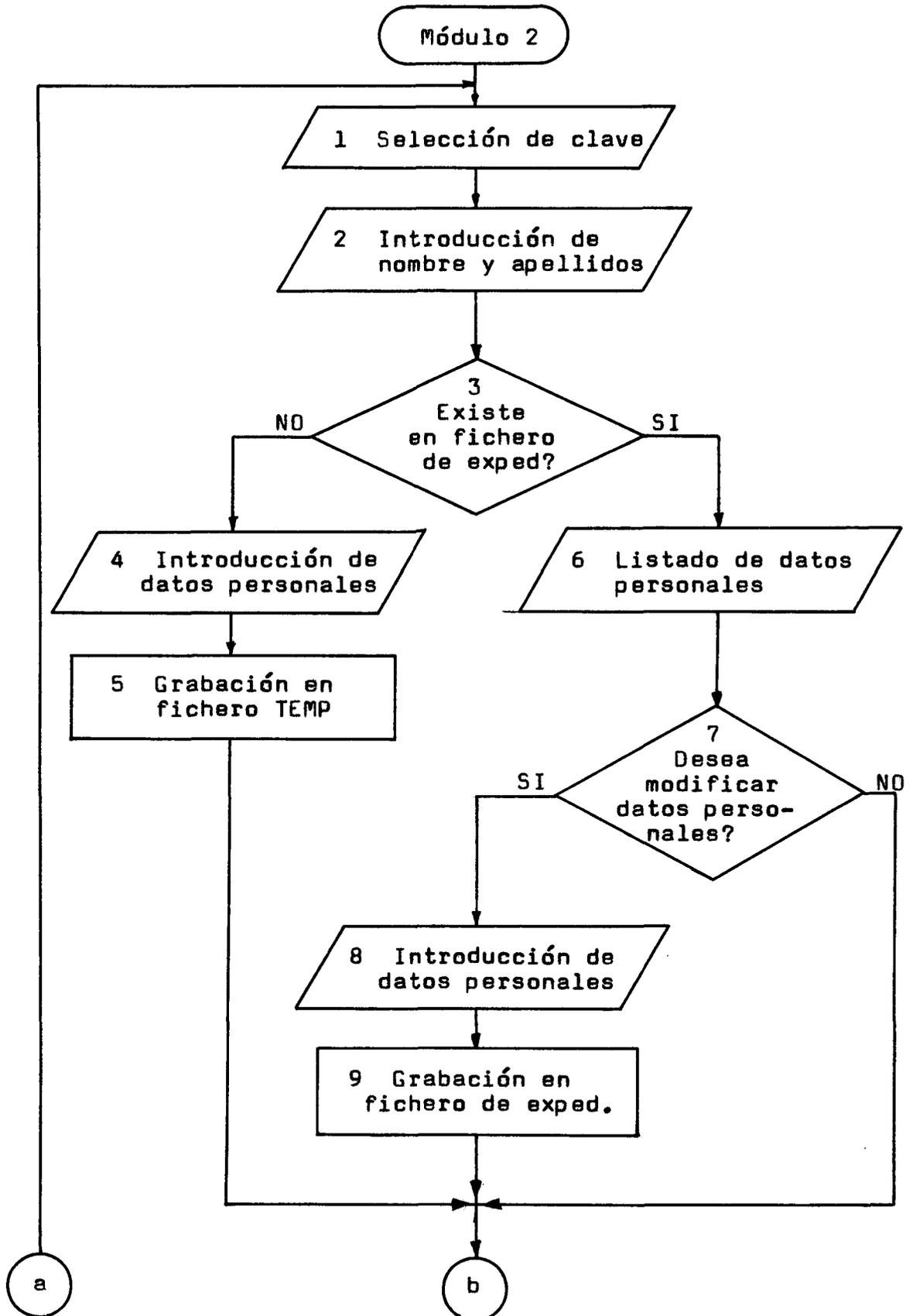
N5 : nº de grabaciones utilizadas en el fichero de expedientes.

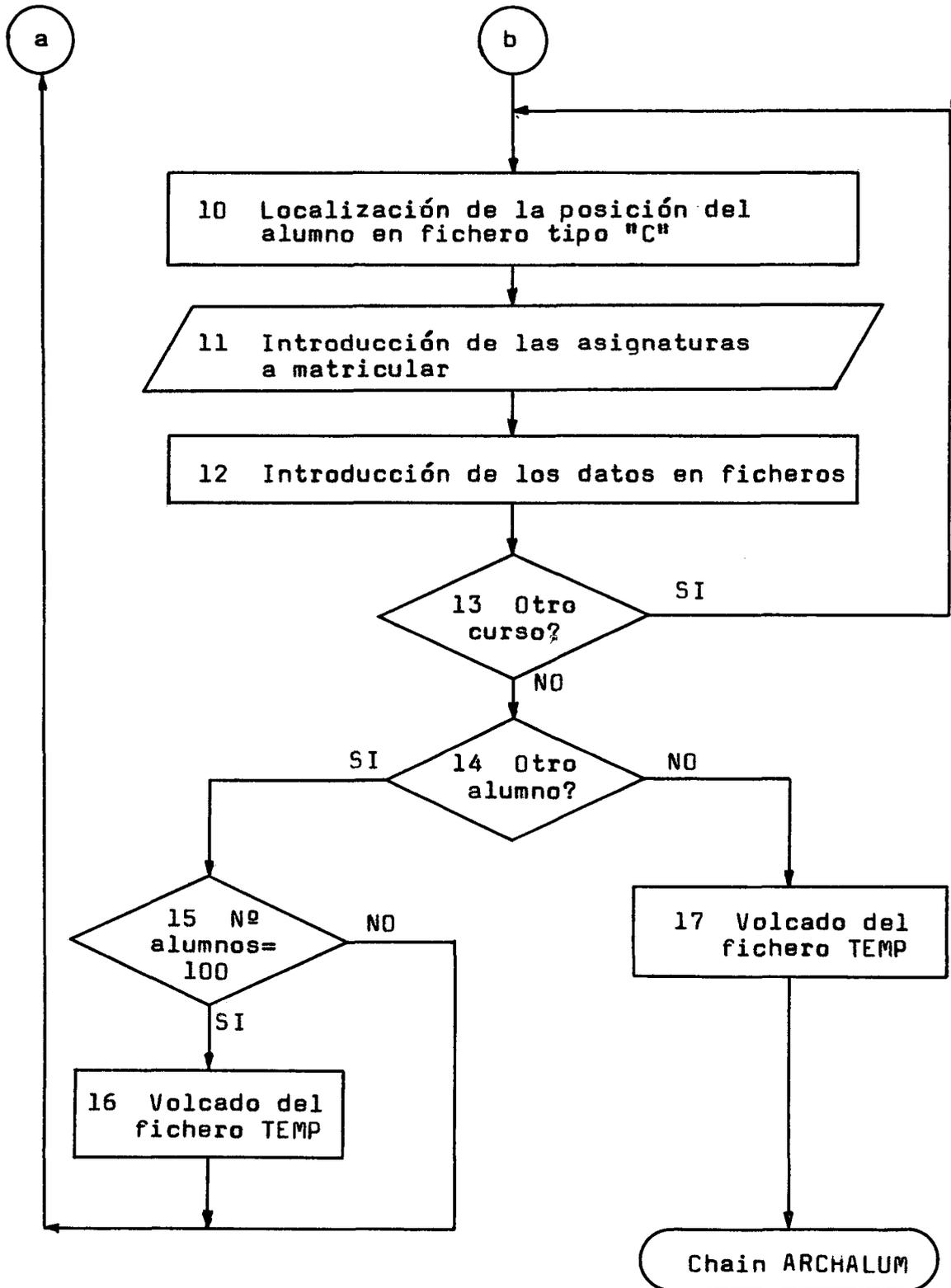
P6 : contiene el nº correspondiente al fichero de expedientes. Puede valer 6 u 8 según se trate del fichero de expedientes no ampliado o ampliado.

-Bloque 17.

Realiza exactamente la misma función que el bloque 16 y utiliza las mismas variables, con la salvedad de que no vuelve al bloque 1, sino que devuelve el control al programa menú.

-Organigrama.





1.4 Módulo 3. Renuncia de convocatorias.

Nombre del programa: DELTA2

Este módulo permite realizar la renuncia de convocatorias de cualquier asignatura (o asignaturas) matriculada.

Explicación del organigrama.

-Bloque 1.

Selección del alumno introduciendo apellidos y nombre.

Variables:

A\$(1) : primer apellido.

A\$(2) : segundo apellido.

A\$(3) : nombre.

-Bloque 2.

Localiza la posición de la grabación correspondiente al alumno, en el fichero de expedientes.

Variable:

M : indica la posición actual de búsqueda.

-Bloque 3.

Explora si el alumno ha pedido ser localizado en el fichero de expedientes. En caso negativo vuelve

al bloque 1. En caso afirmativo sigue con el bloque 4.

-Bloque 4.

Este bloque se encarga de leer en la grabación cuáles son las asignaturas matriculadas actualmente y produce su listado en pantalla.

Variables:

H\$: indica el número en que están ordenadas las asignaturas.

F2\$: contiene el nombre de la asignatura.

A\$(1) : A\$(1)="X" indica matriculado de la convocatoria de febrero.

A\$(2) : A\$(2)="X" indica matriculado de la convocatoria de junio.

A\$(3) : A\$(3)="X" indica matriculado de la convocatoria de septiembre.

V\$(1) a V\$(3) : indica si las convocatorias anteriores se hallan ya calificadas.

-Bloque 5.

Este bloque permite al usuario seleccionar las asignaturas de las que desea renunciar a alguna convocatoria.

Variables:

C(1) a C(15) : contiene los números correspondientes a las asignaturas de renuncia.

-Bloque 6.

Renuncia de las convocatorias de cada una de las asignaturas seleccionadas por el usuario.

Variables:

Z\$(1) : Z\$(1)="X" indica renuncia de la convocatoria de febrero.

Z\$(2) : Z\$(2)="X" indica renuncia de la convocatoria de junio.

Z\$(3) : Z\$(3)="X" indica renuncia de la convocatoria de septiembre.

-Bloque 7.

Introduce los datos de renuncia en fichero de expedientes, actualizando el nº de convocatorias.

Variables:

F4\$: contiene los datos de las convocatorias de todas las asignaturas del curso.

T2 : indica el nº de convocatorias en que se ha lla matriculado el alumno, para cada asigna tura

M : nº de la grabación que ocupa el alumno en el fichero de expedientes.

-Bloque 8.

Introduce los datos de renuncia en los ficheros tipo "C".

Variables:

F3\$: contiene el nombre del fichero tipo "C" so
bre el que se está grabando.

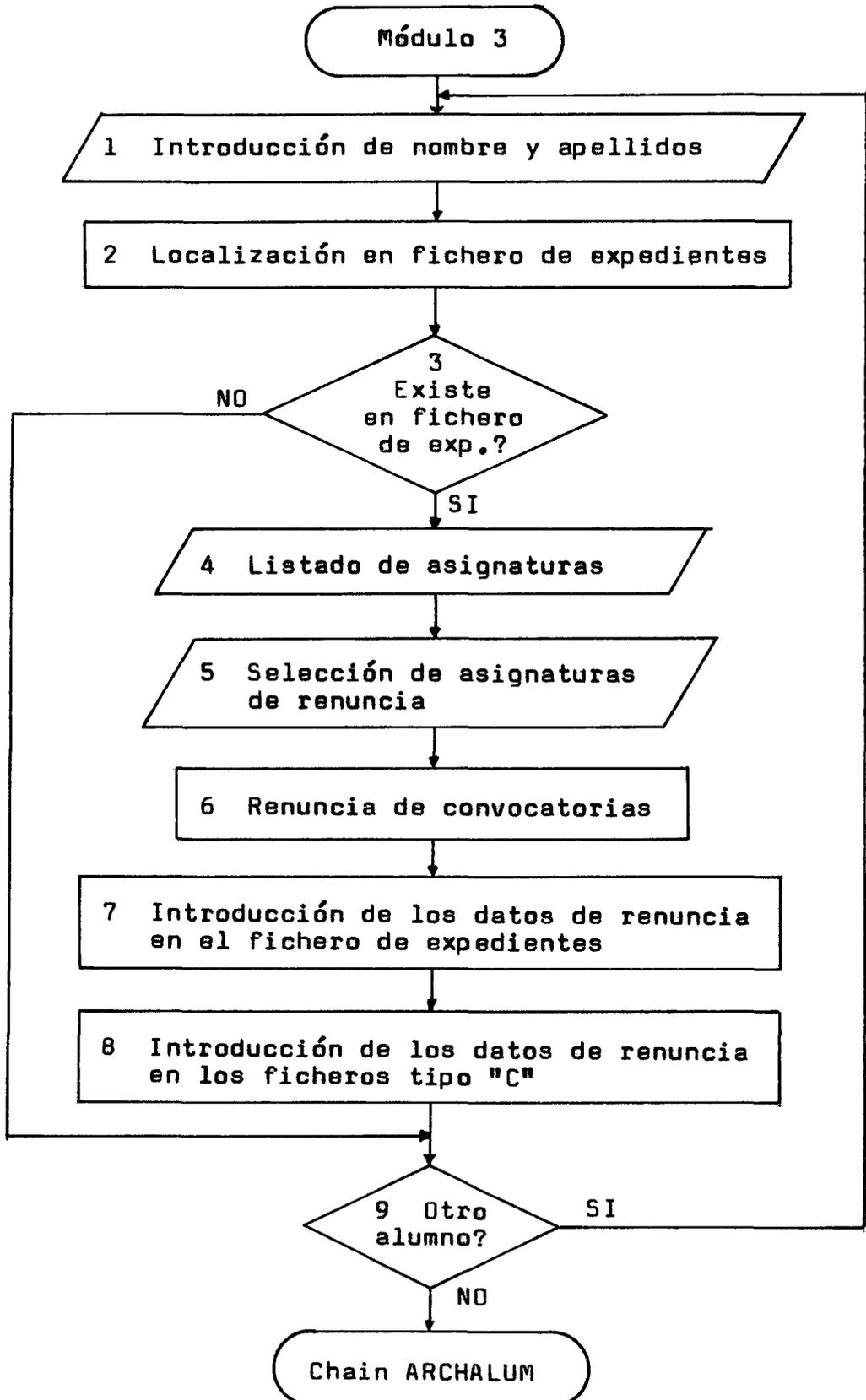
F4\$: contiene los datos de las convocatorias de
todas las asignaturas del curso.

M : nº de la grabación que ocupa el alumno en
el fichero tipo "C".

-Bloque 9.

Explora si ha terminado el comando. En caso nega
tivo vuelve al bloque 1. En caso positivo devuelve
el control al programa menú.

-Organigrama



1.5 Módulo 4. Cambio de convocatorias

Nombre del programa: DELTA3

Este módulo permite realizar el cambio de convocatorias de cualquier asignatura o asignaturas matriculada.

Explicación del organigrama.

-Bloque 1.

Selección del alumno introduciendo apellidos y nombre.

Variables:

A\$(1) : primer apellido.

A\$(2) : segundo apellido.

A\$(3) : nombre.

-Bloque 2.

Localiza la posición de la grabación correspondiente al alumno en el fichero de expedientes.

Variable:

M : indica la posición actual de búsqueda en el fichero de expedientes.

-Bloque 3.

Explora si el alumno ha podido ser localizado en

el fichero de expedientes. En caso negativo vuelve al bloque 1. En caso afirmativo sigue con el bloque 4.

-Bloque 4.

Este bloque se encarga de leer en la grabación cuáles son las asignaturas matriculadas actualmente y produce su listado en pantalla.

Variables:

H\$: indica el número en que están ordenadas las asignaturas.

F2\$: contiene el nombre de las asignaturas.

A\$(1) : A\$(1)="X" indica matriculado de la convocatoria de febrero.

A\$(2) : A\$(2)="X" indica matriculado de la convocatoria de junio.

A\$(3) : A\$(3)="X" indica matriculado de la convocatoria de septiembre.

V\$(1) a V\$(3) : indican si las convocatorias anteriores se hallan ya calificadas.

-Bloque 5.

Este bloque permite al usuario seleccionar las asignaturas de las que desea cambiar alguna convocatoria.

Variables:

C(1) a C(15) : contiene los números correspon-

dientes a las asignaturas de renuncia.

-Bloque 6.

Cambio de convocatorias en cada una de las asignaturas seleccionadas por el usuario.

Variables:

Z\$(1) : contiene la convocatoria que pretende darse de baja.

Z\$(2) : contiene la convocatoria que pretende darse de alta.

-Bloque 7.

Introduce los datos de cambio de convocatorias en el fichero de expedientes.

Variables:

F4\$: contiene los datos de las convocatorias de todas las asignaturas del curso.

M : indica el nº de la grabación que ocupa el alumno en el fichero de expedientes.

-Bloque 8.

Introduce los datos de renuncia en los ficheros tipo "C".

Variables:

F3\$: contiene el nombre del fichero tipo "C" sobre el que se está grabando.

F4\$: contiene los datos de las convocatorias

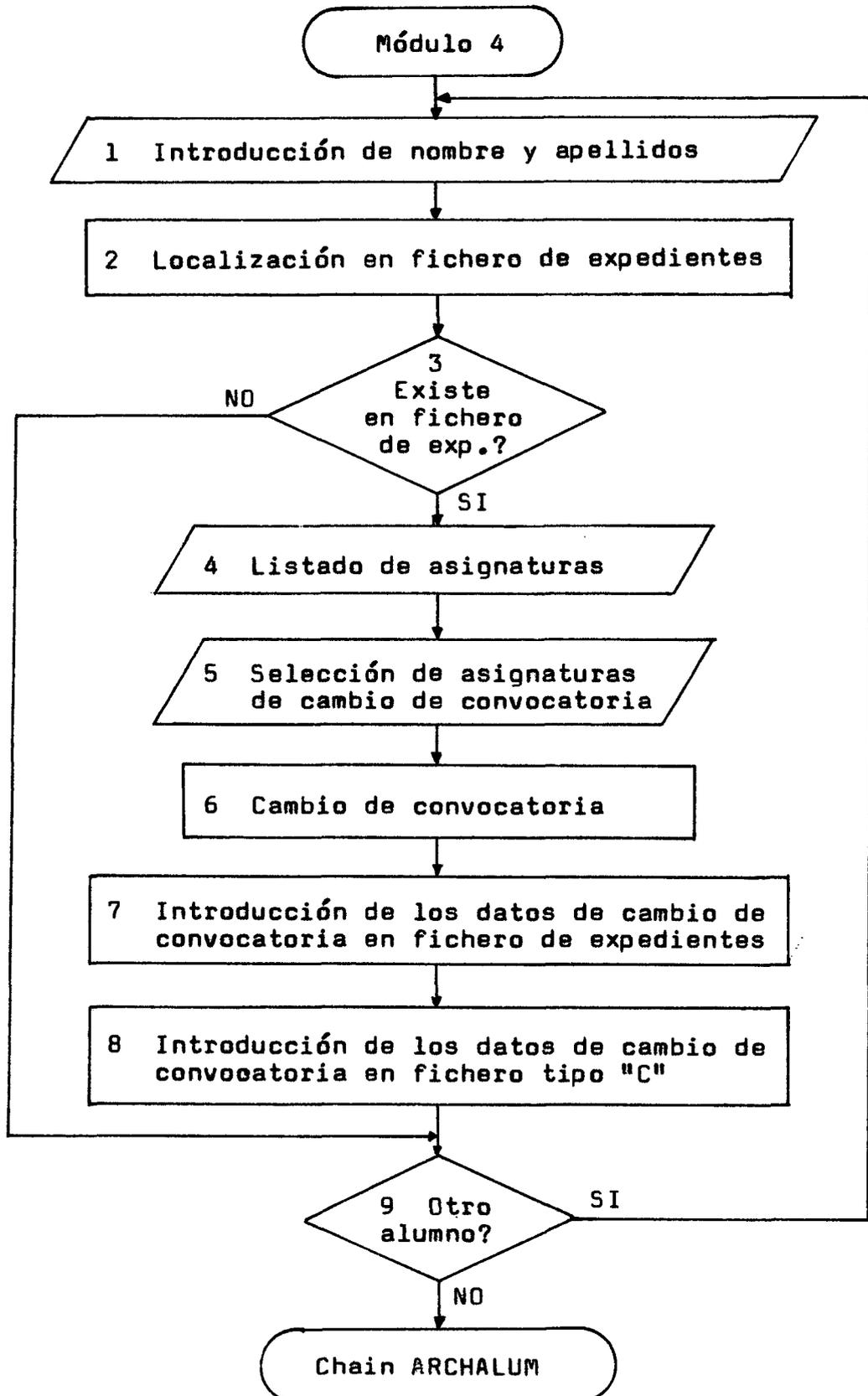
de todas las asignaturas del curso.

M : indica el nº de la grabación que ocupa el alumno en el fichero tipo "C".

-Bloque 9.

Explora si ha terminado el comando. En caso negativo vuelve al bloque 1. En caso positivo devuelve control al programa menú.

-Organigrama.



1.6 Módulo 5. Introducción de notas.

Nombre del programa: DELTA4

Este módulo permite introducir las notas de una o varias asignaturas, en la convocatoria seleccionada, para todos los alumno matriculados de dicha o dichas asignaturas

Explicación del organigrama.

-Bloque 1.

Selecciona la clave que indicará carrera, plan, curso y convocatoria que desea calificarse.

Variable:

Z\$: contiene la clave de carrera, curso, plan y convocatoria.

-Bloque 2.

Listado de todas las asignaturas existentes en el curso seleccionado.

Variables:

H\$: indica el número en que están ordenadas las asignaturas.

F3\$: contiene los nombres de las asignaturas listadas.

I : indica el nº de asignaturas distintas exis-

tentes en todas las especialidades del curso seleccionado.

-Bloque 3.

Este bloque permite al usuario seleccionar las asignaturas que desea calificar.

Variables:

C(1) a C(50) : contienen los números correspondientes a las asignaturas a calificar.

-Bloque 4.

Este bloque realiza el listado de los alumnos matriculados de la asignatura seleccionada en curso, correspondiente a la convocatoria seleccionada por clave, y permite la introducción de las calificaciones.

Variables:

L1 : nº de asignaturas seleccionadas.

A\$(1) : primer apellido del alumno que se está calificando.

A\$(2) : segundo apellido.

A\$(3) : nombre.

K5 : selecciona la línea de la pantalla que le corresponde al alumno por orden alfabético.

V\$: calificación introducida.

-Bloque 5.

Introducción de los datos académicos en fichero tipo "C", actualizando el nº de convocatorias si se ha aprobado la primera de dos.

Variables:

T2 : nº de convocatorias

T2\$: datos académicos codificados.

K1 : nº de la grabación del fichero tipo "C" correspondiente al alumno calificado.

-Bloque 6.

Este bloque explora si la convocatoria a calificar corresponde a trabajo fin de carrera.

Variables:

F9\$: F9\$="S" indica matriculado de proyecto.

-Bloque 7.

Introducción de los datos académicos en fichero de expedientes y asignación a Q del valor -1, para indicar que consta como alumno de carrera terminada.

Variables:

M : posición que ocupa el alumno en el fichero de expedientes.

V\$: calificación del proyecto.

Q : variable que indica si el alumno tiene la carrera terminada.

-Bloque 8.

Busqueda de la siguiente asignatura. En caso afirmativo vuelve al bloque 4. En caso negativo sigue al bloque 9.

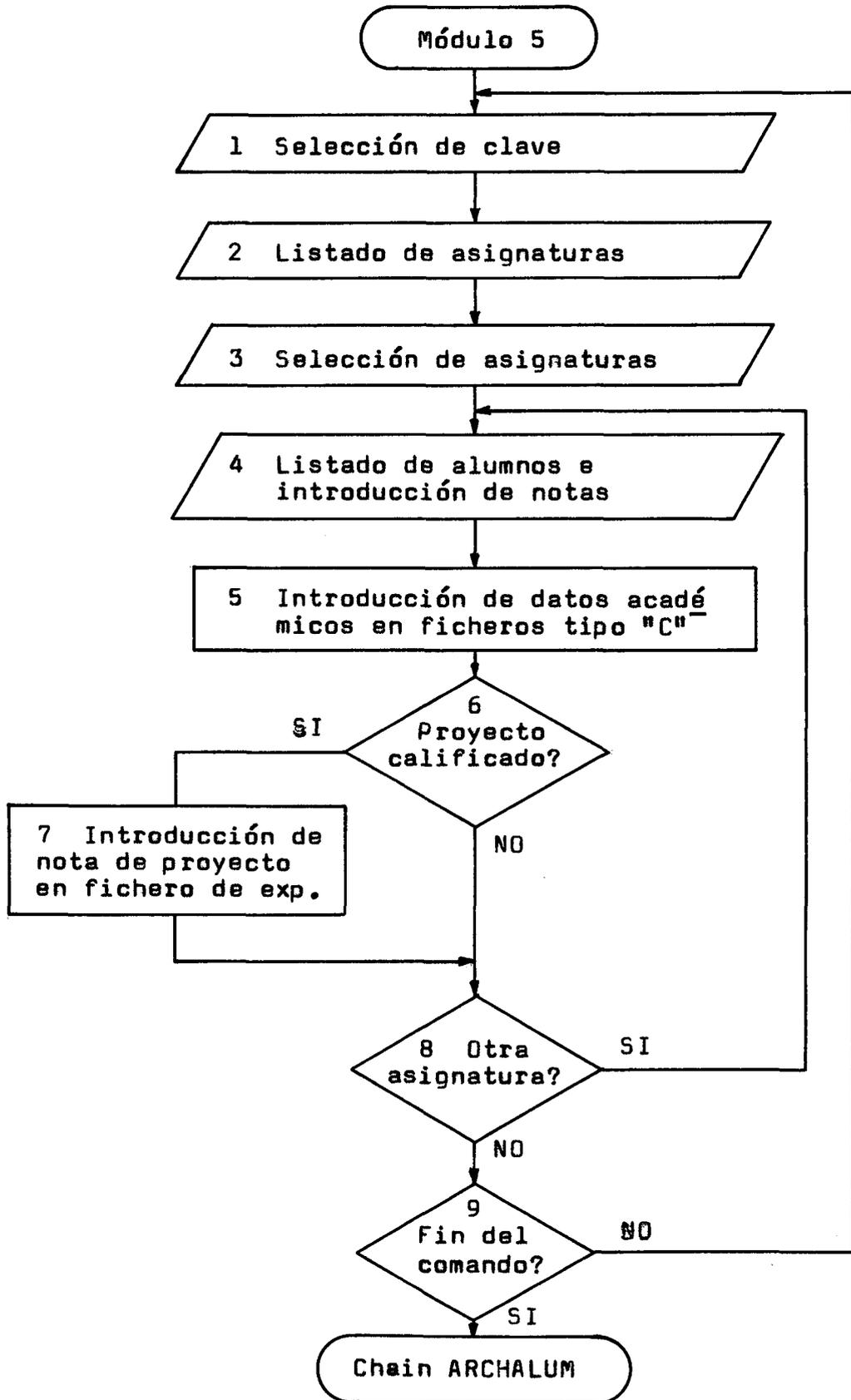
Variable:

K : indica el nº de asignaturas que han sido calificadas en el transcurso del comando.

-Bloque 9.

Explora si ha terminado el comando. En caso negativo vuelve al bloque 1. En caso afirmativo devuelve control al programa menú.

-Organigrama.



1.7 Modulo 6. Volcado de ficheros.

Nombre del programa: DELTA5

Debido a las características de los ficheros tipo "C" estudiados en el apartado referente a la estructura de ficheros, será necesario un volcado anual, que se efectuará justo antes de la matriculación para el nuevo curso y que tiene como objetivo dejar libres los ficheros tipo "C" para los nuevos alumnos matriculados. Para que este volcado sea global, es decir para todos los cursos y todas las carreras, deberá verificarse primero el que todas las asignaturas y convocatorias estén calificadas.

Explicación del organigrama.

-Bloque 1.

Exploración de cada uno de los ficheros tipo "C" para comprobar que todas las asignaturas matriculadas están calificadas.

Variables:

F4\$: contiene la codificación de las convocatorias matriculadas.

F9\$: contiene la codificación de las calificaciones.

-Bloque 2.

Explora la variable que indica si todos los ficheros están calificados. En caso negativo pasa al bloque 3. En caso afirmativo pasa al bloque 4.

Variable:

J : J=0 indica todos los ficheros calificados.

J=1 indica que quedan asignaturas por calificar.

-Bloque 3.

Este bloque realiza el listado de las asignaturas por convocatoria, carrera, plan y curso que están sin calificar.

Variables:

F2\$: nombre de la asignatura.

C4\$: convocatoria no calificada.

-Bloque 4.

Este bloque realiza el volcado de los datos académicos que se encuentran en el fichero tipo "C", hacia el fichero de expedientes. Para ello se localiza primeramente la posición de cada uno de los alumnos en el fichero de expedientes y se le introducen las calificaciones, añadiendo las convocatorias no presentadas en el caso de que las hubiera.

Variables:

M : posición del alumno en el fichero de expe-

dientes

F4\$: datos de convocatorias del fichero tipo "C"

F9\$: datos de las calificaciones del fichero tipo "C"

-Bloque 5.

Purga de los datos de los alumnos del fichero tipo "C" volcado.

Variables:

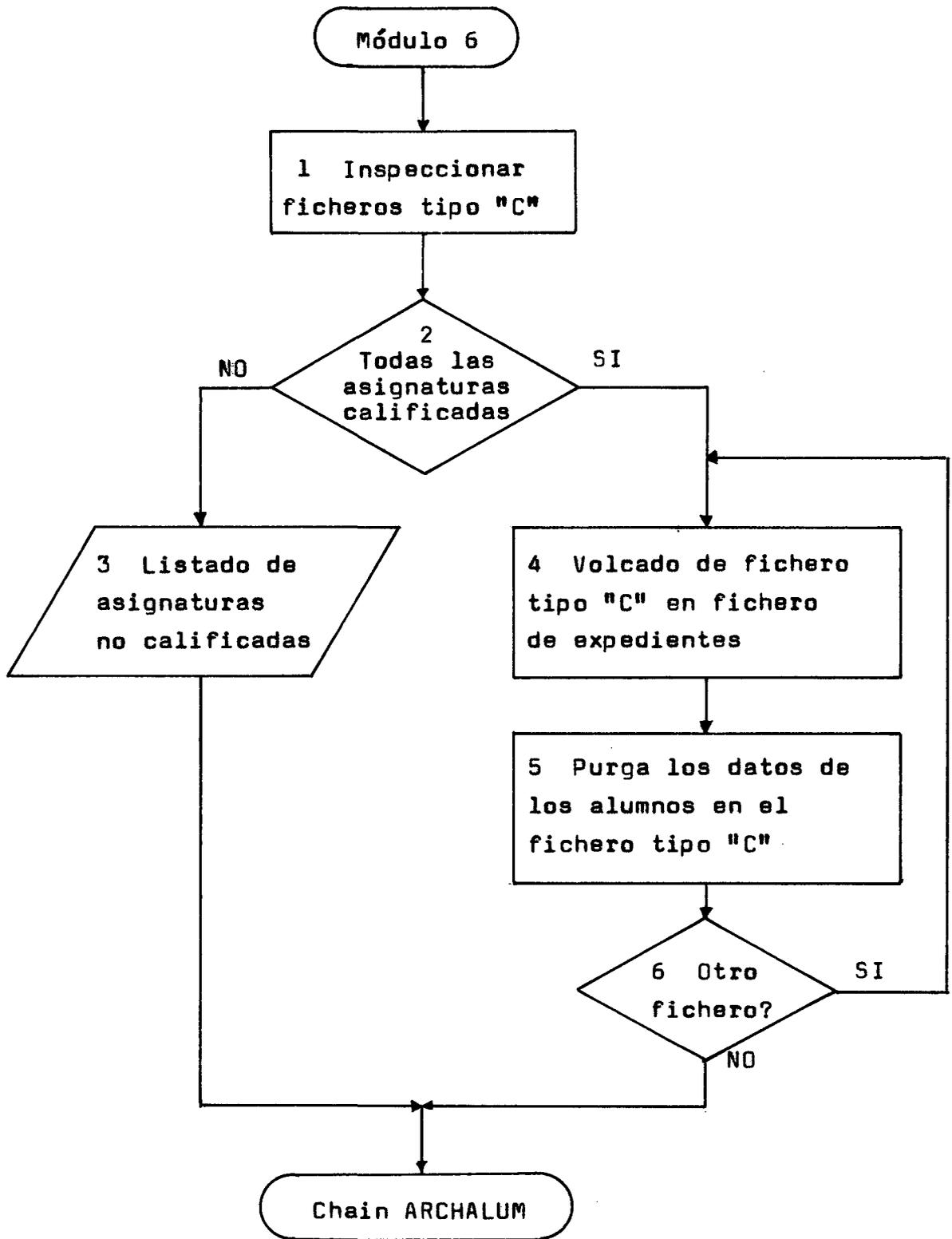
K3 : indica cada una de las grabaciones ocupadas por los alumnos.

N2 : N2+1 inicializa el nº de grabaciones utilizadas en el fichero tipo "C".

-Bloque 6.

Explora si todos los ficheros han sido volcados. En caso negativo busca el siguiente fichero. En caso afirmativo devuelve el control al programa menú.

-Organigrama.



1.8 Módulo 7. Listado por carrera.

Nombre del programa: DELTA6

Este módulo permite listar por orden alfabético, el nombre y apellidos de todos los alumnos matriculados en una carrera, apareciendo cada uno de los alumnos una sola vez, aunque está matriculado simultáneamente en varios cursos. Permite además realizar el listado entre unos límites alfabéticos determinados.

Explicación del organigrama.

-Bloque 1.

Este bloque selecciona la carrera sobre la cual desea efectuarse el listado.

Variables:

H\$: número correspondiente a la carrera.

C1\$: nombre de la carrera.

-Bloque 2.

Permite la selección de límites de listado alfabético. Los límites estandar son "" y "ZZ".

Variables:

L1\$: contiene el límite inferior.

L2\$: contiene el límite superior.

-Bloque 3.

Localización de alumnos. Realiza la búsqueda entre todas las asignaturas, para determinar el alumno que ocupa la siguiente posición por orden alfabético.

Variables:

S(1) a S(7) : punteros de los ficheros tipo "C" referentes a la carrera seleccionada, que indican el nº máximo de grabaciones que existen en cada uno de los ficheros.

S1(1) a S1(7) : punteros de los ficheros tipo "C" referentes a la carrera seleccionada, que indican la última grabación leída del fichero.

M\$(1) a M\$(21) : indica los dos apellidos y el nombre para cada uno de los 7 ficheros referentes a la carrera seleccionada.

-Bloque 4.

Realiza el listado de los alumnos encontrados en la carrera entre los límites seleccionados.

Variables:

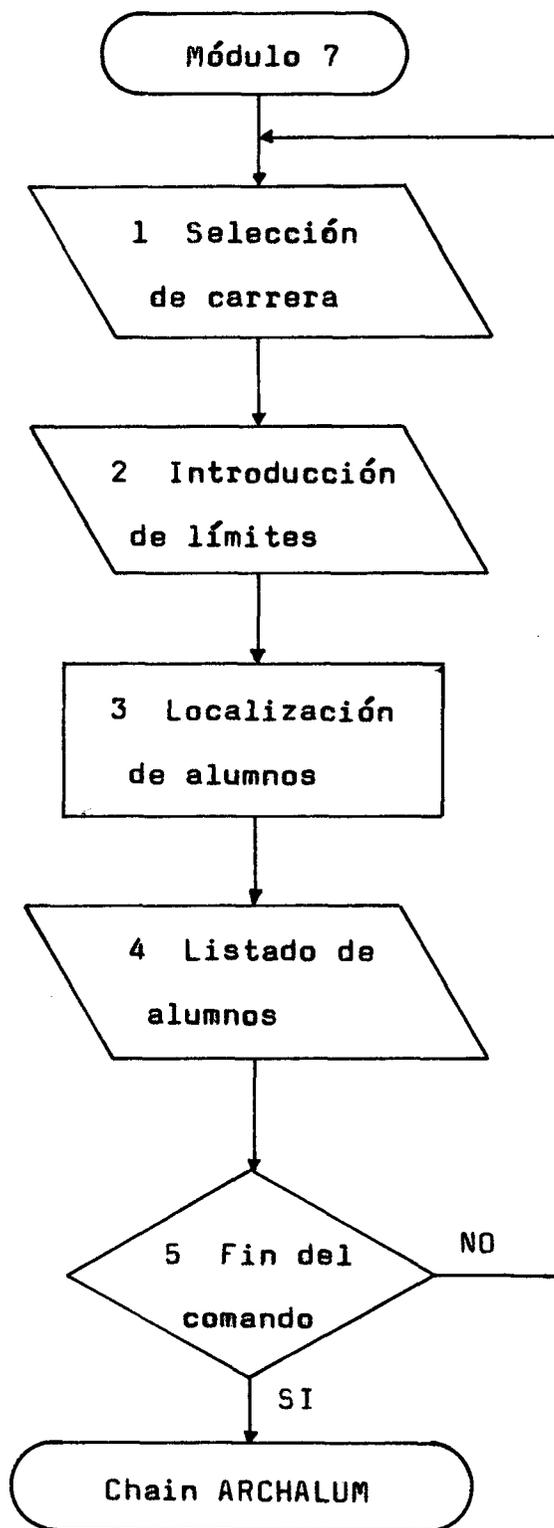
D5\$(1) : primer apellido.

D5\$(2) : segundo apellido.

D5\$(3) : nombre.

-Bloque 5.

Explora si ha terminado el comando. En caso negativo vuelve al bloque 1. En caso afirmativo devuelve el control al programa menú.

-Organigrama.

1.9 Módulo 8. Listado por curso.

Nombre del programa: DELTA7

Este módulo permite listar alfabéticamente todos los alumnos de uno o todos los cursos pertenecientes a un plan de estudios determinado.

Explicación del organigrama.

-Bloque 1.

Selección de clave que indicará la carrera, el plan de estudios y el curso (o todos) que desea listarse.

Variable:

Z\$: contiene la clave correspondiente a carrera plan y curso.

-Bloque 2.

Localización del fichero. En función de la clave seleccionada se abren los ficheros correspondientes.

Variable:

D : sirve para explorar si ha sido seleccionado un curso o todos.

-Bloque 3.

Listado de alumnos. Los alumnos aparecen listados

alfabéticamente por curso.

Variables:

D5\$(1) : primer apellido

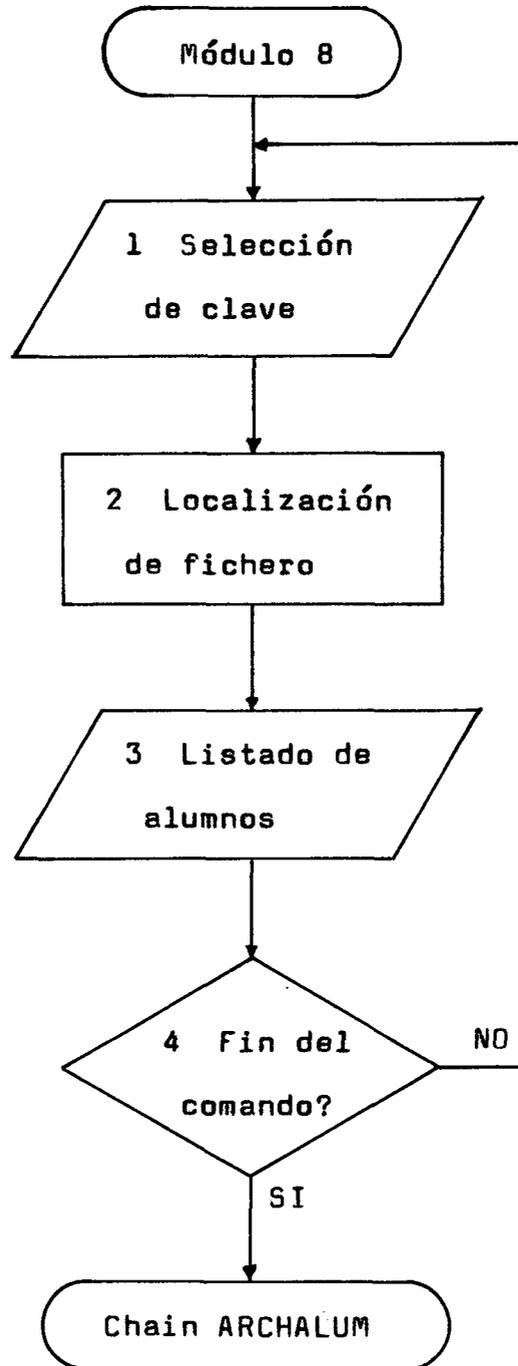
D5\$(2) : segundo apellido

D5\$(3) : nombre

-Bloque 4

Explora si ha terminado el comando. En caso negativo vuelve al bloque 1. En caso afirmativo devuelve control al programa menú.

-Organigrama.



1.10 Módulo 9. Listado por asignatura.

Nombre del programa: DELTA8

Este módulo permite obtener el listado alfabético de alumnos, por asignatura y especialidad, especificando convocatoria, para cada uno de los cursos pertenecientes a un mismo plan y carrera.

Explicación del organigrama.

-Bloque 1.

Selección de clave que indicará carrera, plan y curso (o cursos) sobre los que se desea obtener el listado de alumnos.

Variable:

Z\$: contiene la clave que selecciona carrera
plan y curso.

-Bloque 2.

Listado de asignaturas. Realiza el listado de todas las asignaturas distintas existentes en las distintas especialidades.

Variables:

F(1) a F(50) : cada variable del vector indica el nº de veces que aparece la asignatura correspondiente a esa variable, en las distin

tas especialidades.

J1(50,10) : es una matriz cuyo primer índice designa una asignatura, y cuyo segundo índice señala cada una de las especialidades en que aparece.

J2(50,10) : es una matriz cuyo primer índice designa una asignatura y cuyo segundo índice señala la posición que ocupa dicha asignatura dentro de su especialidad.

H\$: número correspondiente a cada una de las asignaturas ordenadas.

F3\$: nombre de la asignatura

I : nº total de asignaturas distintas encontradas.

-Bloque 3.

Selección de asignaturas por el usuario. Pueden seleccionarse desde una a todas las asignaturas listadas.

Variables:

C(1) a C(50) : contiene el número que designa cada una de las asignaturas seleccionadas.

-Bloque 4.

Listado de alumnos y convocatorias. Para cada uno de los alumnos se leen sus datos de convocatoria, comprobando que está matriculado de la asignatura.

natura seleccionada. En caso afirmativo se imprimen sus datos. De lo contrari se pasa al siguiente alumno.

Variables:

T2\$: contiene los datos codificados de las convocatorias de todas las asignaturas del curso.

T\$: contiene las fechas de convocatorias de la asignatura seleccionada.

H\$: contiene el nº de convocatorias asignadas a la asignatura seleccionada.

M\$(1) : primer apellido.

M\$(2) : segundo apellido.

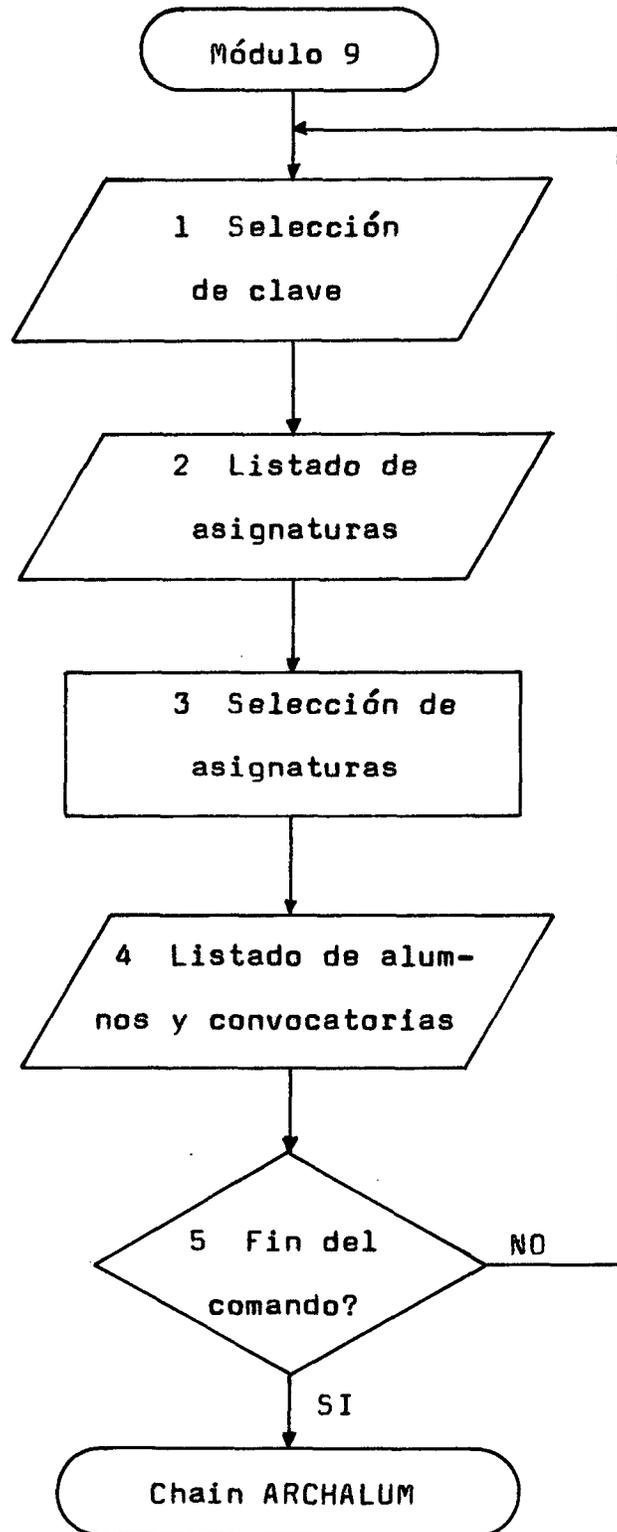
M\$(3) : nombre.

D\$: datos de la convocatoria.

-Bloque 5.

Explora si ha terminado el comando. En caso negativo vuelve al bloque 1. En caso afirmativo devuelve el control al programa menú.

+Organigrama.



1.11 Módulo 10. Tabla por curso.

Nombre del programa: DELTA9

Este comando obtiene una tabla estadística por edad sexo y matriculación, de uno o varios cursos pertenecientes a un mismo plan de estudios. Permite visualizar conjuntamente el número de alumnos por edades indicando si son nuevos o repetidores y su sexo, así como el total de alumnos pertenecientes a un curso determinado.

Explicación del organigrama.

-Bloque 1.

Selección de clave que indicará carrera, plan y curso (o cursos) sobre los que se desea obtener el listado tabulado de datos.

Variable:

Z\$: contiene la clave que selecciona carrera, plan y curso.

-Bloque 2.

Este bloque analiza la edad, sexo y matriculación de cada uno de los alumnos de un curso, incrementando la variable matricial designada por el índice correspondiente.

Variables:

M1 : su valor indica el orden que le corresponde a cada una de las edades.

T1\$(1,6;1) : indica si el alumno es repetidor o no.

M\$(4) : indica si el alumno es varón o hembra.

J1(M1,1) : el índice 1 designa a los varones nuevos.

J1(M1,2) : el índice 2 designa a los varones repetidores.

J1(M1,3) : el índice 3 designa a las hembras nuevas.

J1(M1,4) : el índice 4 designa a las hembras repetidoras.

J1(M1,5) : el índice 5 designa al total de varones.

J1(M1,6) : el índice 6 designa al total de las hembras.

-Bloque 3.

Listado tabular. Presenta en pantalla los datos estadísticos del curso.

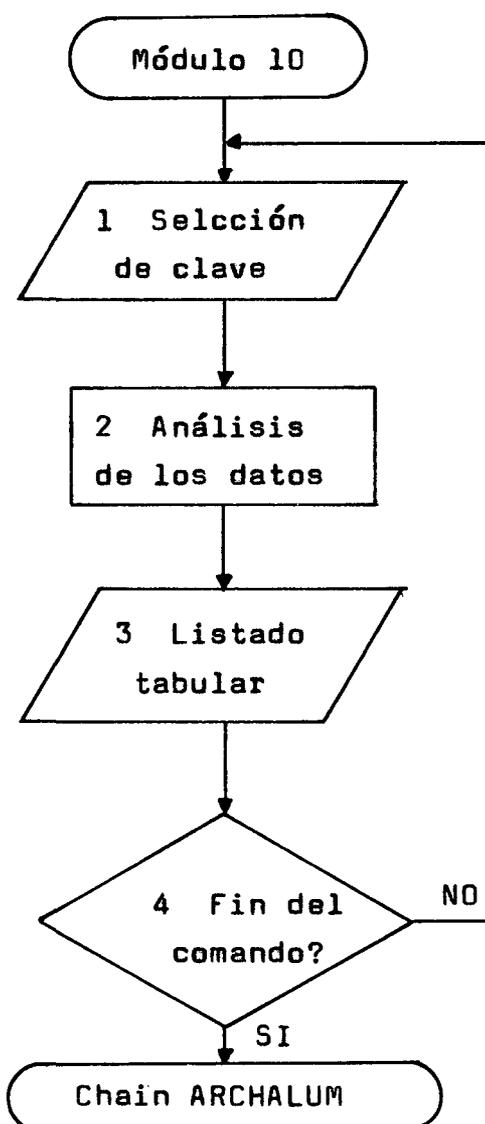
Variables:

F5\$(1) a F5\$(7) : contienen los datos estadísticos referentes al curso.

-Bloque 4.

Se explora si ha terminado el comando. En caso negativo se vuelve al módulo 1. En caso afirmativo se devuelve el control al programa menú.

-Organigrama.



1.12 Módulo 11. Asignaturas existentes.

Nombre del programa: DELTA10

Este comando obtiene el listado de las asignaturas pertenecientes al curso seleccionado. Permite conocer en cualquier momento cuáles son las asignaturas que pertenecen a cualquier curso dentro de un determinado plan de estudios, indicando cada una de las especialidades si se da el caso.

Explicación del organigrama.

-Bloque 1.

Selecciona la clave que indicará la carrera, plan y curso o cursos cuyas asignaturas se desean listar.

Variable:

Z\$: contiene la clave que selecciona carrera, plan y curso.

-Bloque 2.

Búsqueda de las asignaturas. Se selecciona el fichero tipo "C" adecuado y se leen las asignaturas existentes de las distintas especialidades.

Variables:

K : indica el nº del fichero a leer.

J : indica la grabacion del fichero a leer.

-Bloque 3.

Realiza el listado de las asignaturas.

Variables:

D6\$(1) : contiene el nombre de la especialidad.

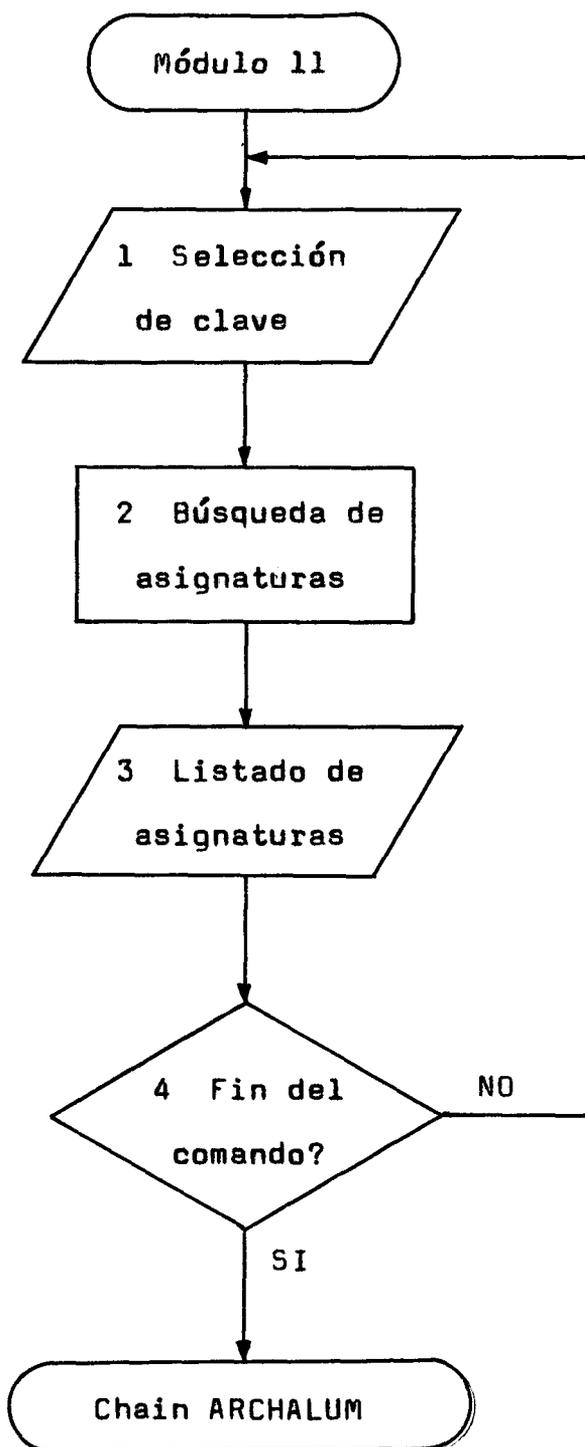
D6\$(2) : contiene el nombre de la sección (para
industriales).

F2\$(1) a F2\$(15) : contiene el nombre de las dis-
tintas asignaturas.

-Bloque 4.

Explora si ha terminado el comando. En caso nega-
tivo vuelve al bloque 1. En caso afirmativo devuel-
ve control al programa menú.

-Organigrama.



1.13 Módulo 12. Ficha del alumno.

Nombre del programa: DELTA11

Este comando visualiza la ficha de expediente actualizada de un alumno. Permite conocer los datos personales de cualquier alumno, así como sus datos académicos actualizados, incluyendo cualquier renuncia o calificación realizada durante el curso.

Explicación del organigrama.

-Bloque 1.

Introducción de nombre y apellidos del alumno cuya ficha se desea visualizar.

Variables:

A\$(1) : primer apellido.

A\$(2) : segundo apellido.

A\$(3) : nombre.

-Bloque 2.

Se explora la existencia del alumno en el fichero de expedientes. En caso de no encontrarlo pasa al bloque 7. Sino sigue con el bloque 3.

Variable:

M : indica el nº de la grabación explorada en el fichero de expedientes.

-Bloque 3.

Listado de datos personales.

Variables:

M\$(4) a M\$(22) : contienen los datos personales.

-Bloque 4.

Localización en fichero tipo "C". Este bloque se encarga de comprobar si el alumno se halla matriculado del curso correspondiente al fichero explorado.

Variable:

M : indica la posición del alumno en el fichero tipo "C". M=0 indica alumno no encontrado.

-Bloque 5.

Este bloque explora una por una las asignaturas que no consten como aprobadas en el fichero de expedientes. (En caso de que el bloque anterior haya detectado M=0, pasa al bloque 6, listando los datos leídos en fichero de expedientes).

Variables:

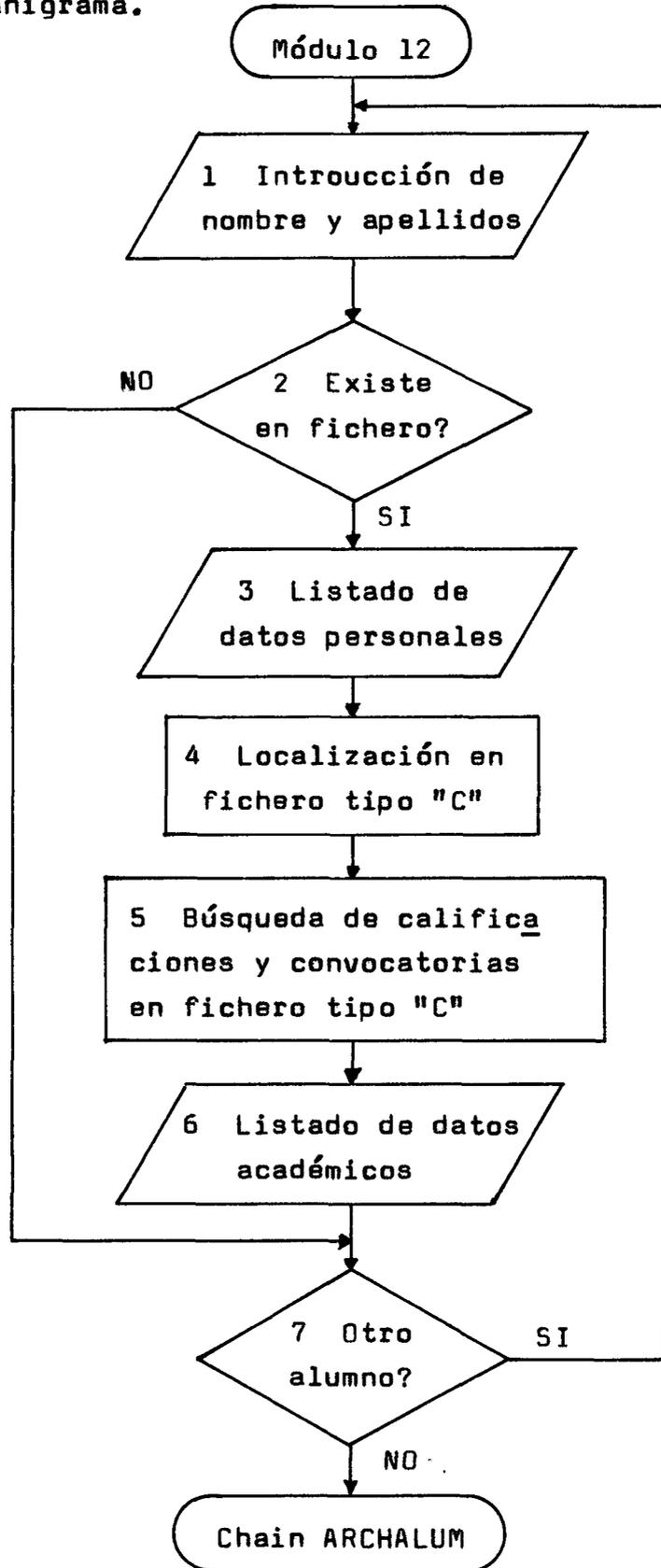
B : nº total de asignaturas.

T2\$: variable que contiene los datos de calificaciones y/o convocatorias.

-Bloque 6.

Explora si ha terminado el comando. En caso negativo vuelve al bloque 1. Sino pasa al programa menú.

-Organigrama.



1.14 Módulo 13. Listado por nacionalidad.

Nombre del programa: DELTA12

Este módulo permite la obtención de listado por nacionalidad. Cabe elegir entre dos opciones: a) Listar todos los alumnos que posean determinada nacionalidad. b) Listar el total de alumnos pertenecientes a cada una de las nacionalidades existentes en el fichero de expedientes.

Explicación del organigrama.

-Bloque 1.

Selección de nacionalidad. Se introduce el nombre de la nacionalidad buscada.

Variable:

D\$: contiene el nombre de la nacionalidad buscada.

-Bloque 2.

Se explora D\$ para conocer la opción elegida. D\$<>"": listado alfabético por nacionalidad (pasa al bloque 3). D\$="": listado del nº de alumno por nacionalidad (pasa al bloque 4).

-Bloque 3.

Este bloque lee consecutivamente la nacionalidad de cada uno de los alumnos existentes en el fichero de expedientes y toma en cuenta que la nacionalidad puede ser definida en masculino o en femenino indistintamente (español o española), por lo que analiza también los datos alfanuméricos cuya longitud sea menor o mayor en un carácter con respecto a la nacionalidad buscada, y en caso de coincidencia de los datos comparados, lista los apellidos y nombre del alumno.

Variables:

M\$(1) : primer apellido.

M\$(2) : segundo apellido.

M\$(3) : nombre.

D\$: nacionalidad seleccionada.

M\$(5) : nacionalidad del alumno.

H2\$: nacionalidad del alumno menos el último ca
racter.

H3\$: nacionalidad seleccionada menos el último
caracter.

-Bloque 4.

Este bloque lee únicamente las nacionalidades contenidas en el fichero de expedientes y las compara con las ya encontradas. En caso de que coincida con alguna ya encontrada, incrementa en uno la variable

numérica asignada adicha nacionalidad. En caso contrario la nacionalidad encontrada ocupará una nueva variable alfanumérica. Al igual que el bloque 3, realiza una comparación por defecto del último carácter de la nacionalidad encontrada, con las nacionalidades encontradas previamente.

Variables:

F3\$(1) a F3\$(45) : contienen las distintas nacionalidades encontradas.

I : nº total de nacionalidades encontradas.

C(1) a C(45) : contienen el total de alumnos que posean la nacionalidad correspondiente al índice del vector.

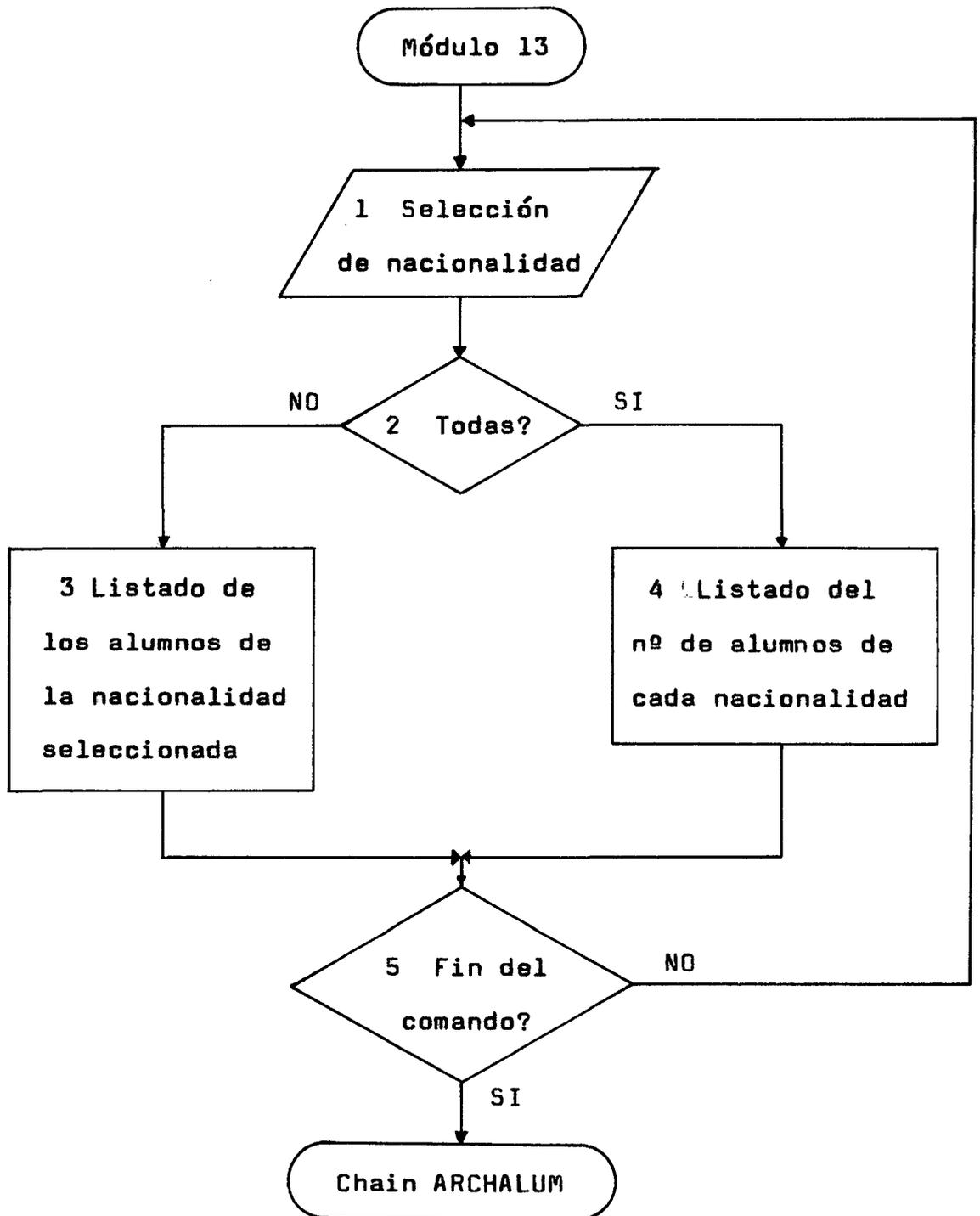
M\$(5) : nacionalidad del alumno.

A : diferencia en la longitud de caracteres entre la nacionalidad analizada y la del alumno.

-Bloque 5.

Explora si ha terminado el comando. En caso negativo vuelve al bloque 1. En caso afirmativo devuelve control al programa menú.

-Organigrama.



1.15 Módulo 14. Listado por carrera terminada.

Nombre del programa: DELTA13

Este módulo obtiene el listado alfabético por carrera terminada. Permite seleccionar límites alfabéticos entre los que aparecen todos los alumnos de las distintas carreras que consten como aprobados de proyecto.

-Bloque 1.

Se introducen los límites de listado alfabético. en caso de omisión, los límites serán "" y "ZZ".

Variables:

L1\$: límite alfabético inferior.

L2\$: límite alfabético superior.

-Bloque 2.

Se localiza la grabación que corresponde al límite inferior, a partir del cual se leerá la variable Q para saber si el alumno consta con carrera terminada o no.

Variables:

M : indica posición de la grabación del fichero a partir de la cual se empezará a leer.

Q : indica si el alumno consta como de carrera terminada (Q=-1).

-Bloque 3.

Listado del alumno. Se imprimen los apellidos y el nombre.

Variables:

M\$(1) : primer apellido.

M\$(2) : segundo apellido.

M\$(3) : nombre.

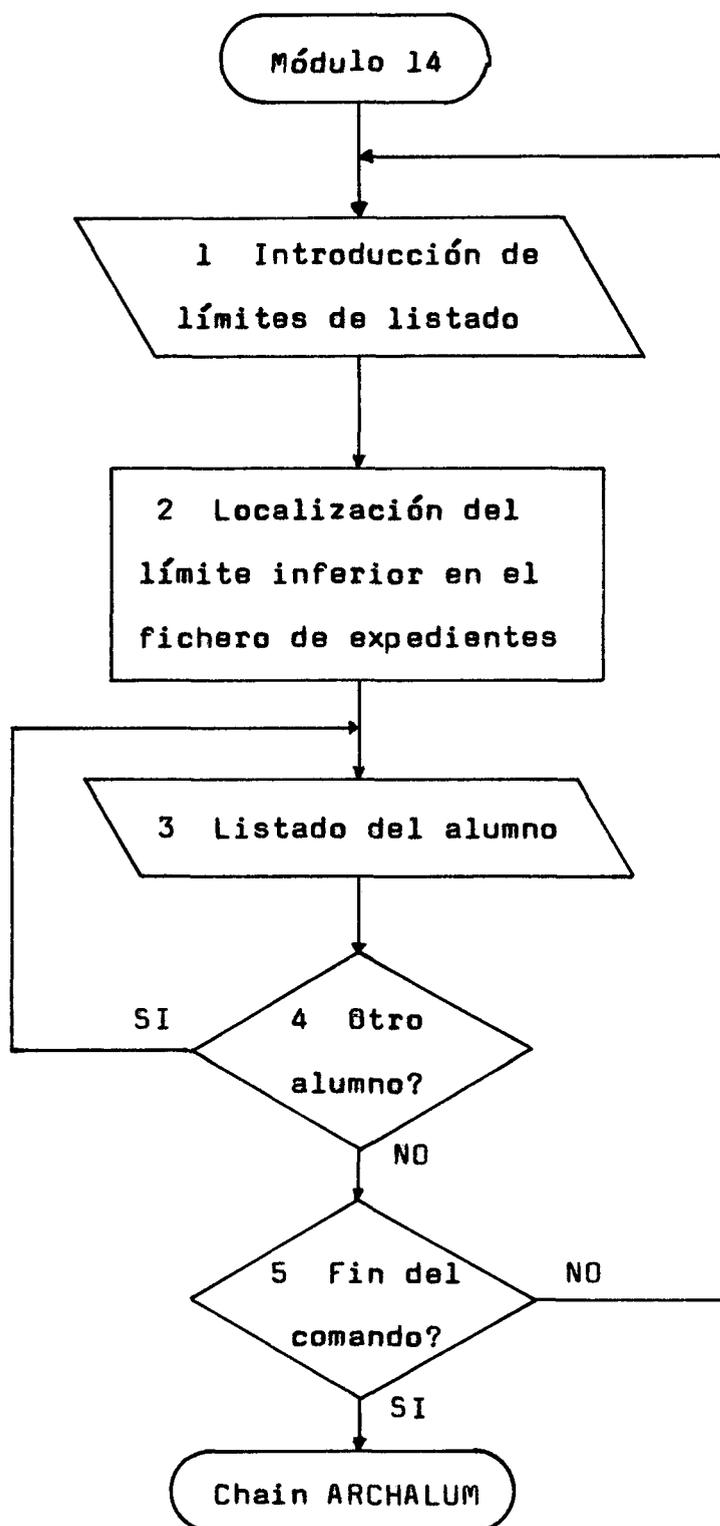
-Bloque 4.

Se explora si queda otro alumno por listar. Para ello se compara el nombre del siguiente alumno que consta como de carrera terminada, con el límite superior. Si es menor que el límite se vuelve al bloque 3. Si es mayor se pasa al bloque 5.

-Bloque 5.

Se explora si ha terminado el comando. En caso negativo se vuelve al bloque 1. en caso afirmativo se devuelve el control al programa menú.

-Organigrama.



2. FICHEROS UTILIZADOS POR EL PROGRAMA "ARCHALUM"

En esta sección se analizan los 39 ficheros encargados de archivar todos los datos con los que trabaja el programa ARCHALUM. Se dividen en tres tipos: 35 ficheros de curso (tipo "C"), 1 fichero de expedientes y 2 ficheros auxiliares. Cada tipo cumple una función determinada, si bien todos están relacionados por el intercambio de datos que mantienen.

2.1 Formato de los ficheros creados en BASIC/3000.

Todos los ficheros utilizados por el programa están formateados en BASIC/3000 y han sido creados con la longitud máxima de grabación que admite el sistema: 319 palabras. Cada palabra se compone de 16 bits. El número de palabras necesario para cada uno de los datos depende del tipo de dato de la siguiente forma:

<u>Tipo de dato</u>	<u>Nº de palabras</u>
entero	1
real	2
alfanumérico	$(\text{longitud}+1)/2$

Esto supone un máximo de 319 datos enteros, o 637 caracteres alfanuméricos, lo que se considera más que suficiente para las características de los datos a almacenar.

2.2 Ficheros tipo "C".

2.2.1 Características.

Cada uno de los ficheros tipo "C" está diseñado para contener todos los datos referentes a un curso determinado, dentro del plan de estudios de cada una de las carreras. Su misión es almacenar durante el curso los datos de los alumnos matriculados. Al iniciar el curso siguiente, todas las grabaciones que contengan datos de alumnos se borran, y los datos pasan al fichero de expedientes mediante el comando "Volcado de ficheros" (Módulo 6), estando nuevamente preparados para aceptar los datos de los nuevos alumnos.

2.2.2 Estructura.

Los ficheros tipo "C" componen un grupo de 35 ficheros con las mismas características estructurales. Cada alumno ocupa una grabación distinta, y estas están ordenadas por el orden alfabético creciente de los apellidos y nombre de los alumnos

Su longitud de grabación es de 319 palabras, y el número estándar de grabaciones es de 25. Esto no supone ninguna limitación en cuanto al número de grabaciones, ya que de ser necesario mayor número de grabaciones, éste es incrementado en 25 por el propio programa (Módulo 2: Matriculación de alumnos), con la ayuda del fichero auxiliar "HELP". Este procedimien-

to se basa en introducir los datos del fichero tipo "C" en el fichero auxiliar HELP, borrar el fichero tipo "C" y volverlo a crear introduciéndole los datos almacenados en el fichero HELP. El nombre de los ficheros está diseñado de tal forma que permite un acceso indexado a cada uno de ellos desde los distintos módulos del programa. La forma es CXXX, donde "C" representa la letra C, y los tres dígitos siguientes definen una carrera, el plan de estudios, y el curso del que se trata.

Este tipo de ficheros contiene tres tipos de grabaciones:

- Índice de fichero. Está situado en la primera grabación del fichero.
- Índice de especialidad. Pueden ser uno o varios e irán situados a continuación del índice de fichero. Contiene el nombre de las asignaturas correspondientes a la especialidad. En caso de que el fichero corresponda al del último curso de una carrera, el último dato de la grabación deberá ser "Trabajo fin de carrera".
- Datos del alumno. Contiene los datos personales y académicos.

N\$	N1	N4	N2	D7\$(1)	D7\$(3)	D7\$(2)
C512	5	25	3	NAVALES	SEGUNDO	TRIENAL
Nombre fichero	Nº grab. usadas	Nº grab. disponib.	Nº de especialidades	Carrera	Curso	Plan

D6\$(1)	D6\$(2)	A1	B	F2\$(1)	F2\$(9)
ARMAMENTO		1	9	TERMODINAMICA	INGLES I
Especialidad	Sección	Identif. de espec.	Nº de asignat.	Asignaturas	

Q	F\$(1)	F\$(2)	F\$(22)	T	T1\$(1)	T2\$(1)	T3\$(1)
1	GARCIA	LOPEZ	1234	1	C51211	2Z2Z...	ØLEMAØ..
Identif.	Datos personales			Datos académicos codificados			

2.2.3 Formato de grabaciones del fichero tipo "C".

2.3 Fichero de expedientes.

2.3.1 Características.

El fichero de expedientes almacena los datos de todos los alumnos que están o hayan estado matriculados de alguno de los cursos de cualquier carrera. Para los alumnos que están actualmente matriculados de un curso solamente guarda aquellos datos académicos que se refieren a las convocatorias, mientras que los datos que se refieren a las notas permanecen en los ficheros tipo "C" hasta que se efectúa el comando "Volcado de ficheros" (Módulo 6).

2.3.2 Estructura.

El fichero de expedientes tiene una longitud máxima de grabación de 319 palabras, y el número estándar de grabaciones es de 1001. Cada alumno ocupa una única grabación, estando estas ordenadas según el orden alfabético creciente de los apellidos y nombre de los alumnos.

Debido a la gran cantidad de datos que debe poder aceptar este fichero, existe la posibilidad de ampliarlo. Para ello he dicurrido un método ingenioso que consiste en crear otro fichero con el número de grabaciones ampliado en 1000 unidades, copiar en él el fichero de expedientes, borrar el fichero de expedientes e introducir el nombre del nuevo fichero de expedientes co

mo el segundo dato de la primera grabación del fichero auxiliar HELP. Todo esto lo realiza automáticamente el comando de matriculación (Módulo 2) cuando descubre que el número de grabaciones disponibles en el fichero de expedientes es insuficiente.

Este fichero contiene dos tipos de grabaciones:

- Índice de fichero. Está contenido en la primera grabación del fichero.
- Datos del alumno. Ocupan todas las grabaciones posteriores al índice.

N5	N6
32	1001
Nº grab. usadas	Nº grab. disponib.

Q	F\$(1)	F\$(22)	T	F3\$(1)	F4\$(1)	F5\$(1)	F3\$(3)	F4\$(3)	F5\$(3)
1	GARCIA	1234	3	C51111	3C2G..	Ø1285	C5131Ø	2ZØT..	Ø
Pro- yecto	Datos personales		Datos académicos codificados						

2.3.3 Formato de grabaciones del fichero de expedientes.

2.4 Ficheros auxiliares.

Son ficheros a los que se accede solamente temporalmente en el transcurso de un comando, y cuya misión es agilizar las operaciones con los ficheros tipo "C" y el fichero de expedientes.

2.4.1 Fichero HELP.

2.4.1.1 Características.

El fichero HELP realiza una doble función. Por una parte guarda el nombre del fichero de expedientes, y por otra se usa para ampliar los ficheros tipo "C". Para esto último se comprueba primero si la longitud del fichero HELP es suficiente (en caso contrario se amplía) y seguidamente se introducen los datos del fichero tipo "C" y vuelven a introducirse los datos previamente almacenados en el fichero HELP. Estos datos permanecen en el fichero HELP hasta que son cambiados por otros en una posible ampliación posterior.

2.4.1.2 Estructura.

El fichero HELP tiene una longitud de grabación de 319 palabras, y una longitud estándar de 26, siendo posible su ampliación por programa. Contiene dos tipos de grabaciones:

- Índice de fichero. Está contenido en la primera grabación del fichero.

+Datos del alumno. Ocupan las posiciones posteriores al índice

2.4.1.3 Formato de grabaciones del fichero HELP.

N3	G\$
26	G2
Nº grab. disponibles	Nombre del fichero de expedientes

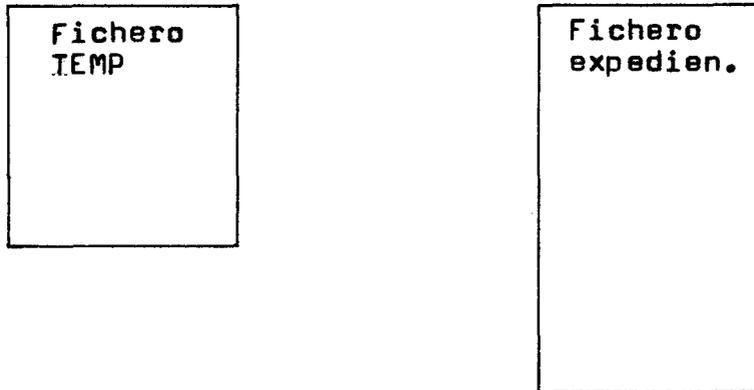
Q	F\$(1)	F\$(22)	T	T1\$(1)	T2\$(1)	T3\$(1)
1	GARCIA	1234	1	C51211	2Z2Z..	ØLEMAØ..
Identif. de espec.	Datos personales			Datos académicos codificados		

2.4.2 Fichero TEMP.

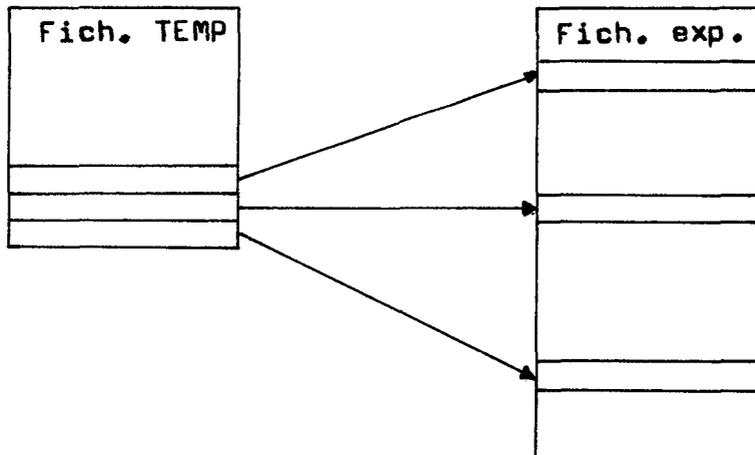
2.4.2.1 Características.

Se trata de un fichero auxiliar cuya misión es facilitar la introducción de datos de alumnos nuevos en el fichero de expedientes. Esto está justificado por el hecho de que al estar el fichero de expedientes ordena alfabéticamente, cada alumno nuevo insertado supone el desplazamiento de todas las grabaciones de alumnos que le siguieran en orden. Dado que el fichero de expedientes está diseñado para contener una gran cantidad de datos, y su desplazamiento ocuparía un tiempo no despreciable, he optado por la solución de ir introduciendo los datos de todos los alumno nuevos en el fichero TEMP hasta el fin del comando de matriculación de alumnos (Módulo 2), en cuyo caso se produce el volcado de los datos contenidos en el fichero TEMP, empezando por el último, y desplazando los datos del fichero de expedientes que sigan en orden al último del fichero TEMP, en tantas posiciones como datos del fichero TEMP quedan por introducir.

Durante la ejecución del comando.



Al dar por terminada la ejecución del comando.



2.4.2.2 Estructura.

La longitud máxima de cada grabación del fichero TEMP es de 319 palabras, y su longitud es de 100 grabaciones. Dadas sus características, este fichero no es ampliable y las grabaciones contenidas son todas del mismo tipo y contienen exclusivamente datos de alumnos.

Q	F\$(1)	F\$(22)	T	F3\$(1)	F4\$(1)	F5\$(1)	F3\$(3)	F4\$(3)	F5\$(3)
1	GARCIA	1234	3	C51111	3C2G..	ø1285	C51310	2ZøT..	ø
Pro- yecto	Datos personales		Datos académicos codificados						

2.4.2.3 Formato de grabaciones del fichero TEMP.

2.5 Dimensión máxima de las variables que introducen datos en ficheros.

El primer índice indica el número de variables que componen el vector. El segundo índice indica el número máximo de caracteres que componen la variable. Como puede observarse, se permite un máximo de 15 asignaturas por especialidad y un máximo de 10 cursos por alumno.

D6\$(2,30)	D7\$(3,30)	F\$(22,30)	F2\$(15,70)
F3\$(10,6)	F4\$(10,31)	F5\$(10,200)	G\$(1,8)
N\$(1,4)	T1\$(1,6)	T2\$(1,31)	T3\$(1,45)

2.6 Interpretación de las codificaciones de los datos académicos.

2.6.1 Codificación en ficheros tipo "C" y HELP.

Q : identifica la especialidad de la que está matriculado el alumno. En caso de no existir especialidad su valor es 1.

T1\$(1) : T1\$(1)="C51211". Los primeros cuatro dígitos hacen referencia al fichero al que se refieren los datos contenidos en T2\$(1) y T3\$(1). El quinto dígito indica la especialidad. El sexto dígito indica nuevo o repetidor (Ø ó 1).

T2\$(1) : T2\$(1)="2Z2Z1G3H....". Está organizado por pares de dígitos. Cada uno de los pares hace re-

ferencia a las distintas asignaturas de la especialidad del alumno, en el orden en que aquellas vienen detalladas en la grabación correspondiente. El primer dígito del par hace referencia al número de convocatorias, y el segundo a la convocatoria en sí. Pueden darse las siguientes condiciones:

a) Nunca ha estado matriculado de la asignatura.

<u>1º dígito</u>	<u>2º dígito</u>
∅	T

b) No está matriculado actualmente de la asignatura y consta como aprobada en años anteriores. El primer dígito indica la convocatoria en la que aprobó. El segundo indica la calificación obtenida.

<u>1º dígito</u>	<u>2º dígito</u>
1	A = Matricula
2	B = Sobres. 10
3	C = Sobres. 9
4	D = Notable 8
5	E = Notable 7
6	F = Aprobado 6
7	G = Aprobado 5

c) No está matriculado actualmente de la asignatura y consta como suspendido en años anteriores.

El primer dígito indica las convocatorias agotadas. El segundo indica la calificación obtenida. Los valores de los distintos dígitos serán:

<u>1º dígito</u>	<u>2º dígito</u>
1	H = Suspenso 4
2	I = Suspenso 3
3	J = Suspenso 2
4	K = Suspenso 1
5	L = Suspenso \emptyset
6	M = No presentado
7	

d) Está matriculado actualmente de la asignatura. El primer dígito indica en qué convocatoria se ha lla y el segundo cuál es la convocatoria.

Para una sola convocatoria:

<u>1º dígito</u>	<u>2º dígito</u>
1 = 1ª convoc.	W = Febrero
2 = 2ª convoc.	X = Septiembre
3 = 3ª convoc.	Y = Junio
4 = 4ª convoc.	
5 = 5ª convoc.	
6 = 6ª convoc.	
7 = 7ª convoc.	

Para dos convocatorias:

<u>1º dígito</u>	<u>2º dígito</u>
2 = 1ª y 2ª conv.	U = Feb. y Jun.
3 = 2ª y 3ª conv.	V = Feb. y Sep.
4 = 3ª y 4ª conv.	Z = Jun. y Sep.
5 = 4ª y 5ª conv.	
6 = 5ª y 6ª conv.	

T3\$(1) : T3\$(1) = "ØLEMAØØØØØØ...". Está organizada por ternas de dígitos. Cada terna hace referencia a las distintas asignaturas de la especialidad del alumno en el orden en que estas vienen detalladas en la grabación correspondiente. Tienen correspondencia con el orden de los pares de la variable T2\$. Cada uno de los dígitos hace hace referencia a las calificaciones obtenidas en el curso actual. Para cada terna el primer dígito almacena la calificación de la convocatoria de febrero, el segundo la de junio y el tercero la de septiembre. Solo podrán constar como calificados aquellos dígitos que correspondan a las convocatorias actualmente matriculadas. Pueden darse las siguientes condiciones:

a) No está matriculado de ninguna convocatoria de la asignatura a la que corresponde la terna.

<u>1º dígito</u>	<u>2º dígito</u>	<u>3º dígito</u>
∅	∅	∅

b) Está matriculado pero no calificado de ninguna convocatoria en la asignatura a la que corresponde la terna.

<u>1º dígito</u>	<u>2º dígito</u>	<u>3º dígito</u>
∅	∅	∅

c) Está matriculado, calificado y aprobado de la primera convocatoria de dos matriculadas. En este caso los dígitos siguientes serán obligatoriamente ∅.

Convocatoria de febrero aprobada

<u>1º dígito</u>	<u>2º dígito</u>	<u>3º dígito</u>
A	∅	∅
B		
C		
D		
E		
F		
G		

Convocatoria de Junio aprobada.

<u>1º dígito</u>	<u>2º dígito</u>	<u>3º dígito</u>
∅	A	∅
	B	
	C	
	D	
	E	
	F	
	G	

d) Está matriculado y calificado de las convocatorias matriculadas. (Téngase en cuenta que nunca podrá estar matriculado de 3, por lo que uno de los dígitos será obligatoriamente ∅.

<u>1º dígito</u>		<u>2º dígito</u>		<u>3º dígito</u>	
∅	K	∅	K	∅	K
A	L	A	L	A	L
B	M	B	M	B	M
C		C		C	
D		D		D	
E		E		E	
F		F		F	
G		G		G	
H		H		H	
I		I		I	
J		J		J	

2.6.2 Codificación en fichero de expedientes y fichero TEMP.

Q : Q=1 indica alumno no matriculado de proyecto.

Q=∅ indica matriculado de proyecto. Q=-1 indica carrera terminada.

F3\$(1) a F3\$(10) : Tienen la misma codificación que la variable T1\$ en los ficheros tipo "C". El índice del 1 al 10 relaciona las tres variables F3\$, F4\$, F5\$ que pertenecen al mismo curso.

T : indica el número total de cursos de los que está y/o ha estado matriculado un alumno.

F4\$(1) a F4\$(10) : tiene la misma codificación que la variable T2\$ en los ficheros tipo "C"

F5\$(1) a F5\$(10) : F5\$(K)="∅12851385.....". Esta variable indica las convocatorias no presentadas en cursos anteriormente matriculados. Está compuesta por un primer dígito constante sin significado (∅), y una serie de cuartetos que van añadiéndose sucesivamente sin que tenga relación con el orden de las asignaturas atribuido a la variable F4\$.

El primer dígito el número que le corresponde a la asignatura en el orden en que vienen detalladas en la grabación del fichero tipo "C" correspondiente. El segundo dígito indica la convocatoria no presentada. El tercer y cuarto dígitos señalan el año de la convocatoria no presentada.

Los valores posibles son:

<u>1º dígito</u>	<u>2º dígito</u>	<u>3º dígito</u>
1	1	...
2	2	83
3	3	84
4		85
5		86
6		...
7		
8		
9		
10		
11		
12		
13		
14		
15		

3. LISTADO DEL PROGRAMA ARCHALUM

ARCHALUM

```
10 REM*****
20 REM
30 REM          MOD 1.      PROGRAMA  MENU
40 REM
50 REM*****
60 COM W#[9]
70 DIM P#[2]
80 W#="/CGM"
90 REM.....
100 REM  FORMATO PAGINA 1.
110 REM.....
120 PRINT '27"h' '27"J";
130 PRINT "      "'27&
    "&dc&cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc&
cccccccccccccccc" '27"&da"
140 PRINT "      "'27"&dc&" '27&
    "&da
    "'27"&dc&" '27"&da"
150 PRINT "      "'27"&dc&" '27&
    "&da
    "'27"&dc&" '27"&da"
160 PRINT "      "'27"&dc&" '27&
    "&da
    "'27"&dc&" '27"&da"
170 PRINT "      "'27"&dc&" '27&
    "&da
    "'27"&dc&" '27"&da"
180 PRINT "      "'27"&dc&" '27&
    "&da  1. MATRICULACION
    "'27"&dc&" '27"&da"
190 PRINT "      "'27"&dc&" '27&
    "&da
    "'27"&dc&" '27"&da"
```

A R C H A L U M

9. TABLA POR CURSO

```

200 PRINT "      "'27"&dCa"'27&
   "&d@  2. RENUNCIA DE CONVOCATORIAS  10. ASIGNATURAS EXISTENTES  &
      "'27"&dCa"'27"&d@"
210 PRINT "      "'27"&dCa"'27&
   "&d@                                     &
      "'27"&dCa"'27"&d@"
220 PRINT "      "'27"&dCa"'27&
   "&d@  3. CAMBIO DE CONVOCATORIAS  11. FICHA DE ALUMNO  &
      "'27"&dCa"'27"&d@"
230 PRINT "      "'27"&dCa"'27&
   "&d@                                     &
      "'27"&dCa"'27"&d@"
240 PRINT "      "'27"&dCa"'27&
   "&d@  4. INTRODUCCION DE NOTAS  12. LISTADO POR NACIONALIDAD&
      "'27"&dCa"'27"&d@"
250 PRINT "      "'27"&dCa"'27&
   "&d@                                     &
      "'27"&dCa"'27"&d@"
260 PRINT "      "'27"&dCa"'27&
   "&d@  5. VOLCADO DE FICHEROS  13. LISTADO POR CARRERA TERM&
INADA "'27"&dCa"'27"&d@"
270 PRINT "      "'27"&dCa"'27&
   "&d@                                     &
      "'27"&dCa"'27"&d@"
280 PRINT "      "'27"&dCa"'27&
   "&d@  6. LISTADO POR CARRERA  14. SALIDA  &
      "'27"&dCa"'27"&d@"
290 PRINT "      "'27"&dCa"'27&
   "&d@                                     &
      "'27"&dCa"'27"&d@"
300 PRINT "      "'27"&dCa"'27&
   "&d@  7. LISTADO POR CURSO  &
      "'27"&dCa"'27"&d@"
310 PRINT "      "'27"&dCa"'27&
   "&d@                                     &
      COMANDO SELECCIONADO?  &

```


DELTA1

```

500 REM*****
502 REM
504 REM          MODULO 2.          MATRICULACION DE ALUMNOS
506 REM
508 REM*****
510 C#[1]="PRIMER APELLIDO"
512 C#[2]="SEGUNDO APELLIDO"
514 C#[3]="NOMBRE"
516 C#[4]="SEXO"
518 C#[5]="NACIONALIDAD"
520 C#[6]="NO. DOC. IDENT. O PASAPORTE"
522 C#[7]="FECHA EXPEDICION"
524 C#[8]="LUGAR DE EXPEDICION"
526 C#[9]="LUGAR DE NACIMIENTO"
528 C#[10]="PROVINCIA"
530 C#[11]="FECHA"
532 C#[12]="EDAD"
534 C#[13]="ESTADO CIVIL"
536 C#[14]="NOMBRE DEL PADRE"
538 C#[15]="NOMBRE DE LA MADRE"
540 C#[16]="REPRESENTANTE LEGAL"
542 C#[17]="DOMICILIO (LOCALIDAD)"
544 C#[18]="DOMICILIO (CALLE)"
546 C#[19]="TELEFONO"
548 C#[20]="PROCEDENCIA (COU O FP (ESPECIALIDAD))"
550 C#[21]="TIPO DE ENSEANZA"
552 C#[22]="NO. DE EXPEDIENTE"
554 COM W#[*]
556 DIM A#[4],A#[3,1],A1#[15],B#[3,10],C#[22,30],C1#[5,15],C2#[4,10]
558 DIM C3#[2,10],D1#[4,1],D2#[4,4],D3#[3,30],D4#[90],D5#[3,30]
560 DIM D6#[2,30],D7#[3,30],E2#[4],F#[22,30],F2#[15,70],F3#[10,6]
562 DIM F4#[10,31],F5#[10,200],G#[8],G1#[8],H1#[6],H2#[31],H3#[45]
564 DIM M#[23,30],N#[5],R#[2],T1#[10,6],T2#[10,31],T3#[10,45],Z#[5]
566 DIM D#[80],W3#[15]

```

```

568 B#[ 1 ]=" CARRERA",B#[ 3 ]=" CURSO",B#[ 2 ]=" PLAN"
570 C1#[ 1 ]=" INDUSTRIAL",C1#[ 2 ]=" TELECOMUNICACION",C1#[ 5 ]=" NAVAL"
572 C1#[ 4 ]=" OBRAS PUBLICAS",C1#[ 3 ]=" TOPOGRAFIA"
574 C2#[ 1 ]=" PRIMERO",C2#[ 2 ]=" SEGUNDO",C2#[ 3 ]=" TERCERO",C2#[ 4 ]=&
" CUARTO"
576 C3#[ 1 ]=" TRIENAL",C3#[ 2 ]=" CUATRIENAL"
578 FILES *,*,*,*,*,*,*,*,*
580 ASSIGN "HELP"+W#,5,X1
582 READ #5,1;N3,G#
584 ASSIGN G#+W#,6,X2
586 P6=6
588 ASSIGN "TEMP"+W#,7,X1
590 SYSTEM X1,"FILE QQQQ=#STDIN;REC=-1"
592 ASSIGN "QQQQ",9,X1
594 N=0
596 I=0
598 I=I+1
600 REM.....
602 REM   FORMATO PAGINA 1. SELECCION DE CARRERA, PLAN Y CURSOS      .
604 REM   (EL No. DE CURSOS SELECCIONADOS SERA EQUIVALENTE A J-1)  .
606 REM.....
608 PRINT '27"h''27"J"
610 PRINT '27"&a1r6C''27&
   "&dC Introduzca clave para seleccionar carrera, plan y curso(s)  &
   "'27"&d@"
612 PRINT LIN(2);TAB(11);"Carrera:";TAB(33);"Plan:";TAB(54);"Curso:"
614 PRINT TAB(11);"-----";TAB(33);"-----";TAB(54)&
;"-----"
616 PRINT TAB(11);"1. ";C1#[ 1 ];TAB(33);"1. ";C3#[ 1 ];TAB(54);"1. ";C2#[
[ 1 ]
618 PRINT TAB(11);"2. ";C1#[ 2 ];TAB(33);"2. ";C3#[ 2 ];TAB(54);"2. ";C2#[
[ 2 ]
620 PRINT TAB(11);"3. ";C1#[ 3 ];TAB(54);"3. ";C2#[ 3 ]
622 PRINT TAB(11);"4. ";C1#[ 4 ];TAB(54);"4. ";C2#[ 4 ]
624 PRINT TAB(11);"5. ";C1#[ 5 ];TAB(54)

```

```

626 PRINT '27"&a14r6C";"CLAVE?";
628 PRINT '27"&a14r12C"'27"&dC      "'27"&d@"'27"&a14r12C";
630 ENTER 255,A,Z#
632 IF Z#[1;1]>"5" OR Z#[1;1]<"1" THEN 628
634 ELSE A#[1]=Z#[1;1]
636 IF Z#[2;1]>"2" OR Z#[2;1]<"1" THEN 628
638 ELSE A#[2]=Z#[2;1]
640 IF (Z#[3;1]>"4" OR Z#[3;1]<"1") AND Z#[3;1]<>" " THEN 628
642 IF Z#[2;1]="1" AND Z#[3;1]="4" THEN 628
644 FOR W0=4 TO LEN(Z#)
646   IF (Z#[W0;1]>"4" OR Z#[W0;1]<"1") THEN 628
648   IF Z#[2;1]="1" AND Z#[W0;1]="4" THEN 628
650 NEXT W0
652 PRINT '27"h"'27"J"
654 FOR J=1 TO 4
656   IF Z#[2+J;1]<>" " AND Z#[2+J;1]<>" " THEN DO
658     D1#[J]=Z#[2+J;1]
660     CONVERT D1#[J] TO A#[J]
662     D2#[J]="C"+A#[1]+A#[2]+D1#[J]
664   DOEND
666   ELSE 672
668 NEXT J
670 REM.....
672 REM  APERTURA DE TODOS LOS FICHEROS CORRESPONDIENTES A CARRERA .
674 REM  Y PLAN SELECCIONADOS .
676 REM.....
678 FOR K1=1 TO 3
680   CONVERT K1 TO K1#
682   ASSIGN "C"+A#[1]+A#[2]+K1#+W#,K1,X1
684 NEXT K1
686 IF A#[2]="2" THEN ASSIGN "C"+A#[1]+A#[2]+"4"+W#,4,X1
688 REM.....
690 REM  INTRODUCCION DEL NOMBRE DEL ALUMNO, Y BUSQUEDA DE ESTE EN .
692 REM  ARCHIVO DE EXPEDIENTES (W1=0: LOCALIZADO; W1=1: NO LOCAL. .
694 REM.....

```

```

696 PRINT '27"A";"Primer apellido";TAB(26);"Segundo apellido";TAB(51)&
;"Nombre"
698 PRINT '27"&dC          "'27"&d@      "'27&
"&dC          "'27"&d@      "'27&
"&dC          "'27"&d@";
700 PRINT '27"A"'27"A";
702 PRINT '27"[          - "'27" ]      "'27&
"["          "'27" ]      "'27"[          "'27" ]";
704 FOR K=1 TO 3
706   PRINT '27"W";
708   PRINT (FOR K1=1 TO K-1,(FOR K2=1 TO 25,'27"C" ););
710   ENTER 255,A,M#[K]
712   M#[K]=UPS$(DEB$(M#[K]))
714   PRINT '27"A"'27"X"
716 NEXT K
718 D4#=M#[ 1 ]+M#[ 2 ]+M#[ 3 ]
720 READ #6,1;N5,N6
722 IF N5=1 THEN DO
724   W1=1
726   GOTO 778
728 DOEND
730 L=2,H=N5
732 M=INT((L+H)/2)
734 IF M=1 THEN DO
736   W1=1
738   GOTO 778
740 DOEND
742 READ #6,M;Q,(FOR K1=1 TO 3,D3#[K1])
744 IF D3#[ 1 ]+D3#[ 2 ]+D3#[ 3 ]>=D4# THEN 758
746 IF L>=H THEN DO
748   W1=1
750   GOTO 778
752 DOEND
754 L=M+1
756 GOTO 732

```

```

758 IF D3#[ 1 ]+D3#[ 2 ]+D3#[ 3 ]=D4# THEN DO
760   W1=0
762   GOTO 778
764 DOEND
766 IF L>=H THEN DO
768   W1=1
770   GOTO 778
772 DOEND
774 H=M-1
776 GOTO 732
778 IF W1=0 THEN DO
780   R2=M,R1=0
782   READ #6;(FOR K1=4 TO 22,M#[K1]),T,(FOR K1=1 TO T,F3#[K1],F4#[K1&
      ],F5#[K1])
784 DOEND
786 ELSE DO
788   FOR K1=4 TO 22
790     M#[K1]=" "
792   NEXT K1
794 DOEND
796 REM.....
798 REM   FORMATO PAGINA 2. IMPRESION DE DATOS PERSONALES DEL
800 REM   ALUMNO (Y LOS DATOS CORRESP. SI EXISTE EN ARCHIV. EXP.)
802 REM.....
804 PRINT '27"&a2r0C";&
      "Sexo Nacionalidad   DNI o Pasaporte   Fecha exp.   Lugar de expe&
dicion"
806 PRINT '27"&a3r2C";'27"&dC "'27"&d@   "'27"&dC   "'27&
      "&d@ "'27"&dC   "'27"&d@   "'27"&dC   "'27&
      "&d@ "'27"&dC   "'27"&d@"
808 PRINT '27"&a3r0C"
810 PRINT M#[ 4 ];TAB(5);M#[ 5 ];TAB(22);M#[ 6 ];TAB(40);M#[ 7 ];TAB(52);M#[ 8&
      ]
812 PRINT '27"&a5r0C";&
      "Lugar de nacimiento   Provincia   Fecha   Edad   Estad&

```

```

o civil"
814 PRINT '27"&a7r0C"'27"&dC          "'27"&d@ "'27&
    "&dC          "'27"&d@ "'27"&dC          "'27"&d@ "'27&
    "&dC "'27"&d@ "'27"&dC          "'27"&d@"
816 PRINT '27"&a6r0C"
818 PRINT M#[9];TAB(21);M#[10];TAB(42);M#[11];TAB(54);M#[12];TAB(60);&
    M#[13]
820 PRINT '27"&a8r0C";&
    "Nombre del padre          Nombre de la madre          Representante&
legal"
822 PRINT '27"&a10r0C"'27"&dC          "'27"&d@ "'27&
    "&dC          "'27"&d@ "'27&
    "&dC          "'27"&d@"
824 PRINT '27"&a9r0C"
826 PRINT M#[14];TAB(26);M#[15];TAB(52);M#[16]
828 PRINT '27"&a11r0C";&
    "Domicilio (Localidad)      Domicilio (Calle)          Telefon&
o"
830 PRINT '27"&a13r0C"'27"&dC          "'27"&d@ "'27&
    "&dC          "'27"&d@ "'27&
    "&dC          "'27"&d@"
832 PRINT '27"&a12r0C"
834 PRINT M#[17];TAB(25);M#[18];TAB(58);M#[19]
836 PRINT '27"&a15r0C";&
    "Procedencia (COU/FP)      Tipo ensenanza          No. expediente"
838 PRINT '27"&a16r0C"'27"&dC          "'27"&d@ "'27&
    "&dC          "'27"&d@ "'27"&dC          "'27"&d@"
840 PRINT '27"&a15r0C"
842 PRINT M#[20];TAB(25);M#[21];TAB(43);M#[22]
844 IF W1=1 THEN DO
846   PRINT '27"&a22r6C"'27&
    "&dC Introduzca datos personales.          "&
    '27"&d@"
848 DOEND
850 IF W1=0 THEN DO

```

```

852 PRINT '27"&a22r6C"'27&
    "&dC Desea modificar algun dato? (SI/NO).           "&
    '27"&d@"
854 PRINT '27"&a22r43C";
856 ENTER 255,W2,L#
858 IF L#[1;1]<>"S" AND L#[1;1]<>"N" THEN 854
860 IF L#[1;1]="N" THEN 1114
862 ELSE W1=2
864 DOEND
866 REM.....
868 REM TOMA DE DATOS PERSONALES DEL ALUMNO (W1=0) O MODIFICACION .
870 REM DE DATOS PERSONALES (W1=2) .
872 REM.....
874 PRINT '27"&a4r0C";
876 ENTER 255,W2,D6#[1]
878 IF W1=2 AND D6#[1]=" " THEN 884
880 IF D6#[1]<>"V" AND D6#[1]<>"H" THEN 874
882 M#[4]=D6#[1]
884 PRINT '27"&a4r5C";
886 ENTER 255,W2,D6#[1]
888 IF W1=2 AND D6#[1]=" " THEN 894
890 M#[5]=UPS$(DEB$(D6#[1,1,15]))
892 IF M#[5]=" " THEN 884
894 PRINT '27"&a4r22C";
896 ENTER 255,W2,D6#[1]
898 IF W1=2 AND D6#[1]=" " THEN 904
900 M#[6]=UPS$(DEB$(D6#[1,1,15]))
902 IF M#[6]=" " THEN 894
904 PRINT '27"&a4r40C";
906 ENTER 255,W2,D6#[1]
908 IF W1=2 AND D6#[1]=" " THEN 914
910 M#[7]=UPS$(DEB$(D6#[1,1,10]))
912 IF M#[7]=" " THEN 904
914 PRINT '27"&a4r52C";
916 ENTER 255,W2,D6#[1]

```

```

918 IF W1=2 AND D6#[ 1]=" " THEN 924
920 M#[ 8]=UPS$(DEB$(D6#[ 1, 1, 19]))
922 IF M#[ 8]=" " THEN 914
924 PRINT '27"&a7r0C";
926 ENTER 255,W2,D6#[ 1]
928 IF W1=2 AND D6#[ 1]=" " THEN 934
930 M#[ 9]=UPS$(DEB$(D6#[ 1, 1, 19]))
932 IF M#[ 9]=" " THEN 924
934 PRINT '27"&a7r21C";
936 ENTER 255,W2,D6#[ 1]
938 IF W1=2 AND D6#[ 1]=" " THEN 944
940 M#[ 10]=UPS$(DEB$(D6#[ 1, 1, 19]))
942 IF M#[ 10]=" " THEN 934
944 PRINT '27"&a7r42C";
946 ENTER 255,W2,D6#[ 1]
948 IF W1=2 AND D6#[ 1]=" " THEN 954
950 M#[ 11]=UPS$(DEB$(D6#[ 1, 1, 10]))
952 IF M#[ 11]=" " THEN 944
954 PRINT '27"&a7r54C";
956 ENTER 255,W2,D6#[ 1]
958 IF W1=2 AND D6#[ 1]=" " THEN 964
960 M#[ 12]=UPS$(DEB$(D6#[ 1, 1, 2]))
962 IF M#[ 12]=" " THEN 954
964 PRINT '27"&a7r60C";
966 ENTER 255,W2,D6#[ 1]
968 IF W1=2 AND D6#[ 1]=" " THEN 974
970 M#[ 13]=UPS$(DEB$(D6#[ 1, 1, 12]))
972 IF M#[ 13]=" " THEN 964
974 PRINT '27"&a10r0C";
976 ENTER 255,W2,D6#[ 1]
978 IF W1=2 AND D6#[ 1]=" " THEN 984
980 M#[ 14]=UPS$(DEB$(D6#[ 1, 1, 22]))
982 IF M#[ 14]=" " THEN 974
984 PRINT '27"&a10r26C";
986 ENTER 255,W2,D6#[ 1]

```

```

988 IF W1=2 AND D6$(1)=" THEN 994
990 M$(15)=UPS$(DEB$(D6$(1,1,22)))
992 IF M$(15)=" THEN 984
994 PRINT '27"&a10r52C";
996 ENTER 255,W2,D6$(1)
998 IF W1=2 AND D6$(1)=" THEN 1004
1000 M$(16)=UPS$(DEB$(D6$(1,1,20)))
1002 IF M$(16)=" THEN 994
1004 PRINT '27"&a13r0C";
1006 ENTER 255,W2,D6$(1)
1008 IF W1=2 AND D6$(1)=" THEN 1014
1010 M$(17)=UPS$(DEB$(D6$(1,1,21)))
1012 IF M$(17)=" THEN 1004
1014 PRINT '27"&a13r25C";
1016 ENTER 255,W2,D6$(1)
1018 IF W1=2 AND D6$(1)=" THEN 1024
1020 M$(18)=UPS$(DEB$(D6$(1,1,29)))
1022 IF M$(18)=" THEN 1014
1024 PRINT '27"&a13r58C";
1026 ENTER 255,W2,D6$(1)
1028 IF W1=2 AND D6$(1)=" THEN 1034
1030 M$(19)=UPS$(DEB$(D6$(1,1,14)))
1032 IF M$(19)=" THEN 1024
1034 PRINT '27"&a16r0C";
1036 ENTER 255,W2,D6$(1)
1038 IF W1=2 AND D6$(1)=" THEN 1044
1040 M$(20)=UPS$(DEB$(D6$(1,1,21)))
1042 IF M$(20)=" THEN 1034
1044 PRINT '27"&a16r25C";
1046 ENTER 255,W2,D6$(1)
1048 IF W1=2 AND D6$(1)=" THEN 1054
1050 M$(21)=UPS$(DEB$(D6$(1,1,14)))
1052 IF M$(21)=" THEN 1044
1054 PRINT '27"&a16r43C";
1056 ENTER 255,W2,D6$(1)

```

```

1058 IF W1=2 AND D6#[1]=" " THEN 1072
1060 M#[22]=UPS$(DEB$(D6#[1,1,11]))
1062 IF M#[22]=" " THEN 1054
1064 REM.....
1066 REM   INTRODUCCION EN ARCHIVO DE EXPEDIENTES DE LOS DATOS      .
1068 REM   PERSONALES MODIFICADOS SI SE DA EL CASO (W1=2)        .
1070 REM.....
1072 IF W1=2 THEN DO
1074   PRINT #6,M;Q,(FOR K1=1 TO 22,M#[K1]),T,(FOR K1=1 TO T,F3#[K1],&
      F4#[K1],F5#[K1])
1076 DOEND
1078 REM.....
1080 REM   CREACION DE EXPEDIENTE EN ARCHIVO TEMPORAL (FILE 'TEMP') .
1082 REM   SI SE DA EL CASO (W1=1)                                .
1084 REM.....
1086 IF W1=1 THEN DO
1088   P1=N,P2=P5=7,P4=P3=1
1090   GOSUB 9000
1092   N=N+1
1094   R1=1
1096   R2=R0
1098   PRINT #7,R2;1
1100   PRINT #7:(FOR K1=1 TO 22,M#[K1])
1102   PRINT #7;0
1104   PRINT #7;" ", " ", " ",END
1106 DOEND
1108 REM.....
1110 REM   MATRICULACION SUCESIVA DE LOS CURSOS SELECCIONADOS      .
1112 REM.....
1114 FOR K2=1 TO J-1
1116   REM.....
1118   REM   FORMATO PAGINA 3.                                       .
1120   REM.....
1122   PRINT '27"h''27"J"
1124   R3=0

```

```

1126 READ #A[K2],1;N$,N1,N4,N2,D7#[1],D7#[3],D7#[2]
1128 PRINT TAB(6);"Carrera: ";D7#[1];TAB(35);"Plan: ";D7#[2];TAB(57)&
;"Curso: ";D7#[3];LIN(1)
1130 REM.....
1132 REM SELECCION DE ESPECIALIDAD SI SE DA EL CASO .
1134 REM.....
1136 FOR K3=1 TO N2
1138 READ #A[K2],K3+1;D6#[1],D6#[2],A1[K3]
1140 IF D6#[1]<>" AND K3=1 THEN PRINT TAB(6);"Especialidad:";LIN(&
1);TAB(5);"-----";
1142 CONVERT A1[K3] TO L$
1144 IF D6#[1]<>" THEN PRINT LIN(1);TAB(6);L$;". ";D6#[1];
1146 IF D6#[2]<>" THEN PRINT TAB(35);"Seccion: ";D6#[2];
1148 NEXT K3
1150 IF D6#[1]<>" THEN DO
1152 PRINT '27"&a22r6C''27&
"&dC Especialidad seleccionada? &
''27"&d@"
1154 PRINT '27"&a22r34C";
1156 ENTER 255,W2,L3$
1158 IF (L3$<"1" OR L3$>L$) THEN 1154
1160 CONVERT L3$ TO B1
1162 READ #A[K2],B1+1;D6#[1],D6#[2],A1[K3]
1164 PRINT '27"&a2r6C''27"J"
1166 PRINT TAB(6);"Especialidad: ";D6#[1];
1168 IF D6#[2]<>" THEN PRINT TAB(45);"Seccion: ";D6#[2];
1170 DOEND
1172 ELSE B1=1
1174 REM.....
1176 REM LOCALIZACION DE LA POSICION DEL ALUMNO EN EL FICHERO .
1178 REM TIPO 'C' Y POSIBLE AMPLIACION DE ESTE .
1180 REM.....
1182 IF N1=N4 THEN DO
1184 READ #5,1;N3
1186 IF N4>N3-1 THEN DO

```

```

1188      ASSIGN *,5
1190      PURGE X6,"HELP"+W$
1192      CREATE X7,"HELP"+W$,N4+1,319
1194      ASSIGN "HELP"+W$,5,X8
1196      PRINT #5,1;N4+1,G$
1198      DOEND
1200      FOR K3=1 TO N2
1202      READ #A[K2],K3+1;D6#[1],D6#[2],A1[K3],B,(FOR K1=1 TO B,F2#[&
K1])
1204      PRINT #5,K3+1;D6#[1],D6#[2],A1[K3],B,(FOR K1=1 TO B,F2#[K1]&
)
1206      NEXT K3
1208      P1=N1,P2=A[K2],P3=P4=N2+2,P5=5
1210      GOSUB 9000
1212      IF R3=1 THEN 1266
1214      ASSIGN *,A[K2]
1216      PURGE X1,D2#[K2]+W$
1218      CREATE X2,D2#[K2]+W$,N4+25,319
1220      ASSIGN D2#[K2]+W$,A[K2],X3
1222      PRINT #A[K2],1;D2#[K2],N1+1,N4+25,N2,D7#[1],D7#[3],D7#[2]
1224      FOR K3=1 TO N2
1226      READ #5,K3+1;D6#[1],D6#[2],A1[K3],B,(FOR K1=1 TO B,F2#[K1])
1228      PRINT #A[K2],K3+1;D6#[1],D6#[2],A1[K3],B,(FOR K1=1 TO B,F2#[&
[K1])
1230      NEXT K3
1232      FOR K3=P3 TO N1
1234      IF K3<>R0 THEN DO
1236      READ #P5,K3;Q,(FOR K1=1 TO 22,F#[K1]),T,(FOR K1=1 TO T,&
T1#[K1],T2#[K1],T3#[K1])
1238      PRINT #P2,K3;Q,(FOR K1=1 TO 22,F#[K1]),T,(FOR K1=1 TO T,&
T1#[K1],T2#[K1],T3#[K1])
1240      DOEND
1242      NEXT K3
1244      DOEND
1246      ELSE DO

```

```

1248     P1=N1, P2=P5=A[K2], P3=P4=N2+2
1250     GOSUB 9000
1252     IF R3=1 THEN 1266
1254     PRINT #A[K2], 1; D2#[K2], N1+1, N4, N2, D7#[ 1 ], D7#[ 3 ], D7#[ 2 ]
1256     DOEND
1258     REM.....
1260     REM  LECTURA DE DATOS ACADEMICOS ACTUALES Y MODIFICACION DE
1262     REM  ESTOS MEDIANTE LAMADA A SUBROUTINA #2
1264     REM.....
1266     IF R1=0 THEN READ #6, R2; Q, (FOR K1=1 TO 22, F#[K1]), T, (FOR K1=1&
TO T, F3#[K1], F4#[K1], F5#[K1])
1268     ELSE READ #7, R2; Q, (FOR K1=1 TO 22, F#[K1]), T, (FOR K1=1 TO T, F3#[&
K1], F4#[K1], F5#[K1])
1270     READ #P2, B1+1; D6#[ 1 ], D6#[ 2 ], A1[ 1 ], B, (FOR K1=1 TO B, F2#[K1])
1272     FOR K3=1 TO T
1274         IF F3#[K3, 1; 4]=D2#[K2] THEN DO
1276             IF R3=0 THEN DO
1278                 F3#[K3, 6, 6]="1"
1280                 H3#=""
1282                 FOR K1=1 TO B
1284                     H3#[K1*3-2; 3]="000"
1286                 NEXT K1
1288             DOEND
1290             GOSUB 9300
1292             GOTO 1320
1294         DOEND
1296     NEXT K3
1298     T=T+1
1300     CONVERT B1 TO B1$
1302     F3#[ T ]=D2#[ K2 ]+B1$+"0"
1304     F4#[ T ]=H3#=""
1306     FOR K1=1 TO B
1308         F4#[ T, 1+(K1-1)*2; 1 ]="0"
1310         F4#[ T, K1*2; 1 ]="T"
1312         H3#[ K1*3-2; 3 ]="000"

```

```

1314 NEXT K1
1316 GOSUB 9300
1318 F5#[T]="0"
1320 IF R3=1 THEN DO
1322     FOR K1=1 TO T
1324         IF F3#[K1,1;4]=H1#[1;4] THEN DO
1326             FOR K4=1 TO LEN(F4#[K1])/2
1328                 IF F4#[K1,2*K4;1]<>H2#[2*K4;1] THEN DO
1330                     IF H2#[2*K4;1]<"U" THEN H2#[2*K4-1;2]=F4#[K1,2*K4-1;2&
]
1332         DOEND
1334     NEXT K4
1336 DOEND
1338 NEXT K1
1340 DOEND
1342 REM.....
1344 REM REINTRODUCCION DE LOS DATOS ACADEMICOS (MODIFICADOS O NO).
1346 REM EN FICHEROS CORRESPONDIENTES
1348 REM.....
1350 IF R1=0 THEN PRINT #6,R2;0,(FOR K1=1 TO 22,F#[K1]),T,(FOR K1=1&
TO T,F3#[K1],F4#[K1],F5#[K1])
1352 ELSE PRINT #7,R2;0,(FOR K1=1 TO 22,F#[K1]),T,(FOR K1=1 TO T,F3#&
[K1],F4#[K1],F5#[K1])
1354 IF R3=1 THEN PRINT #P2,R0;B1,(FOR K1=1 TO 22,F#[K1]),1,H1#,H2#,&
H3#
1356 ELSE PRINT #P2,R0;B1,(FOR K1=1 TO 22,F#[K1]),1,F3#[K3],F4#[K3],&
H3#
1358 NEXT K2
1360 REM.....
1362 REM SE EXPLORA SI HA TERMINADO EL COMANDO. SI SE DA EL CASO
1364 REM VOLCADO DE FICHERO 'TEMP' EN FICHERO EXPEDIENTES. CIERRE
1366 REM DE FICHEROS CORRESPONDIENTES. DEVOLUCION DE CONTROL.
1368 REM.....
1370 PRINT '27"1"
1372 PRINT '27"A"'27"&a22r5C"'27&

```

```

      "&dC
      . "'27"&d@"
1374 PRINT '27"A";TAB(7);"Otro alumno (SI/NO)?";
1376 ENTER 255,W6,R#
1378 PRINT '27"m"
1380 IF R#="S" OR R#="SI" THEN DO
1382   FOR K1=1 TO 4
1384     ASSIGN *,K1
1386   NEXT K1
1388   IF N=100 THEN DO
1390     GOSUB 9700
1392     GOTO 594
1394   DOEND
1396 ELSE GOTO 596
1398 DOEND
1400 IF R#="N" OR R#="NO" THEN DO
1402   GOSUB 9700
1404   IF P6=8 THEN DO
1406     READ #5,1;N3,G#
1408     PRINT #5,1;N3,G1#
1410   DOEND
1412   FOR K1=1 TO 8
1414     ASSIGN *,K1
1416   NEXT K1
1418   IF P6=8 THEN DO
1420     PURGE X7,G#+W#
1422   DOEND
1424 DOEND
1426 ELSE DO
1428   PRINT '27"G''27"A";
1430   GOTO 1370
1432 DOEND
1434 CHAIN "ARCHALUM"+W#

```

&

DELTA2

```

1500 REM*****
1502 REM
1504 REM      MODULO 3.      RENUNCIA DE CONVOCATORIAS
1506 REM
1508 REM*****
1510 COM W#[*]
1512 DIM A#[3,30],A2#[3,1],A3#[3,1],A4#[3,1],A5#[3,1],A6#[3,1]
1514 DIM B#[22,30],C[15],D#[80],D6#[2,30],D7#[3,30],E1#[6],E2#[31]
1516 DIM E6#[15,1],E7#[15,1],E8#[15,1]
1518 DIM F#[15,58],F2#[15,58],F3#[10,6],F4#[10,31],F5#[10,200]
1520 DIM F6#[15,1],F7#[15,9],F8#[15,1],F9#[10,45]
1522 DIM G#[8],H#[2],I1[20],I2[20],N1[10],N2[10],V#[3,1],Z#[3,1]
1524 REM.....
1526 REM  APERTURA DE FICHERO DE EXPEDIENTES(G#) ,FICHERO 'HELP' Y
1528 REM  FICHERO @@@@ (ASCII) DECLARADO PARA LINPUT #9.
1530 REM.....
1532 FILES *,*,*,*,*,*,*,*,*
1534 ASSIGN "HELP"+W#,5,A
1536 READ #5,1;A,G#
1538 ASSIGN #,5
1540 SYSTEM X,"FILE @@@@=$STDINX;REC=-1"
1542 ASSIGN "@@@@",9,X
1544 ASSIGN G#+W#,6,A
1546 REM.....
1548 REM  FORMATO PAGINA 1. INTRODUCCION DE DATOS DEL ALUMNO Y
1550 REM  LOCALIZACION EN EL FICHERO DE EXPEDIENTES
1552 REM.....
1554 READ #6,1;H1
1556 PRINT '27"h''27"J"
1558 PRINT '27"A";"Primer apellido";TAB(26);"Segundo apellido";TAB(51)&
    ;"Nombre"
1560 PRINT '27"&dC          "'27"&d@    "'27&
    "&dC                    "'27"&d@    "'27&
    "&dC                    "'27"&d@";

```

```

1562 PRINT '27"A"'27"A";
1564 PRINT '27"[
      "C          "'27" ] "'27&
      "'27" ] "'27"[          "'27" ]";
1566 FOR K=1 TO 3
1568 PRINT '27"W";
1570 PRINT (FOR K1=1 TO K-1,(FOR K2=1 TO 25,'27"C";)););
1572 ENTER 255,A,A#[K]
1574 A#[K]=UPS$(DEB$(A#[K]))
1576 PRINT '27"A"'27"X"
1578 NEXT K
1580 IF H1=3 THEN H1=4
1582 ON END #6 THEN 1608
1584 L=2,H=H1
1586 M=INT((L+H)/2)
1588 IF M=1 THEN 1608
1590 READ #6,M;A,(FOR K=1 TO 3,B#[K])
1592 IF B#[1]+B#[2]+B#[3]>=A#[1]+A#[2]+A#[3] THEN 1600
1594 IF L>=H THEN 1608
1596 L=M+1
1598 GOTO 1586
1600 IF B#[1]+B#[2]+B#[3]=A#[1]+A#[2]+A#[3] THEN 1614
1602 IF L>=H THEN 1608
1604 H=M-1
1606 GOTO 1586
1608 PRINT '27"&a22r0C"'27&
      "&dC
      "'27"@", '27"A", '27"A"
1610 PRINT &
      "No hay nadie matriculado con ese nombre. Otro alumno (SI/NO)?";
1612 GOTO 1964
1614 REM.....
1616 REM APERTURA DE FICHEROS TIPO 'C' Y LISTADO DE ASIGNATURAS .
1618 REM.....
1620 READ #6,M;B,(FOR K=1 TO 22,B#[K]),T,(FOR K=1 TO T,F3#[K],F4#[K],&
      F5#[K])

```

```

1622 PRINT LIN(3);"Asignaturas matriculadas",TAB(59);"Convocatorias"
1624 PRINT TAB(61);"FEB";TAB(65);"JUN";TAB(69);"SEP"
1626 PRINT (FOR J=1 TO 72,"-";);
1628 I=0
1630 FOR K1=1 TO T
1632   ASSIGN F3#[K1,1,4]+W#,1,D
1634   READ #1,1;N#,N1[K1],N4,N2[K1],D7#[1],D7#[2],D7#[3]
1636   CONVERT F3#[K1,5;1] TO A
1638   READ #1,A+1;D6#[1],D6#[2],A1,B1,(FOR K=1 TO B1,F2#[K])
1640   FOR K=1 TO B1
1642     IF F4#[K1,1+(K-1)*2;1]<>"0" AND F4#[K1,K*2;1]>="U" THEN DO
1644       L=N2[K1]+2,H=N1[K1]
1646       M1=INT((L+H)/2)
1648       READ #1,M1;Q,A#[1],A#[2],A#[3]
1650       IF A#[1]+A#[2]+A#[3]>=B#[1]+B#[2]+B#[3] THEN 1658
1652       IF L>=H THEN 1648
1654       L=M1+1
1656       GOTO 1646
1658       IF A#[1]+A#[2]+A#[3]=B#[1]+B#[2]+B#[3] THEN 1666
1660       IF L>=H THEN 1648
1662       H=M1-1
1664       GOTO 1646
1666       READ #1,M1;Q,(FOR K2=1 TO 22,B#[K2]),Q1,E1#,E2#,F9#[K1]
1668       I=I+1,I1[I]=K1,I2[I]=K
1670       A#[1]=A#[2]=A#[3]=V#[1]=V#[2]=V#[3]=" "
1672       IF F4#[K1,K*2;1]="Z" THEN DO
1674         A#[2]=A#[3]="X"
1676         IF F9#[K1,3*K-1;1]<>"0" THEN V#[2]="Y"
1678         IF F9#[K1,3*K;1]<>"0" THEN V#[3]="Y"
1680       DOEND
1682       IF F4#[K1,K*2;1]="Y" THEN DO
1684         A#[2]="X"
1686         IF F9#[K1,3*K-1;1]<>"0" THEN V#[2]="Y"
1688       DOEND
1690       IF F4#[K1,K*2;1]="X" THEN DO

```

```

1692         A#[ 3 ]="X"
1694         IF F9#[ K1,3*K;1 ]<>"0" THEN V#[ 3 ]="Y"
1696     DOEND
1698     IF F4#[ K1,K*2;1 ]="W" THEN DO
1700         A#[ 1 ]="X"
1702         IF F9#[ K1,3*K-2;1 ]<>"0" THEN V#[ 1 ]="Y"
1704     DOEND
1706     IF F4#[ K1,K*2;1 ]="V" THEN DO
1708         A#[ 1 ]=A#[ 3 ]="X"
1710         IF F9#[ K1,3*K-2;1 ]<>"0" THEN V#[ 1 ]="Y"
1712         IF F9#[ K1,3*K;1 ]<>"0" THEN V#[ 3 ]="Y"
1714     DOEND
1716     IF F4#[ K1,K*2;1 ]="U" THEN DO
1718         A#[ 1 ]=A#[ 2 ]="X"
1720         IF F9#[ K1,3*K-2;1 ]<>"0" THEN V#[ 1 ]="Y"
1722         IF F9#[ K1,3*K-1;1 ]<>"0" THEN V#[ 2 ]="Y"
1724     DOEND
1726     CONVERT I TO H#
1728     IF LEN(H#)=1 THEN H#=" "+H#
1730     PRINT H#;". " ;F2#[ K ];TAB(63);A#[ 1 ];TAB(67);A#[ 2 ];TAB(71);A#&
    [ 3 ]
1732     E6#[ I ]=V#[ 1 ],E7#[ I ]=V#[ 2 ],E8#[ I ]=V#[ 3 ]
1734     F6#[ I ]=A#[ 1 ],F7#[ I ]=A#[ 2 ],F8#[ I ]=A#[ 3 ]
1736     F#[ I ]=F2#[ K ]
1738     DOEND
1740     NEXT K
1742     ASSIGN *,1
1744     NEXT K1
1746     PRINT (FOR J=1 TO 72,"-");
1748     REM.....
1750     REM SELECCION DE LAS ASIGNATURAS DE RENUNCIA
1752     REM.....
1754     A2#[ 1 ]=A6#[ 2 ]=A6#[ 3 ]="W",A2#[ 2 ]=A4#[ 1 ]=A4#[ 3 ]="Y",A3#[ 2 ]=A3#[ 3 ]=&
        "Z"
1756     A2#[ 3 ]=A4#[ 2 ]=A6#[ 1 ]="X",A3#[ 1 ]=A5#[ 2 ]="U",A5#[ 1 ]=A5#[ 3 ]="V"

```

```

1758 PRINT '27"&a22r0C"'27&
      "&dC
      "'27"0", '27"A", '27"A"
1760 D$=&
      "Introduzca numero(s) correspondiente(s) a asignatura(s) de renunci
ia"
1762 PRINT '27"G";D$;'27"A"
1764 FOR W0=1 TO 9999
1766 NEXT W0
1768 PRINT (FOR K4=1 TO 71, " ";);". ";'27"G";'27"A";
1770 INPUT "Numero(s) correspondiente(s) ?";
1772 FOR K1=1 TO 15
1774   IF BUF(0)=4 AND K1=1 THEN 1958
1776   IF BUF(0)=4 THEN 1810
1778   IF BUF(0)<>5 THEN DO
1780     PRINT '27"M", '27"A", '27"A"
1782     GOTO 1768
1784   DOEND
1786   INPUT :C[K1]:
1788   IF C[K1]<1 OR C[K1]>I THEN DO
1790     D$=&
     "No encuentro asignatura que corresponda a este numero
!"
1792     PRINT '27"&a21r0C";'7;D$
1794     ENTER 3,A,D
1796     PRINT '27"A", '27"A"
1798     GOTO 1760
1800   DOEND
1802 NEXT K1
1804 REM.....
1806 REM   FORMATO PAGINA 2. INTRODUCCION DE LOS DATOS DE RENUNCIA
1808 REM.....
1810 PRINT '27"&a6r0C"'27"J"'27&
      "&dC
      "'27"&d0"

```

```

1812 PRINT LIN(1),TAB(20);"CONVOCATORIAS DE RENUNCIA"
1814 PRINT TAB(25);"FEB";TAB(30);"JUN";TAB(35);"SEP"
1816 PRINT TAB(26),'27"&dC "'27"&d@";SPA(4);'27"&dC "'27"&d@";SPA(4);&
    '27"&dC "'27"&d@", '27"G"
1818 PRINT '27"A";
1820 FOR K=1 TO K1-1
1822   PRINT '27"&a5r0C"
1824   PRINT F#[C[K]];TAB(62);F6#[C[K]];TAB(66);F7#[C[K]];TAB(70);F8#[&
    C[K]];TAB(71);"."
1826   PRINT '27"&a22r0C"'27&
    "&dC                                     &
        "'27"@"
1828   PRINT '27"&a21r0C";&
    "Introduzca 'X' en la casilla de renuncia, sino pulse 'ESPACIO' &
    ."
1830   PRINT '27"&a10r26C"'27"[ "; " ";'27"]"; " ";'27"[ "; " ";'27"]";&
    " ";'27"[ "; " ";'27"]";
1832   Z#[1]="A"
1834   FOR K3=1 TO 3
1836     PRINT '27"A";'27"W";
1838     IF Z#[K3]<>" " AND Z#[K3]<>" " AND Z#[K3]<>"X" THEN K3=K3-1
1840     ELSE DO
1842       IF K3=3 THEN 1852
1844     DOEND
1846     PRINT '27"&a10r26C";
1848     PRINT (FOR K2=1 TO K3,'27"C"'27"C"'27"C"'27"C"'27"C";);
1850     LINPUT #9;Z#[K3+1]
1852     PRINT '27"A";'27"X"
1854   NEXT K3
1856   IF (E6#[C[K]]="Y" AND Z#[1]="X") OR (E7#[C[K]]="Y" AND Z#[2]=&
    "X") OR (E8#[C[K]]="Y" AND Z#[3]="X") THEN DO
1858     PRINT '27"&a21r0C";'7;&
    "Error ! Intenta renunciar a convocatoria ya calificada &
    ."
1860     ENTER 3,X,Z#[1]

```

```

1862     GOTO 1828
1864     DOEND
1866     IF (F6#[C[K]]<>"X" AND Z#[1]="X") OR (F7#[C[K]]<>"X" AND Z#[2]=&
"X") OR (F8#[C[K]]<>"X" AND Z#[3]="X") THEN DO
1868     PRINT '27"&a21r0C";'7;&
"Error ! Intenta renunciar a convocatoria no matriculada      &
.
1870     ENTER 3,X,Z#[1]
1872     GOTO 1828
1874     DOEND
1876     REM.....
1878     REM  MODIFICACION DE LOS DATOS DE RENUNCIA POR ASIGNATURA      .
1880     REM  E INTRODUCCION GLOBAL EN EL FICHERO DE EXPEDIENTES      .
1882     REM.....
1884     FOR J=1 TO 3
1886     IF Z#[J]="X" THEN DO
1888     IF F4#[ I1[C[K]], I2[C[K]]*2;1 ]=A2#[J] THEN F4#[ I1[C[K]], I2[C&
[K]]*2;1 ]="T"
1890     IF F4#[ I1[C[K]], I2[C[K]]*2;1 ]=A3#[J] THEN F4#[ I1[C[K]], I2[C&
[K]]*2;1 ]=A4#[J]
1892     IF F4#[ I1[C[K]], I2[C[K]]*2;1 ]=A5#[J] THEN F4#[ I1[C[K]], I2[C&
[K]]*2;1 ]=A6#[J]
1894     CONVERT F4#[ I1[C[K]], 1+(I2[C[K]]-1)*2;1 ] TO T2
1896     T2=T2-1
1898     CONVERT T2 TO F4#[ I1[C[K]], 1+(I2[C[K]]-1)*2;1 ]
1900     DOEND
1902     NEXT J
1904     NEXT K
1906     PRINT #6,M;B,(FOR K=1 TO 22,B#[K]),T,(FOR K=1 TO T,F3#[K],F4#[K],&
F5#[K])
1908     REM.....
1910     REM  LOCALIZACION DEL ALUMNO E INTRODUCCION DE LOS DATOS      .
1912     REM  MODIFICADOS DE CONVOCATORIA EN LOS FICHEROS TIPO C      .
1914     REM.....
1916     FOR K=1 TO T

```

```

1918 ASSIGN F3#[K,1,4]+W#,1,D
1920 L=N2[K]+2,H=N1[K]
1922 IF L<=H THEN DO
1924     M=INT((L+H)/2)
1926     READ #1,M;B,A#[1],A#[2],A#[3]
1928     IF A#[1]+A#[2]+A#[3]>=B#[1]+B#[2]+B#[3] THEN 1936
1930     IF L>=H THEN 1950
1932     L=M+1
1934     GOTO 1924
1936     IF A#[1]+A#[2]+A#[3]=B#[1]+B#[2]+B#[3] THEN 1944
1938     IF L>=H THEN 1950
1940     H=M-1
1942     GOTO 1924
1944     PRINT #1,M;B,(FOR K1=1 TO 22,B#[K1]),1,F3#[K],F4#[K],F9#[K]
1946 DOEND
1948 ASSIGN *,1
1950 NEXT K
1952 REM.....
1954 REM DEVOLUCION DE CONTROL Y CIERRE DE FICHEROS #6 Y #9 .
1956 REM.....
1958 PRINT '27"&a21r-5C"
1960 PRINT (FOR K4=1 TO 71," ");". ";'27"G";'27"A";
1962 PRINT "Otro alumno (SI/NO)?";
1964 ENTER 255,A,H#
1966 IF H#="SI" OR H#="S" THEN 1556
1968 IF H#="NO" OR H#="N" THEN DO
1970     ASSIGN *,6
1972     ASSIGN *,9
1974 DOEND
1976 ELSE GOTO 1958
1978 CHAIN "ARCHALUM"+W#

```

>

DELTA3

```

2000 REM*****
2002 REM
2004 REM      MODULO 4.      CAMBIO DE CONVOCATORIAS
2006 REM
2008 REM*****
2010 COM W[*]
2012 DIM A#[ 3,30 ],A2#[ 3, 1 ],A3#[ 3, 1 ],A4#[ 3, 1 ],A5#[ 3, 1 ],A6#[ 3, 1 ]
2014 DIM B#[ 22,30 ],C[ 15 ],D#[ 80 ],D6#[ 2,30 ],D7#[ 3,30 ],F#[ 15,58 ]
2016 DIM E1#[ 6 ],E2#[ 31 ],E6#[ 15, 1 ],E7#[ 15, 1 ],E8#[ 15, 1 ]
2018 DIM F2#[ 15,58 ],F3#[ 10, 6 ],F4#[ 10,31 ],F5#[ 10,200 ],F6#[ 15, 1 ]
2020 DIM F7#[ 15, 9 ],F8#[ 15, 1 ],F9#[ 10,45 ],G#[ 8 ],G6#[ 3, 1 ],G7#[ 3, 1 ]
2022 DIM H#[ 2 ],I1[ 20 ],I2[ 20 ],N1[ 10 ],N2[ 10 ],V#[ 3, 1 ],Z#[ 3,3 ]
2024 REM.....
2026 REM      APERTURA DE FICHERO DE EXPEDIENTES(G#) ,FICHERO 'HELP' Y
2028 REM      FICHERO QQQQ(ASCII) DECLARADO PARA LINPUT #9.
2030 REM.....
2032 FILES *,*,*,*,*,*,*,*,*
2034 ASSIGN "HELP"+W#,5,A
2036 READ #5,1;A,G#
2038 ASSIGN *,5
2040 SYSTEM X,"FILE QQQQ=#STDIN;REC=-3"
2042 ASSIGN "QQQQ",9,X
2044 ASSIGN G#+W#,6,A
2046 REM.....
2048 REM      FORMATO PAGINA 1. INTRODUCCION DE DATOS DEL ALUMNO Y
2050 REM      LOCALIZACION EN EL FICHERO DE EXPEDIENTES.
2052 REM.....
2054 READ #6,1;H1
2056 PRINT '27"h"'27"J"
2058 PRINT '27"A':"Primer apellido";TAB(26);"Segundo apellido";TAB(51)&
; "Nombre"
2060 PRINT '27"&dC          "'27"&d@    "'27&
"&dC          "'27"&d@    "'27&
"&dC          "'27"&d@";

```

```

2062 PRINT '27"A"'27"A";
2064 PRINT '27"[
      "C          "'27" ]    "'27&
      "'27" ]    "'27"[          "'27" ]";
2066 FOR K=1 TO 3
2068   PRINT '27"W";
2070   PRINT (FOR K1=1 TO K-1,(FOR K2=1 TO 25,'27"C";)););
2072   ENTER 255,A,A#[K]
2074   A#[K]=UPS$(DEB$(A#[K]))
2076   PRINT '27"A"'27"X"
2078 NEXT K
2080 IF H1=3 THEN H1=4
2082 ON END #6 THEN 2108
2084 L=2,H=H1
2086 M=INT((L+H)/2)
2088 IF M=1 THEN 2108
2090 READ #6,M;A,(FOR K=1 TO 3,B#[K])
2092 IF B#[1]+B#[2]+B#[3]>A#[1]+A#[2]+A#[3] THEN 2108
2094 IF L>=H THEN 2108
2096 L=M+1
2098 GOTO 2086
2100 IF B#[1]+B#[2]+B#[3]=A#[1]+A#[2]+A#[3] THEN 2114
2102 IF L>=H THEN 2108
2104 H=M-1
2106 GOTO 2086
2108 PRINT '27"&a22r0C"'27&
      "&dC
      "'27"@", '27"A", '27"A"
2110 PRINT &
      "No hay nadie matriculado con ese nombre. Otro alumno (SI/NO)?";
2112 GOTO 2518
2114 REM.....
2116 REM   APERTURA DE FICHEROS TIPO C Y LISTADO DE ASIGNATURAS
2118 REM.....
2120 READ #6,M;B,(FOR K=1 TO 22,B#[K]),T,(FOR K=1 TO T,F3#[K],F4#[K],&
      F5#[K])

```

```

2122 PRINT LIN(3);"Asignaturas matriculadas",TAB(59);"Convocatorias"
2124 PRINT TAB(61);"FEB";TAB(65);"JUN";TAB(69);"SEP"
2126 PRINT (FOR J=1 TO 72,"-"););
2128 I=0
2130 FOR K1=1 TO T
2132   ASSIGN F3#[K1,1,4]+W$,1,D
2134   READ #1,1;N$,N1[K1],N4,N2[K1],D7#[1],D7#[2],D7#[3]
2136   CONVERT F3#[K1,5;1] TO A
2138   READ #1,A+1;D6#[1],D6#[2],A1,B1,(FOR K=1 TO B1,F2#[K])
2140   FOR K=1 TO B1
2142     IF F4#[K1,1+(K-1)*2;1]>"0" AND F4#[K1,K*2;1]>="U" THEN DO
2144       L=N2[K1]+2,H=N1[K1]
2146       M1=INT((L+H)/2)
2148       READ #1,M1;Q,A#[1],A#[2],A#[3]
2150       IF A#[1]+A#[2]+A#[3]>=B#[1]+B#[2]+B#[3] THEN 2158
2152       IF L>=H THEN 2148
2154       L=M1+1
2156       GOTO 2146
2158       IF A#[1]+A#[2]+A#[3]=B#[1]+B#[2]+B#[3] THEN 2166
2160       IF L>=H THEN 2148
2162       H=M1-1
2164       GOTO 2146
2166       READ #1,M1;Q,(FOR K2=1 TO 22,B#[K2]),Q1,E1$,E2$,F9#[K1]
2168       I=I+1,I1[I]=K1,I2[I]=K
2170       A#[1]=A#[2]=A#[3]=V#[1]=V#[2]=V#[3]=" "
2172       IF F4#[K1,K*2;1]="Z" THEN DO
2174         A#[2]=A#[3]="X"
2176         IF F9#[K1,3*K-1;1]>"0" THEN V#[2]="Y"
2178         IF F9#[K1,3*K;1]>"0" THEN V#[3]="Y"
2180       DOEND
2182       IF F4#[K1,K*2;1]="Y" THEN DO
2184         A#[2]="X"
2186         IF F9#[K1,3*K-1;1]>"0" THEN V#[2]="Y"
2188       DOEND
2190       IF F4#[K1,K*2;1]="X" THEN DO

```

```

2192         A#[3]="X"
2194         IF F9#[K1,3*K;1]<>"0" THEN V#[3]="Y"
2196     DOEND
2198     IF F4#[K1,K*2;1]="W" THEN DO
2200         A#[1]="X"
2202         IF F9#[K1,3*K-2;1]<>"0" THEN V#[1]="Y"
2204     DOEND
2206     IF F4#[K1,K*2;1]="V" THEN DO
2208         A#[1]=A#[3]="X"
2210         IF F9#[K1,3*K-2;1]<>"0" THEN V#[1]="Y"
2212         IF F9#[K1,3*K;1]<>"0" THEN V#[3]="Y"
2214     DOEND
2216     IF F4#[K1,K*2;1]="U" THEN DO
2218         A#[1]=A#[2]="X"
2220         IF F9#[K1,3*K-2;1]<>"0" THEN V#[1]="Y"
2222         IF F9#[K1,3*K-1;1]<>"0" THEN V#[2]="Y"
2224     DOEND
2226     CONVERT I TO H#
2228     IF LEN(H#)=1 THEN H#=" "+H#
2230     PRINT H#;"  ";F2#[K];TAB(63);A#[1];TAB(67);A#[2];TAB(71);A#&
    [3]
2232     E6#[I]=V#[1],E7#[I]=V#[2],E8#[I]=V#[3]
2234     F6#[I]=A#[1],F7#[I]=A#[2],F8#[I]=A#[3]
2236     F#[I]=F2#[K]
2238     DOEND
2240     NEXT K
2242     ASSIGN *,1
2244     NEXT K1
2246     PRINT (FOR J=1 TO 72,"-");
2248     REM.....
2250     REM SELECCION DE ASIGNATURAS DE CAMBIO DE CONVOCATORIA
2252     REM.....
2254     A2#[1]=A6#[2]=A6#[3]="W",A2#[2]=A4#[1]=A4#[3]="Y",A3#[2]=A3#[3]=&
        "Z"
2256     A2#[3]=A4#[2]=A6#[1]="X",A3#[1]=A5#[2]="U",A5#[1]=A5#[3]="V"

```

```

2258 PRINT '27"&a22r0C"'27&
      "&dC
      ''27"0", '27"A", '27"A"
2260 D$=&
      "Introduzca numero(s) correspond. a asignatura(s) de cambio de con&
vocat."
2262 PRINT '27"G";D$;'27"A"'27"A"
2264 FOR W0=1 TO 9999
2266 NEXT W0
2268 PRINT (FOR K4=1 TO 71," ");!";'27"G";'27"A";
2270 INPUT "Numero(s) correspondiente(s) ?":
2272 FOR K1=1 TO 15
2274     IF BUF(0)=4 AND K1=1 THEN 2512
2276     IF BUF(0)=4 THEN 2312
2278     IF BUF(0)<>5 THEN DO
2280         PRINT '27"M", '27"A", '27"A"
2282         GOTO 2268
2284     DOEND
2286     INPUT :C(K1):
2288     IF C(K1)<1 OR C(K1)>I THEN DO
2290         D$=&
          "No encuentro asignatura que corresponda a este numero
          !"
2292         PRINT '27"&a21r0C";'7;D$
2294         ENTER 3,A,D
2296         PRINT '27"A", '27"A"
2298         GOTO 2260
2300     DOEND
2302 NEXT K1
2304 REM.....
2306 REM   FORMATO PAGINA 2. INTRODUCCION DE LOS DATOS DE CAMBIO
2308 REM   DE CONVOCATORIA
2310 REM.....
2312 PRINT '27"&a6r0C"'27"J"'27&
      "&dC

```

```

                "'27"&d@"
2314 FOR K=1 TO K1-1
2316   PRINT "'27"&a5r0C"
2318   PRINT F#[C[K]];TAB(62);F6#[C[K]];TAB(66);F7#[C[K]];TAB(70);F8#[
      C[K]];TAB(71);"."
2320   PRINT "'27"&a10r1C"
2322   PRINT "DOY DE BAJA LA CONVOCATORIA DE ";"'27"&dC    "'27"&d@";
2324   PRINT " Y DOY DE ALTA LA DE ";"'27"&dC    "'27"&d@"
2326   PRINT "'27"&a22r0C"'27&
      &dC
                "'27"@"
2328   PRINT "'27"&a21r0C"
2330   PRINT &
      "Introduzca FEB, JUN o SEP donde corresponda
      "
2332   PRINT "'27"&a10r31C";LIN(-1);"'27"[    "'27" ]";LIN(-1);
2334   PRINT "                ";"'27"A"'27"[    "'27" ]"
2336   PRINT "'27"W"
2338   PRINT "'27"&a11r31C";
2340   LINPUT #9;Z#[1]
2342   PRINT "'27"X"
2344   IF (E6#[C[K]]="Y" AND Z#[1]="FEB") OR (E7#[C[K]]="Y" AND Z#[1]=&
      "JUN") OR (E8#[C[K]]="Y" AND Z#[1]="SEP") THEN DO
2346     PRINT "'27"&a22r0C";'7;&
      "ERROR ! Intenta dar de baja una convocatoria ya calificada."
2348     ENTER 3,A,Z#[1]
2350     GOTO 2328
2352   DOEND
2354   IF (Z#[1]="FEB" AND F6#[C[K]]<>"X") OR (Z#[1]="JUN" AND F7#[C[K]
      ]<>"X") OR (Z#[1]="SEP" AND F8#[C[K]]<>"X") THEN DO
2356     PRINT "'27"&a22r0C";'7;&
      "ERROR ! Intenta dar de baja una convocatoria no matriculada"
2358     ENTER 3,A,Z#[1]
2360     GOTO 2328
2362   DOEND

```

```

2364   IF Z#[1]<>"FEB" AND Z#[1]<>"JUN" AND Z#[1]<>"SEP" THEN DO
2366     IF Z#[1]=" " THEN 2458
2368     PRINT '27"&a21r0C";&
        "Introduzca datos correctos (FEB, JUN o SEP)                &
        "
2370     GOTO 2332
2372   DOEND
2374   PRINT '27"W"
2376   PRINT '27"&a11r55C";
2378   LINPUT #9;Z#[2]
2380   PRINT '27"X"
2382   IF Z#[1]=Z#[2] THEN 2332
2384   IF (E6#[C[K]]="Y" AND Z#[2]="FEB") OR (E7#[C[K]]="Y" AND Z#[2]=&
        "JUN") OR (E8#[C[K]]="Y" AND Z#[2]="SEP") THEN DO
2386     PRINT '27"&a22r0C";'7;&
        "ERROR ! Intenta dar de alta una convocatoria ya calificada"
2388     ENTER 3,A,Z#[2]
2390     GOTO 2328
2392   DOEND
2394   IF (Z#[2]="FEB" AND F6#[C[K]]="X") OR (Z#[2]="JUN" AND F7#[C[K]]&
        ]="X") OR Z#[2]="SEP" AND F8#[C[K]]="X" THEN DO
2396     PRINT '27"&a21r0C";'7;&
        "ERROR ! Intenta dar de alta una convocatoria ya matriculada  &
        "
2398     ENTER 3,A,Z#[2]
2400     GOTO 2328
2402   DOEND
2404   IF Z#[2]<>"FEB" AND Z#[2]<>"JUN" AND Z#[2]<>"SEP" THEN DO
2406     IF Z#[2]=" " THEN 2458
2408     PRINT '27"&a21r0C";&
        "Introduzca datos correctos (FEB, JUN o SEP)                &
        "
2410     GOTO 2374
2412   DOEND
2414   REM.....

```

```

2416 REM CAMBIO DE LAS CONVOCATORIS E INTRODUCCION EN ARCHIVO .
2418 REM DE EXPEDIENTES .
2420 REM.....
2422 IF Z#[ 1]="FEB" THEN G6#[ 1]="Y",G6#[ 2]="X",G6#[ 3]="Z"
2424 IF Z#[ 1]="JUN" THEN G6#[ 1]="W",G6#[ 2]="X",G6#[ 3]="V"
2426 IF Z#[ 1]="SEP" THEN G6#[ 1]="W",G6#[ 2]="Y",G6#[ 3]="U"
2428 IF Z#[ 2]="FEB" THEN G7#[ 1]="W",G7#[ 2]="U",G7#[ 3]="V"
2430 IF Z#[ 2]="JUN" THEN G7#[ 1]="Y",G7#[ 2]="Z",G7#[ 3]="U"
2432 IF Z#[ 2]="SEP" THEN G7#[ 1]="X",G7#[ 2]="Z",G7#[ 3]="V"
2434 IF F4#[ I1[ C[K]], I2[ C[K]]*2;1 ]>="W" AND F4#[ I1[ C[K]], I2[ C[K]]*2;&
1 ]<="Y" THEN DO
2436 FOR J=1 TO 2
2438 FOR J1=1 TO 3
2440 IF G6#[ J]=G7#[ J1 ] THEN F4#[ I1[ C[K]], I2[ C[K]]*2;1 ]=G7#[ J1 ]
2442 NEXT J1
2444 NEXT J
2446 DOEND
2448 ELSE DO
2450 FOR J1=1 TO 3
2452 IF G6#[ 3]=G7#[ J1 ] THEN F4#[ I1[ C[K]], I2[ C[K]]*2;1 ]=G7#[ J1 ]
2454 NEXT J1
2456 DOEND
2458 NEXT K
2460 PRINT #6,M;B,(FOR K=1 TO 22,B#[K]),T,(FOR K=1 TO T,F3#[K],F4#[K],&
F5#[K])
2462 REM.....
2464 REM LOCALIZACION DEL ALUMNO E INTRODUCCION DE LOS DATOS .
2466 REM MODIFICADOS DE CONVOCATORIA EN LOS FICHEROS TIPO C .
2468 REM.....
2470 FOR K=1 TO T
2472 ASSIGN F3#[K, 1, 4]+W#, 1, D
2474 L=N2[K]+2,H=N1[K]
2476 IF L<=H THEN DO
2478 M=INT((L+H)/2)
2480 READ #1,M;B,A#[ 1 ],A#[ 2 ],A#[ 3 ]

```

```

2482     IF A#[1]+A#[2]+A#[3]>=B#[1]+B#[2]+B#[3] THEN 2490
2484     IF L>=H THEN 2504
2486     L=M+1
2488     GOTO 2478
2490     IF A#[1]+A#[2]+A#[3]=B#[1]+B#[2]+B#[3] THEN 2498
2492     IF L>=H THEN 2504
2494     H=M-1
2496     GOTO 2478
2498     PRINT #1,M;B,(FOR K1=1 TO 22,B#[K1]),1,F3#[K],F4#[K],F9#[K]
2500     DOEND
2502     ASSIGN *,1
2504     NEXT K
2506     REM.....
2508     REM   DEVOLUCION DE CONTROL Y CIERRE DE FICHEROS #6 Y #9
2510     REM.....
2512     PRINT '27"&a21r-5C"
2514     PRINT (FOR K4=1 TO 71," "):". ":'27"G"':'27"A"
2516     PRINT "Otro alumno (SI/NO)?";
2518     ENTER 255,A,H#
2520     IF H#="SI" OR H#="S" THEN 2056
2522     IF H#="NO" OR H#="N" THEN DO
2524     ASSIGN *,6
2526     ASSIGN *,9
2528     DOEND
2530     ELSE GOTO 2512
2532     CHAIN "ARCHALUM"+W#
>

```

DELTA4

```

3000 REM*****
3002 REM
3004 REM          MODULO 5.      INTRODUCCION  DE  NOTAS
3006 REM
3008 REM*****
3010 COM W#[ * ]
3012 DIM A#[ 22,30 ],B#[ 4, 12 ],C[ 50 ],C1#[ 5, 15 ],C2#[ 4, 10 ],C3#[ 2, 10 ]
3014 DIM C4#[ 3,20 ],D#[ 50 ],D5#[ 3,30 ],D6#[ 2,30 ],D7#[ 3,30 ],F[ 50 ]
3016 DIM F2#[ 15,65 ],F3#[ 45,65 ],G#[ 8 ],H#[ 2 ],J1[ 50, 10 ],J2[ 50, 10 ],N#[ 5 ]
3018 DIM T1#[ 10,6 ],T2#[ 10,31 ],T3#[ 10,200 ],V#[ 3 ],Z#[ 4 ]
3020 B#[ 1 ]=" CARRERA",B#[ 3 ]=" CURSO",B#[ 2 ]=" PLAN",B#[ 4 ]=" CONVOCATORIA"
3022 C1#[ 1 ]=" INDUSTRIAL",C1#[ 2 ]=" TELECOMUNICACION",C1#[ 5 ]=" NAVAL "
3024 C1#[ 4 ]=" OBRAS PUBLICAS",C1#[ 3 ]=" TOPOGRAFIA"
3026 C1#[ 4 ]=" OBRAS PUBLICAS",C1#[ 3 ]=" TOPOGRAFIA"
3028 C2#[ 1 ]=" PRIMERO",C2#[ 2 ]=" SEGUNDO",C2#[ 3 ]=" TERCERO",C2#[ 4 ]=&
"CUARTO"
3030 C3#[ 1 ]=" TRIENAL",C3#[ 2 ]=" CUATRIENAL "
3032 C4#[ 1 ]=" FEBRERO          1",C4#[ 2 ]=" JUNIO                2",C4#[ 3 ]=&
"SEPTIEMBRE          3"
3034 FILES *,*,*,*,*,*,*,*,*
3036 SYSTEM X,"FILE QQQQ=#STDIN;REC=-1"
3038 ASSIGN "QQQQ",9,X
3040 ASSIGN "HELP"+W#,5,X
3042 READ #5,1;X,G#
3044 ASSIGN *,5
3046 ASSIGN G#+W#,6,X
3048 READ #6,1;N5
3050 K5=1
3052 REM.....
3054 REM  FORMATO DE PAGINA 1 E INTRODUCCION DE CLAVE QUE .
3056 REM  QUE SELECCIONA FICHERO Y CONVOCATORIA .
3058 REM.....
3060 PRINT '27"h" '27"J"
3062 PRINT '27"&a0r0C" ;'27&

```

```

      "&dC
      "'27"&d@"
3064 PRINT '27"&a0r0C";&
      "Introduzca clave para seleccionar carrera, plan, curso y convocat&
      oria"
3066 PRINT LIN(2),TAB(8);B#[1];SPA(30);B#[4]
3068 PRINT TAB(8);"-----";SPA(22);"-----"
3070 PRINT (FOR J=1 TO 3,TAB(8);C1#[J];TAB(23);J,SPA(15);C4#[J]);
3072 PRINT (FOR J=4 TO 5,TAB(8);C1#[J];TAB(23);J,LIN(1));
3074 PRINT LIN(1),TAB(8);B#[2];LIN(1);TAB(8);"-----"
3076 PRINT (FOR J=1 TO 2,TAB(8);C3#[J];TAB(23);J,LIN(1));
3078 PRINT LIN(1);TAB(8);B#[3];LIN(1);TAB(8);"-----"
3080 PRINT (FOR J=1 TO 3,TAB(8);C2#[J];TAB(23);J,LIN(1));
3082 PRINT TAB(8);C2#[4];TAB(23);4
3084 PRINT '27"&a22r45C";"CLAVE?"'27"&dC      "'27"&d@"
3086 PRINT '27"&a22r51C";
3088 ENTER 255,A,Z#
3090 IF Z#[1;1]>"5" OR Z#[1;1]<"1" THEN 3084
3092 IF Z#[2;1]>"2" OR Z#[2;1]<"1" THEN 3084
3094 IF Z#[3;1]>"4" OR Z#[3;1]<"1" THEN 3084
3096 IF Z#[2;1]="1" AND Z#[3;1]="4" THEN 3084
3098 IF Z#[4;1]>"3" OR Z#[4;1]<"1" THEN 3084
3100 CONVERT Z#[4;1] TO D
3102 REM.....
3104 REM  APERTURA DE FICHERO, FORMATO DE PAGINA 2 Y LISTADO
3106 REM  DE ASIGNATURAS DEL CURSO
3108 REM.....
3110 ASSIGN "C"+Z#[1,3]+W#[1,X
3112 READ #1,1;N#,N1,N4,N2,D7#[1],D7#[3],D7#[2]
3114 PRINT '27"h"'27"J"
3116 PRINT '27"&a0r0C";&
      "Carrera:          Plan:          Curso:          Convocat&
      atoria:"
3118 PRINT D7#[1];TAB(20);D7#[2];TAB(39);D7#[3];
3120 PRINT '27"&a2r59C";C4#[D,1;10]

```

```

3122 PRINT LIN(1);"ASIGNATURAS:"
3124 PRINT (FOR K1=1 TO 72,"-";)
3126 I=1,F3#[1]=" "
3128 FOR K=2 TO N2+1
3130   READ #1,K;D6#[1],D6#[2],A1,B,(FOR K1=1 TO B,F2#[K1])
3132   FOR K1=B TO 1 STEP -1
3134     FOR K2=1 TO I
3136       IF F2#[K1]=F3#[K2] THEN DO
3138         F[K2]=F[K2]+1,J1[K2,F[K2]]=A1,J2[K2,F[K2]]=K1
3140         GOTO 3154
3142       DOEND
3144     NEXT K2
3146     I=I+1,F3#[I]=F2#[K1],F[I]=1,J1[I,1]=A1,J2[I,1]=K1
3148     CONVERT I-1 TO H#
3150     IF LEN(H#)=1 THEN H#=" "+H#
3152     PRINT H#;" .";F3#[I]
3154   NEXT K1
3156 NEXT K
3158 PRINT (FOR K3=1 TO 72,"-";)
3160 PRINT LIN(1);'27&
      "&dC
      "'27"&da"
3162 REM.....
3164 REM          SELECCION DE ASIGNATURAS POR EL USUARIO
3166 REM.....
3168 IF N1>N2+1 THEN DO
3170   PRINT '27"A''27"A";&
      "Introduzca numero(s) correspondiente(s) a asignatura(s) seleccion&
ada(s)";'27"A"
3172   FOR W0=1 TO 9999
3174   NEXT W0
3176   PRINT '27"A''27"A"
3178   PRINT SPA(71);". .";'27"G';'27"A";
3180   INPUT "Numero(s) correspondiente(s) ?";
3182   FOR K1=1 TO I

```

```

3184     IF BUF(0)=4 THEN 3212
3186     IF BUF(0)<>5 THEN DO
3188         PRINT '27"M", '27"A", '27"A"
3190         GOTO 3178
3192     DOEND
3194     INPUT :C[K1]:
3196     IF C[K1]<1 OR C[K1]>I-1 THEN DO
3198         D$="No encuentro asignatura que corresponda al numero "
3200         PRINT '27"A" '27"A" ; '7;D$;C[K1];SPA(17);"."
3202         ENTER 3,A,H$
3204         PRINT '27"A", '27"A"
3206         GOTO 3170
3208     DOEND
3210     NEXT K1
3212     REM.....
3214     REM   FORMATO PAGINA 3. LISTADO DE ALUMNOS CON INTRODUCCION
3216     REM   DE NOTAS PARA CADA ASIGNATURA.
3218     REM.....
3220     PRINT '27"h" '27"J" ; '27"&a0r0C" ; '27&
      "&dC
      " '27"&d@"
3222     PRINT '27"&a2r0C" ; &
      "Carrera:           Plan:           Curso:           Convoc&
atoria:"
3224     PRINT D7#[ 1 ];TAB(20);D7#[ 2 ];TAB(39);D7#[ 3 ];
3226     PRINT '27"&a4r59C" ; C4#[ D, 1 ; 10 ]
3228     PRINT '27"&a5r0C" ; '27&
      "&dC
      " '27"C"
3230     PRINT LIN(1);SPA(2);"ALUMNO";TAB(64);"NOTA"
3232     PRINT (FOR K3=1 TO 72, "-" ; )
3234     L1=K1-1
3236     FOR K=1 TO L1
3238         K5=0
3240         PRINT '27"&a5r0C"

```

```

3242 PRINT (FOR K4=1 TO 71," ");". ."
3244 PRINT '27"&a6r2C";"ASIGNATURA:      ";F3#[C[K]+1];LIN(3)
3246 PRINT '27"&a10r0C"'27"J"
3248 PRINT '27"&a10r0C"'27"J"
3250 FOR K1=N2+2 TO N1
3252     READ #1,K1;Q,(FOR K3=1 TO 22,A#[K3]),A,T1#[1],T2#[1],T3#[1]
3254     FOR K2=1 TO F[C[K]+1]
3256         F9#=T2#[1,J2[C[K]+1,K2]*2+1]
3258         IF Q=J1[C[K]+1,K2] AND ((F9#="Z" AND (D=2 OR D=3)) OR (
            F9#="Y" AND D=2) OR (F9#="X" AND D=3) OR (F9#="W" AND D=1) OR (F9#&
            ="V" AND (D=1 OR D=3)) OR (F9#="U" AND (D=1 OR D=2)) OR F9#="S")&
            THEN DO
3260             K5=K5+1
3262             IF K5=1 THEN PRINT '27"&a0r0C";&
                "Introduzca nota (0 a 10, 'MAT' o 'NP')"
                .";LIN(8)
3264             PRINT SPA(2);A#[1];TAB(22);A#[2];TAB(42);A#[3];
3266             IF K5>6 THEN PRINT '27"&a10r0C"'27"M"'27"M"
3268             IF K5=1 THEN DO
3270                 PRINT '27"&a10r65C";'27"&dC      "'27"&d0"
3272                 PRINT '27"&a10r65C";
3274                 ENTER 255,A,V#
3276             DOEND
3278             IF K5=2 THEN DO
3280                 PRINT '27"&a12r65C";'27"&dC      "'27"&d0"
3282                 PRINT '27"&a12r65C";
3284                 ENTER 255,A,V#
3286             DOEND
3288             IF K5=3 THEN DO
3290                 PRINT '27"&a14r65C";'27"&dC      "'27"&d0"
3292                 PRINT '27"&a14r65C";
3294                 ENTER 255,A,V#
3296             DOEND
3298             IF K5=4 THEN DO
3300                 PRINT '27"&a16r65C";'27"&dC      "'27"&d0"

```

```

3302          PRINT '27"&a16r65C";
3304          ENTER 255,A,V$
3306        DOEND
3308        IF K5=5 THEN DO
3310          PRINT '27"&a18r65C";'27"&dC    "'27"&d@"
3312          PRINT '27"&a18r65C";
3314          ENTER 255,A,V$
3316        DOEND
3318        IF K5=6 THEN DO
3320          PRINT '27"&a20r65C";'27"&dC    "'27"&d@"
3322          PRINT '27"&a20r65C";
3324          ENTER 255,A,V$
3326        DOEND
3328        PRINT LIN(1)
3330        IF K5>6 THEN DO
3332          PRINT '27"&a20r65C";'27"&dC    "'27"&d@"
3334          PRINT '27"&a20r65C";
3336          ENTER 255,A,V$
3338          PRINT LIN(1)
3340        DOEND
3342        REM.....
3344        REM  VERIFICACION DEL DATO IUNTRODUCIDO COMO NOTA
3346        REM.....
3348        V$=DEB$(V$)
3350        IF V$="MAT" THEN T3$(1,(J2[C[K]+1,K2]-1)*3+D;1)="A"
3352        IF V$="10" THEN T3$(1,(J2[C[K]+1,K2]-1)*3+D;1)="B"
3354        IF V$="9" THEN T3$(1,(J2[C[K]+1,K2]-1)*3+D;1)="C"
3356        IF V$="8" THEN T3$(1,(J2[C[K]+1,K2]-1)*3+D;1)="D"
3358        IF V$="7" THEN T3$(1,(J2[C[K]+1,K2]-1)*3+D;1)="E"
3360        IF V$="6" THEN T3$(1,(J2[C[K]+1,K2]-1)*3+D;1)="F"
3362        IF V$="5" THEN T3$(1,(J2[C[K]+1,K2]-1)*3+D;1)="G"
3364        IF V$="4" THEN T3$(1,(J2[C[K]+1,K2]-1)*3+D;1)="H"
3366        IF V$="3" THEN T3$(1,(J2[C[K]+1,K2]-1)*3+D;1)="I"
3368        IF V$="2" THEN T3$(1,(J2[C[K]+1,K2]-1)*3+D;1)="J"
3370        IF V$="1" THEN T3$(1,(J2[C[K]+1,K2]-1)*3+D;1)="K"

```

```

3372          IF V$="0" THEN T3$(1,(J2[ C[K]+1,K2]-1)*3+D;1)="L"
3374          IF V$="NP" THEN T3$(1,(J2[ C[K]+1,K2]-1)*3+D;1)="M"
3376          IF V$<>"MAT" AND V$<>"10" AND V$<>"9" AND V$<>"8" AND &
V$<>"7" AND V$<>"6" AND V$<>"5" AND V$<>"4" AND V$<>"3" AND V$<>&
"2" AND V$<>"1" AND V$<>"0" AND V$<>"NP" THEN DO
3378          PRINT '27"&a0r0C"'7;:&
"ERROR ! Nota incorrecta                                     &
.
3380          ENTER 3,X,H$
3382          PRINT '27"&a0r0C";:&
"Introduzca nota (0 a 10; 'MAT' o 'NP')                       &
.
3384          IF K5<=6 THEN GOTO K5 OF 3268,3278,3288,3298,3308,&
3318
3386          ELSE GOTO 3330
3388          DOEND
3390          IF V$="MAT" OR V$="10" OR V$="9" OR V$="8" OR V$="7"&
OR V$="6" OR V$="5" THEN DO
3392          IF D=1 AND (F9$="V" OR F9$="U") THEN DO
3394          T2$(1,J2[ C[K]+1,K2]*2;1)="W"
3396          CONVERT T2$(1,J2[ C[K]+1,K2]*2-1;1) TO T2
3398          T2=T2-1
3400          CONVERT T2 TO T2$(1,J2[ C[K]+1,K2]*2-1;1)
3402          DOEND
3404          IF D=2 AND F9$="Z" THEN DO
3406          T2$(1,J2[ C[K]+1,K2]*2;1)="Y"
3408          CONVERT T2$(1,J2[ C[K]+1,K2]*2-1;1) TO T2
3410          T2=T2-1
3412          CONVERT T2 TO T2$(1,J2[ C[K]+1,K2]*2-1;1)
3414          DOEND
3416          DOEND
3418          REM.....
3420          REM REINTRODUCCION DE LOS DATOS DEL ALUMNO EN .
3422          REM FICHERO TIPO C Y EXPLORACION DE POSIBLE NOTA DE .
3424          REM PROYECTO PARA INTRODUCIR DATOS EN FICH. EXPED. .

```

```

3426          REM.....
3428          PRINT #1,K1;Q,(FOR K3=1 TO 22,A#[K3]),1,T1#[1],T2#[1],&
          T3#[1]
3430          IF F9#="S" AND (V#="MAT" OR V#="10" OR V#="9" OR V#="8"&
          OR V#="7" OR V#="6" OR V#="5") THEN DO
3432              L=2,H=N5
3434              M=INT((L+H)/2)
3436              READ #6,M;Q1,(FOR K3=1 TO 3,D5#[K3])
3438              IF D5#[1]+D5#[2]+D5#[3]>=A#[1]+A#[2]+A#[3] THEN 3446
3440              IF L>=H THEN 3482
3442              L=M+1
3444              GOTO 3434
3446              IF D5#[1]+D5#[2]+D5#[3]=A#[1]+A#[2]+A#[3] THEN 3454
3448              IF L>=H THEN 3482
3450              H=M-1
3452              GOTO 3434
3454          READ #6,M;Q1,(FOR K3=1 TO 22,A#[K3]),T,(FOR K3=1 TO T&
          ,T1#[K3],T2#[K3],T3#[K3])
3456          FOR K3=1 TO T
3458              IF T1#[K3,2;3]=Z#[1;3] THEN DO
3460                  Q1=LEN(T2#[K3])
3462                  IF V#="MAT" THEN T2#[K3,Q1;1]="A"
3464                  IF V#="10" THEN T2#[K3,Q1;1]="B"
3466                  IF V#="9" THEN T2#[K3,Q1;1]="C"
3468                  IF V#="8" THEN T2#[K3,Q1;1]="D"
3470                  IF V#="7" THEN T2#[K3,Q1;1]="E"
3472                  IF V#="6" THEN T2#[K3,Q1;1]="F"
3474                  IF V#="5" THEN T2#[K3,Q1;1]="G"
3476              DOEND
3478          NEXT K3
3480          PRINT #6,M;-1,(FOR K3=1 TO 22,A#[K3]),T,(FOR K3=1 TO &
          T,T1#[K3],T2#[K3],T3#[K3])
3482          DOEND
3484          DOEND
3486          NEXT K2

```

```

3488     NEXT K1
3490     IF K5=0 THEN DO
3492     PRINT '27"&a1r0C":&
        "No hay ningun alumno matriculado de esta asignatura en ";C4#[D,1];&
10]
3494     ENTER 5,X,V#
3496     DOEND
3498     NEXT K
3500     IF K5<>0 THEN PRINT '27"&a2r0C":
3502 DOEND
3504 ELSE DO
3506     PRINT '27"A"'27"A":&
        "No hay ningun alumno matriculado en este curso           &
        ."
3508     ENTER 5,X,V#
3510 DOEND
3512 REM.....
3514 REM   DEVOLUCION DE CONTROL Y CIERRE DE FICHEROS #6 Y #9      .
3516 REM.....
3518 PRINT '27"A"'27"A":&
        " Para devolver control pulse 'RETURN'. Para repetir pulse 'R'   &
        ."
3520 PRINT '27"A":
3522 LINPUT #9;L#
3524 IF L#<>" " AND L#<>"R" THEN 3520
3526 ASSIGN *,1
3528 IF L#="R" THEN 3060
3530 ASSIGN *,6
3532 ASSIGN *,9
3534 CHAIN "ARCHALUM"+W#

```

>

DELTA5

```

4000 REM*****
4002 REM
4004 REM          MODULO 6.      VOLCADO DE FICHEROS TIPO "C"
4006 REM
4008 REM*****
4010 COM W[*]
4012 DIM C4#[ 3,20 ],D5#[ 3,30 ],D6#[ 2,30 ],D7#[ 3,30 ],E8#[ 4 ],E9#[ 1 ],E0#[ 1 ]
4014 DIM F#[ 22,30 ],F2#[ 15,65 ],F3#[ 45,65 ],F4#[ 10,31 ],F9#[ 10,45 ],G#[ 8 ]
4016 DIM J#[ 3 ],N#[ 5 ],T1#[ 10,6 ],T2#[ 10,31 ],T3#[ 10,45 ],V[ 45,3 ]
4018 C4#[ 1 ]="FEBRERO          1",C4#[ 2 ]="JUNIO          2"
4020 C4#[ 3 ]="SEPTIEMBRE      3"
4022 FILES *,*,*,*,*,*,*,*,*,*
4024 ASSIGN "HELP"+W#,5,X1
4026 READ #5,1;N3,G#
4028 ASSIGN G#+W#,6,X1
4030 READ #6,1;N5
4032 SYSTEM X,"FILE @@@@=#STDINX;REC=-1"
4034 ASSIGN "QQQQ",9,X
4036 J=0
4038 REM.....
4040 REM          FORMATO PAGINA 1A. EXPLORACION DE TODOS LOS FICHEROS
4042 REM          TIPO 'C' PARA COMPROBAR SI QUEDA ALGUNA ASIGNATURA
4044 REM          POR CALIFICAR
4046 REM.....
4048 PRINT '27"h"'27"J"
4050 FOR J1=1 TO 5
4052     FOR J2=1 TO 2
4054         IF J2=1 THEN J4=3
4056         ELSE J4=4
4058         FOR J3=1 TO J4
4060             CONVERT J1 TO J#[ 1;1 ]
4062             CONVERT J2 TO J#[ 2;1 ]
4064             CONVERT J3 TO J#[ 3;1 ]
4066             ASSIGN "C"+J#[ 1;3 ]+W#,1,X1

```

```

4068      READ #1,1;N#,N1,N4,N2,D7#[ 1 ],D7#[ 3 ],D7#[ 2 ]
4070      K=0,I=1,F3#[ 1 ]=" ",V[ 1, 1 ]=V[ 1, 2 ]=V[ 1, 3 ]=V[ 2, 1 ]=V[ 2, 2 ]=V[ 2, 3 ]&
      =0
4072      FOR K9=1 TO 3
4074          FOR K1=N2+2 TO N1
4076              READ #1,K1;Q,(FOR K3=1 TO 22,F#[K3]),T,F3#[ 1 ],F4#[ 1 ],&
      F9#[ 1 ]
4078              FOR K2=1 TO LEN(F4#[ 1 ])/2
4080                  IF F4#[ 1,2*K2;1 ]<"U" THEN 4190
4082                  IF F4#[ 1,2*K2;1 ]="Z" THEN DO
4084                      IF F9#[ 1,3*K2-1;1 ]="0" AND K9=2 THEN DO
4086                          A=2
4088                          GOSUB 9900
4090                      DOEND
4092                      IF F9#[ 1,3*K2;1 ]="0" AND K9=3 THEN DO
4094                          A=3
4096                          GOSUB 9900
4098                      DOEND
4100                      GOTO 4190
4102                  DOEND
4104                  IF F4#[ 1,2*K2;1 ]="Y" THEN DO
4106                      IF F9#[ 1,3*K2-1;1 ]="0" AND K9=2 THEN DO
4108                          A=2
4110                          GOSUB 9900
4112                      DOEND
4114                      GOTO 4190
4116                  DOEND
4118                  IF F4#[ 1,2*K2;1 ]="X" THEN DO
4120                      IF F9#[ 1,3*K2;1 ]="0" AND K9=3 THEN DO
4122                          A=3
4124                          GOSUB 9900
4126                      DOEND
4128                      GOTO 4190
4130                  DOEND
4132                  IF F4#[ 1,2*K2;1 ]="W" THEN DO

```

```

4134             IF F9#[1,3*K2-2;1]="0" AND K9=1 THEN DO
4136                 A=1
4138                 GOSUB 9900
4140             DOEND
4142             GOTO 4190
4144         DOEND
4146     IF F4#[1,2*K2;1]="V" THEN DO
4148         IF F9#[1,3*K2-2;1]="0" AND K9=1 THEN DO
4150             A=1
4152             GOSUB 9900
4154         DOEND
4156         IF F9#[1,3*K2;1]="0" AND K9=3 THEN DO
4158             A=3
4160             GOSUB 9900
4162         DOEND
4164         GOTO 4190
4166     DOEND
4168     IF F4#[1,2*K2;1]="U" THEN DO
4170         IF F9#[1,3*K2-2;1]="0" AND K9=1 THEN DO
4172             A=1
4174             GOSUB 9900
4176         DOEND
4178         IF F9#[1,3*K2-1;1]="0" AND K9=2 THEN DO
4180             A=2
4182             GOSUB 9900
4184         DOEND
4186         GOTO 4190
4188     DOEND
4190     NEXT K2
4192     NEXT K1
4194     NEXT K9
4196     ASSIGN *,1
4198     IF K=1 THEN PRINT (FOR K3=1 TO 72,"-";)
4200     NEXT J3
4202     NEXT J2

```

```

4204 NEXT J1
4206 IF J=1 THEN DO
4208   PRINT LIN(1);TAB(1);'27&
      "&C NINGUN VOLCADO EFECTUADO !!. Para devolver control pulse 'RET&
URN'.   "'27"&d@"
4210   GOTO 4382
4212 DOEND
4214 REM.....
4216 REM   FORMATO PAGINA 1B. EJECUCION DE VOLCADO
4218 REM.....
4220 FOR J1=1 TO 5
4222   FOR J2=1 TO 2
4224     IF J2=1 THEN J4=3
4226     ELSE J4=4
4228     FOR J3=1 TO J4
4230       CONVERT J1 TO J#[1;1]
4232       CONVERT J2 TO J#[2;1]
4234       CONVERT J3 TO J#[3;1]
4236       ASSIGN "C"+J#[1;3]+W#,1,X1
4238       READ #1,1;N#,N1,N4,N2,D7#[1],D7#[3],D7#[2]
4240       FOR K1=N2+2 TO N1
4242         READ #1,K1;Q,(FOR K3=1 TO 22,F#[K3]),T,F3#[1],F4#[1],F9#[&
1 ]
4244         REM.....
4246         REM   LOCALIZACION DE ALUMNO EN ARCHIVO DE EXPEDIENTES
4248         REM.....
4250         L=2,H=N5
4252         M=INT((L+H)/2)
4254         READ #6,M;Q,(FOR K3=1 TO 3,D5#[K3])
4256         IF D5#[1]+D5#[2]+D5#[3]>=F#[1]+F#[2]+F#[3] THEN 4264
4258         IF L>=H THEN 4346
4260         L=M+1
4262         GOTO 4252
4264         IF D5#[1]+D5#[2]+D5#[3]=F#[1]+F#[2]+F#[3] THEN 4280
4266         IF L>=H THEN 4346

```

```

4268          H=M-1
4270          GOTO 4252
4272          REM.....
4274          REM   INTRODUCCION DE NOTAS Y CASOS DE NO PRESENTADO .
4276          REM   EN ARCHIVO DE EXPEDIENTES. .
4278          REM.....
4280          READ #6,M;Q,(FOR K3=1 TO 22,F#[K3]),T,(FOR K4=1 TO T,T1#[&
          K4 ],T2#[K4 ],T3#[K4 ])
4282          FOR K5=1 TO T
4284              IF F3#[1]=T1#[K5 ] THEN DO
4286                  T2#[K5 ]=F4#[1 ]
4288                  FOR K6=1 TO LEN(F4#[1 ])/2
4290                      IF F4#[1,2*K6;1]<>"T" AND F4#[1,2*K6;1]>"R" THEN &
          DO
4292                          FOR K8=1 TO 3
4294                              IF F9#[1,3*K6+K8-3;1]="0" THEN F9#[1,3*K6+K8-3;&
          1 ]="N"
4296                          NEXT K8
4298                          IF F9#[1,3*K6-2;1]<=F9#[1,3*K6-1;1 ] THEN DO
4300                              IF F9#[1,3*K6-2;1]<=F9#[1,3*K6;1 ] THEN E9#=F9#[&
          1,3*K6-2;1 ]
4302                                  ELSE E9#=F9#[1,3*K6;1 ]
4304                                  DOEND
4306                                  ELSE DO
4308                                      IF F9#[1,3*K6-1;1]<=F9#[1,3*K6;1 ] THEN E9#=F9#[&
          1,3*K6-1;1 ]
4310                                      ELSE E9#=F9#[1,3*K6;1 ]
4312                                  DOEND
4314                                  IF E9#<"N" THEN T2#[K5,2*K6;1]=E9#
4316                                  IF Q>0 THEN DO
4318                                      FOR K7=1 TO 3
4320                                          IF F9#[1,3*K6+K7-3;1]="M" THEN DO
4322                                              CONVERT K6 TO E9#
4324                                              CONVERT K7 TO E0#
4326                                              E8#=DAT$(14,17)

```

```

4328             T3#[ K5 ]=T3#[ K5 ]+E9#+E0#+E8#[ 3;2 ]
4330             DOEND
4332             NEXT K7
4334             DOEND
4336             DOEND
4338             NEXT K6
4340             DOEND
4342             NEXT K5
4344             PRINT #6,M;@,(FOR K3=1 TO 22,F#[K3]),T,(FOR K4=1 TO T,T1#&
             [ K4 ],T2#[ K4 ],T3#[ K4 ])
4346             NEXT K1
4348             REM.....
4350             REM  PURGA DE ALUMNOS DEL FICHERO VOLCADO .
4352             REM.....
4354             IF N1>N2+1 THEN DO
4356                 FOR K3=N2+2 TO N4
4358                     PRINT #1,K3
4360                 NEXT K3
4362                 PRINT #1,1;N#,N2+1,N4,N2,D7#[ 1 ],D7#[ 3 ],D7#[ 2 ]
4364                 PRINT "FICHERO ":"C";J#[ 1;3 ];" VOLCADO"
4366             DOEND
4368             ASSIGN *,1
4370             NEXT J3
4372             NEXT J2
4374             NEXT J1
4376             PRINT LIN(1);TAB(1);'27&
             "&dC FIN DE VOLCADO DE FICHEROS. Para devolver control pulse 'RETU&
RN'.             "'27"&d@"
4378             ASSIGN *,5
4380             ASSIGN *,6
4382             PRINT '27"A";
4384             LINPUT #9;L$
4386             IF L#<>" THEN 4382
4388             PRINT '27"m"
4390             ASSIGN *,9
4392             CHAIN "ARCHALUM"+W#

```

DELTA6

```

4500 REM*****
4502 REM
4504 REM          MODULO 7.      LISTADO ALFABETICO POR CARRERA      *
4506 REM
4508 REM*****
4510 COM W#[*]
4512 DIM C1#[5,16],D5#[3,30],D7#[3,30],M#[23,30],N#[5],L#[1],L2#[2]
4514 DIM S[7],S1[7]
4516 C1#[1]="INDUSTRIAL",C1#[2]="TELECOMUNICACION"
4518 C1#[3]="TOPOGRAFIA",C1#[4]="OBRAS PUBLICAS",C1#[5]="NAVAL"
4520 SYSTEM X,"FILE @@@@=$STDIN;REC=-1"
4522 ASSIGN "@@@@",9,X
4524 FILES *,*,*,*,*,*,*,*,*
4526 REM.....
4528 REM  FORMATO PAGINA 1. SELECCION DE CARRERA. APERTURA DE
4530 REM  FICHEROS. INTRODUCCION DE LIMITES.
4532 REM.....
4534 PRINT '27"h''27"J"
4536 PRINT '27"&a1r5C''27&
      "&dC Para seleccionar carrera introduzca numero correspondiente "&
      '27"&d@  "'27"&dCNo.? "'27"&d@"
4538 PRINT LIN(2)
4540 FOR K=1 TO 5
4542   CONVERT K TO H#
4544   PRINT TAB(29);H#;".  ";C1#[K]
4546 NEXT K
4548 PRINT '27"&a1r72C''27"[ "; " ";'27"]"
4550 PRINT '27"W";
4552 LINPUT #9;L#
4554 PRINT '27"X"
4556 IF L#<>"1" AND L#<>"2" AND L#<>"3" AND L#<>"4" AND L#<>"5" THEN &
      4548
4558 ASSIGN "C"+L#+ "11"+W#,1,X1
4560 ASSIGN "C"+L#+ "12"+W#,2,X1

```

```

4562 ASSIGN "C"+L$+"13"+W$,3,X1
4564 ASSIGN "C"+L$+"21"+W$,4,X1
4566 ASSIGN "C"+L$+"22"+W$,5,X1
4568 ASSIGN "C"+L$+"23"+W$,6,X1
4570 ASSIGN "C"+L$+"24"+W$,7,X1
4572 PRINT '27"&a17r5C"'27&
      "&dC Introduzca limites de listado alfabetico si los desea      &
      "'27"&d@"
4574 PRINT '27"&a20r13C";"Limite inferior? "'27"&dC "'27"&d@"'27&
      "&a20r45C";"Limite superior? "'27"&dC "'27"&d@"
4576 PRINT '27"&a20r30C";
4578 LINPUT #9;L1$
4580 L1$=UPS$(L1$)
4582 IF L1$<>" " AND (L1$<"A" OR L1$>"Z") THEN 4576
4584 PRINT '27"&a20r62C";
4586 LINPUT #9;L2$
4588 L2$=UPS$(L2$)
4590 IF L2$<>" " AND (L2$<"A" OR L2$>"Z") THEN 4584
4592 IF L2$="" THEN L2$="ZZ"
4594 CONVERT L$ TO L
4596 REM.....
4598 REM  FORMATO PAGINA 2. LOC ALIZACION Y LISTADO ALFABETICO DE
4600 REM  ALUMNOS. CIERRE DE FICHEROS Y DEVOLUCION DE CONTROL
4602 REM.....
4604 PRINT '27"h"'27"J"
4606 PRINT '27"&a1r9C";"ALUMNOS";SPA(30);"CARRERA: ";C1$[L]
4608 PRINT '27"&a2r6C";(FOR K=1 TO 66;"=";)
4610 FOR K=1 TO 7
4612   READ #K,1;N$,N1,N4,N2
4614   S[K]=N1+1
4616   S1[K]=N2+2
4618 NEXT K
4620 PRINT '27"1"
4622 FOR K=1 TO 7
4624   IF S[K]>S1[K] THEN DO

```

```

4626     READ #K,S1[K];Q,M#[3*K-2],M#[3*K-1],M#[3*K]
4628     IF L1#<=M#[3*K-2] THEN DO
4630         GOTO 4644
4632     DOEND
4634     ELSE DO
4636         S1[K]=S1[K]+1
4638         GOTO 4624
4640     DOEND
4642 DOEND
4644 NEXT K
4646 D7#[1]=D7#[2]=D7#[3]=L2#
4648 D5#[1]=D5#[2]=D5#[3]=L2#
4650 FOR K=1 TO 7
4652     IF S[K]>S1[K] THEN DO
4654         IF M#[3*K-2]+M#[3*K-1]+M#[3*K]<D5#[1]+D5#[2]+D5#[3] THEN DO
4656             D5#[1]=M#[3*K-2],D5#[2]=M#[3*K-1],D5#[3]=M#[3*K]
4658             J=K
4660         DOEND
4662     DOEND
4664 NEXT K
4666 IF L2#>D5#[1] THEN DO
4668     IF D5#[1]<>D7#[1] OR D5#[2]<>D7#[2] OR D5#[3]<>D7#[3] THEN DO
4670         PRINT TAB(9);D5#[1]+ "    "+D5#[2]+ "    "+D5#[3]
4672     DOEND
4674     D7#[1]=D5#[1],D7#[2]=D5#[2],D7#[3]=D5#[3]
4676     S1[J]=S1[J]+1
4678     IF S[J]>S1[J] THEN DO
4680         READ #J,S1[J];Q,M#[3*J-2],M#[3*J-1],M#[3*J]
4682     DOEND
4684     GOTO 4648
4686 DOEND
4688 IF D7#[1]=L2# AND D7#[2]=L2# AND D7#[3]=L2# THEN DO
4690     PRINT TAB(9);"NINGUN ALUMNO LOCALIZADO EN ESTE INTERVALO"
4692 DOEND
4694 FOR K=1 TO 7

```

```
4696  ASSIGN *,K
4698  NEXT K
4700  PRINT LIN(4), '27"&a23r6C"'27&
      "&dC Para devolver control pulse 'RETURN'. Para repetir pulse'R' &
      "'27"&d0"
4702  PRINT '27"A"!'27"&a23r70C";
4704  INPUT #9;L$
4706  IF L$<>" " AND L$<>"R" THEN 4702
4708  PRINT '27"m"
4710  IF L$="R" THEN 4534
4712  ASSIGN *,9
4714  CHAIN "ARCHALUM"+W$
>
```

DELTA7

```

5000 REM*****
5002 REM
5004 REM      MODULO B.      OBTENCION DE LISTADO ALFABETICO
5006 REM      POR PLAN DETALLANDO CURSO
5008 REM
5010 REM*****
5012 COM W[*]
5014 DIM B#[ 4, 12 ], C1#[ 5, 15 ], C2#[ 4, 10 ], C3#[ 2, 10 ], D5#[ 3, 30 ], D7#[ 3, 30 ]
5016 DIM N#[ 5 ], Z#[ 4 ]
5018 B#[ 1 ]=" CARRERA ", B#[ 3 ]=" CURSO ", B#[ 2 ]=" PLAN "
5020 C1#[ 1 ]=" INDUSTRIAL ", C1#[ 2 ]=" TELECOMUNICACION ", C1#[ 5 ]=" NAVAL "
5022 C1#[ 4 ]=" OBRAS PUBLICAS ", C1#[ 3 ]=" TOPOGRAFIA "
5024 C2#[ 1 ]=" PRIMERO ", C2#[ 2 ]=" SEGUNDO ", C2#[ 3 ]=" TERCERO ", C2#[ 4 ]=&
" CUARTO "
5026 C3#[ 1 ]=" TRIENAL ", C3#[ 2 ]=" CUATRIENAL "
5028 FILES *,*,*,*,*,*,*,*,*
5030 SYSTEM X,"file qqqq=$stdin;rec=-1"
5032 ASSIGN "QQQQ",9,X
5034 REM.....
5036 REM  FORMATO PAGINA 1. INTRODUCCION DE CLAVE
5038 REM.....
5040 PRINT '27"h'"27"J"
5042 PRINT '27"&a1r6C'"27&
" &dC Introduzca clave para seleccionar carrera, plan y curso      &
" '27"&d@"
5044 PRINT LIN(2);TAB(11);" Carrera:";TAB(33);" Plan:";TAB(54);" Curso:"
5046 PRINT TAB(11);"-----";TAB(33);"-----";TAB(54)&
;"-----"
5048 PRINT TAB(11);"1. ";C1#[ 1 ];TAB(33);"1. ";C3#[ 1 ];TAB(54);&
"0. TODOS"
5050 PRINT TAB(11);"2. ";C1#[ 2 ];TAB(33);"2. ";C3#[ 2 ];TAB(54);"1. ";C2#[ &
[ 1 ]
5052 PRINT TAB(11);"3. ";C1#[ 3 ];TAB(54);"2. ";C2#[ 2 ]
5054 PRINT TAB(11);"4. ";C1#[ 4 ];TAB(54);"3. ";C2#[ 3 ]

```

```

5056 PRINT TAB(11);"5. ";C1#[5];TAB(54);"4. ";C2#[4]
5058 PRINT '27"&a14r6C";"CLAVE?";
5060 PRINT '27"&a14r12C"'27"&dC "'27"&d@"'27"&a14r12C";
5062 ENTER 255,A,Z#
5064 IF Z#[1;1]>"5" OR Z#[1;1]<"1" THEN 5060
5066 IF Z#[2;1]>"2" OR Z#[2;1]<"1" THEN 5060
5068 IF Z#[3;1]>"4" OR Z#[3;1]<"0" THEN 5060
5070 IF Z#[2;1]="1" AND Z#[3;1]="4" THEN 5060
5072 IF Z#[3;1]="0" THEN DO
5074 PRINT '27"&a22r6C"'27&
      "&dC Para visualizar pagina siguiente pulse 'RETURN'.      &
      "'27"&d@"
5076 LINPUT #9;L#
5078 DOEND
5080 REM.....
5082 REM FORMATO PAGINA 2. APERTURA FICHERO 'C'. LISTADO DE
5084 REM ALUMNOS. CIERRE DE FICHERO 'C'. DEVOLUCION DE CONTROL
5086 REM Y CIERRE DE FICHERO #9
5088 REM.....
5090 PRINT '27"h"'27"J"
5092 CONVERT Z#[3;1] TO D
5094 IF D=0 AND Z#[2;1]="2" THEN L=1,H=4
5096 IF D=0 AND Z#[2;1]="1" THEN L=1,H=3
5098 IF D<>0 THEN L=H=D
5100 FOR K=L TO H
5102 CONVERT K TO Z#[3;1]
5104 ASSIGN "C"+Z#[1;3]+W#,K,X1
5106 READ #K,1;N#,N1,N4,N2,D7#[1],D7#[3],D7#[2]
5108 PRINT LIN(2);'27"U"
5110 PRINT TAB(6);"Carrera: ";D7#[1];TAB(35);"Plan: ";D7#[2];TAB(57)&
      ;"Curso: ";D7#[3]
5112 PRINT TAB(5);(FOR A=1 TO 67,"=";)
5114 PRINT '27"1"
5116 FOR J=N2+2 TO N1
5118 READ #K,J;Q,D5#[1],D5#[2],D5#[3]

```

```

5120     PRINT TAB(6);D5#[1]+ "    "+D5#[2]+ "    "+D5#[3]
5122     NEXT J
5124     ASSIGN *,K
5126     IF K=H THEN DO
5128     PRINT LIN(4),'27"&a23r5C''27&
        "&dC Para devolver control pulse 'RETURN'. Para repetir pulse 'R' &
        "'27"&d0"
5130     PRINT '27"A';'27"&a23r70C';
5132     LINPUT #9;L$
5134     IF L$<>" AND L$<>"R" THEN 5130
5136     PRINT '27"m"
5138     IF L$="R" THEN 5040
5140     DOEND
5142     ELSE DO
5144     LINPUT #9;L$
5146     PRINT '27"m"
5148     DOEND
5150     PRINT LIN(2)
5152     NEXT K
5154     ASSIGN *,9
5156     CHAIN "ARCHALUM"+W$
>

```

DELTAB

```
5500 REM*****
5502 REM *
5504 REM      MODULO 9.  OBTENCION DE LISTADO ALFABETICO DE *
5506 REM      ALUMNOS POR ASIGNATURA Y ESPECIALIDAD *
5508 REM      ESPECIFICANDO CONVOCATORIA PARA UNO O VARIOS CURSOS *
5510 REM      PERTENECIENTES A UN MISMO PLAN Y CARRERA. *
5512 REM *
5514 REM*****
5516 COM W#[*]
5518 DIM B#[ 4, 12 ], C[ 50 ], C1#[ 5, 15 ], C2#[ 4, 10 ], C3#[ 2, 10 ], D#[ 50 ]
5520 DIM D6#[ 2, 30 ], D7#[ 3, 30 ], F[ 50 ], F2#[ 15, 65 ], F3#[ 45, 65 ], H#[ 2 ]
5522 DIM J1[ 50, 10 ], J2[ 50, 10 ], M#[ 23, 30 ], N#[ 5 ], T1#[ 10, 6 ], T2#[ 10, 31 ]
5524 DIM T3#[ 10, 45 ], Z#[ 4 ]
5526 B#[ 1 ]=" CARRERA", B#[ 3 ]=" CURSO", B#[ 2 ]=" PLAN"
5528 C1#[ 1 ]=" INDUSTRIAL", C1#[ 2 ]=" TELECOMUNICACION", C1#[ 5 ]=" NAVAL"
5530 C1#[ 4 ]=" OBRAS PUBLICAS", C1#[ 3 ]=" TOPOGRAFIA"
5532 C2#[ 1 ]=" PRIMERO", C2#[ 2 ]=" SEGUNDO", C2#[ 3 ]=" TERCERO", C2#[ 4 ]=&
" CUARTO"
5534 C3#[ 1 ]=" TRIENAL", C3#[ 2 ]=" CUATRIENAL"
5536 FILES *,*,*,*,*,*,*,*,*
5538 SYSTEM X, "FILE @@@@=$STDIN;REC=-1"
5540 ASSIGN "QQQQ",9,X
5542 REM.....
5544 REM      FORMATO PAGINA 1. INTRODUCCION DE CLAVE.
5546 REM.....
5548 PRINT '27"h" '27"J"
5550 PRINT '27"&a1r6C" '27&
" &dC Introduzca clave para seleccionar carrera, plan y curso &
" '27"&d@"
5552 PRINT LIN(2);TAB(11);" Carrera:";TAB(33);" Plan:";TAB(54);" Curso:"
5554 PRINT TAB(11);"-----";TAB(33);"-----";TAB(54)&
;"-----"
5556 PRINT TAB(11);"1. ";C1#[ 1 ];TAB(33);"1. ";C3#[ 1 ];TAB(54);&
"0. TODOS"
```

```

5558 PRINT TAB(11);"2. ";C1$(2);TAB(33);"2. ";C3$(2);TAB(54);"1. ";C2$&
[1]
5560 PRINT TAB(11);"3. ";C1$(3);TAB(54);"2. ";C2$(2)
5562 PRINT TAB(11);"4. ";C1$(4);TAB(54);"3. ";C2$(3)
5564 PRINT TAB(11);"5. ";C1$(5);TAB(54);"4. ";C2$(4)
5566 PRINT '27"&a14r6C";"CLAVE?";
5568 PRINT '27"&a14r12C''27"&dC "'27"&d@''27"&a14r12C";
5570 ENTER 255,A,Z#
5572 IF Z$(1;1)>"5" OR Z$(1;1)<"1" THEN 5568
5574 IF Z$(2;1)>"2" OR Z$(2;1)<"1" THEN 5568
5576 IF Z$(3;1)>"4" OR Z$(3;1)<"0" THEN 5568
5578 IF Z$(2;1)="1" AND Z$(3;1)="4" THEN 5568
5580 IF Z$(3;1)="0" THEN DO
5582 PRINT '27"&a22r6C''27&
&dC Para visualizar pagina siguiente pulse 'RETURN'.
''27"&d@"
5584 LINPUT #9;L$
5586 REM.....
5588 REM FORMATO PAGINA 2. APERTURA DE FICHERO 'C'. LISTADO DE
5590 REM ASIGNATURAS.
5592 REM.....
5594 PRINT '27"h''27"J"
5596 DOEND
5598 CONVERT Z$(3;1) TO D
5600 IF D=0 AND Z$(2;1)="2" THEN L=1,H=4
5602 IF D=0 AND Z$(2;1)="1" THEN L=1,H=3
5604 IF D<>0 THEN L=H=D
5606 FOR K=L TO H
5608 CONVERT K TO Z$(3;1)
5610 ASSIGN "C"+Z$(1;3)+W$,K,X1
5612 READ #K;1;N$,N1,N4,N2,D7$(1),D7$(3),D7$(2)
5614 PRINT LIN(2);'27"U"
5616 PRINT TAB(6);"Carrera: ";D7$(1);TAB(35);"Plan: ";D7$(2);TAB(57)&
;"Curso: ";D7$(3)
5618 PRINT '27"1"

```



```

5684 DOEND
5686 FOR K1=1 TO I
5688     IF BUF(0)=4 THEN 5716
5690     IF BUF(0)<>5 THEN DO
5692         PRINT '27"M", '27"A", '27"A"
5694         GOTO 5672
5696     DOEND
5698     INPUT :C[K1]:
5700     IF C[K1]<1 OR C[K1]>I-1 THEN DO
5702         D$=&
          *
          No encuentro asignatura que corresponda al numero "
5704         PRINT '27"A"; '7;D$;C[K1];SPA(14);"."
5706         ENTER 3,A,H$
5708         PRINT '27"A", '27"A"
5710         GOTO 5658
5712     DOEND
5714 NEXT K1
5716 L1=K1-1
5718 REM.....
5720 REM   FORMATO PAGINA 3. LISTADO ALFABETICO DE ALUMNOS
5722 REM   (INDICANDO CONVOCATORIAS) DE LAS ASIGNATURAS
5724 REM   SELECCIONADAS. CIERRE FICHERO 'C'.
5726 REM.....
5728 PRINT '27"h" '27"J"
5730 FOR K1=1 TO L1
5732     FOR K2=1 TO F[C[K1]+1]
5734         READ #K, J1[C[K1]+1, K2]+1; D6#[1], D6#[2], A1, B, (FOR K3=1 TO B, &
          F2#[K3])
5736         IF D6#[1]<>" THEN PRINT TAB(6); "Especialidad: "; D6#[1];
5738         IF D6#[2]<>" THEN PRINT TAB(45); "Seccion: "; D6#[2];
5740         IF D6#[1]<>" THEN PRINT LIN(1)
5742         PRINT TAB(6); "Asignatura: "; F2#[J2[C[K1]+1, K2]]
5744         PRINT LIN(1); TAB(6); "ALUMNO"; TAB(55); "CONVOCATORIAS"
5746         PRINT TAB(5); (FOR A=1 TO 67, "="); '27"m" '27"1"
5748         FOR K3=N2+2 TO N1

```

```

5750      READ #K,K3;Q
5752      IF Q=A1 THEN DO
5754          READ #K;(FOR K4=1 TO 22,M#[K4]),T,T1#[1],T2#[1],T3#[1]
5756          IF T2#[1,J2[ C[K1 ]+1,K2 ]*2;1 ]>"T" THEN DO
5758              T#=T2#[1,J2[ C[K1 ]+1,K2 ]*2;1 ]
5760              H#=T2#[1,J2[ C[K1 ]+1,K2 ]*2-1;1 ]
5762              IF T#="Z" OR T#="V" OR T#="U" THEN DO
5764                  CONVERT H# TO A
5766                  A=A-1
5768                  CONVERT A TO N#
5770                  D#=N#+ " Y "+H#
5772                  IF T#="Z" THEN D#=D#+ " (JUN Y SEP)"
5774                  IF T#="V" THEN D#=D#+ " (FEB Y SEP)"
5776                  IF T#="U" THEN D#=D#+ " (FEB Y JUN)"
5778              DOEND
5780          ELSE DO
5782              IF T#="Y" THEN D#=H#+ " (JUN)"
5784              IF T#="X" THEN D#=H#+ " (SEP)"
5786              IF T#="W" THEN D#=H#+ " (FEB)"
5788          DOEND
5790          PRINT TAB(6);M#[1 ]+" " +M#[2 ]+" " +M#[3 ];TAB(55);D#
5792      DOEND
5794  DOEND
5796  NEXT K3
5798  IF K2<F[ C[K1 ]+1 ] THEN DO
5800      LINPUT #9;L#
5802      PRINT '27"m"
5804      PRINT '27"&a2r0C"'27"1"'27"J"
5806  DOEND
5808  NEXT K2
5810  IF K1<L1 THEN DO
5812      LINPUT #9;L#
5814      PRINT '27"m"
5816      PRINT '27"&a2r0C"'27"1"'27"J"
5818  DOEND

```

```

5820 NEXT K1
5822 IF K<H THEN DO
5824 LINPUT #9:L#
5826 PRINT '27"m"
5828 PRINT '27"&a0r0C"'27"1"'27"J"
5830 DOEND
5832 ASSIGN *,K
5834 NEXT K
5836 REM.....
5838 REM DEVOLUCION DE CONTROL Y CIERRE DE FICHERO #9. .
5840 REM.....
5842 PRINT LIN(4),'27"&a23r6C"'27&
      "&dC Para devolver control pulse 'RETURN'. Para repetir pulse'R' &
      "'27"&d0"
5844 PRINT '27"A';'27"&a23r70C";
5846 LINPUT #9:L#
5848 IF L#<>" AND L#<>"R" THEN 5844
5850 PRINT '27"m"
5852 IF L#="R" THEN 5548
5854 ASSIGN *,9
5856 CHAIN "ARCHALUM"+W#

```

>

DELTA9

```

6000 REM*****
6002 REM *
6004 REM      MODULO 10.  OBTENCION DE TABLA SEGUN SEXO Y *
6006 REM      MATRICULACION, POR PLAN Y DETALLANDO *
6008 REM      CURSO. *
6010 REM *
6012 REM*****
6014 COM W[*]
6016 DIM B#[ 4, 12 ], C[ 50 ], C1#[ 5, 16 ], C2#[ 4, 10 ], C3#[ 2, 10 ], D7#[ 3, 30 ]
6018 DIM F5#[ 10, 200 ], J1[ 50, 10 ], M#[ 23, 30 ], N#[ 5 ], T1#[ 10, 6 ], Z#[ 4 ]
6020 B#[ 1 ]=" CARRERA", B#[ 3 ]=" CURSO", B#[ 2 ]=" PLAN"
6022 C1#[ 1 ]=" INDUSTRIAL", C1#[ 2 ]=" TELECOMUNICACION", C1#[ 5 ]=" NAVAL"
6024 C1#[ 4 ]=" OBRAS PUBLICAS", C1#[ 3 ]=" TOPOGRAFIA"
6026 C2#[ 1 ]=" PRIMERO", C2#[ 2 ]=" SEGUNDO", C2#[ 3 ]=" TERCERO", C2#[ 4 ]=&
" CUARTO"
6028 C3#[ 1 ]=" TRIENAL", C3#[ 2 ]=" CUATRIENAL"
6030 FILES *,*,*,*,*,*,*,*,*
6032 SYSTEM X,"file qqqq=#stdin;rec=-1"
6034 ASSIGN "QQQQ",9,X
6036 REM.....
6038 REM      FORMATO PAGINA 1. OBTENCION DE CLAVE.
6040 REM.....
6042 PRINT '27"h'"27"J"
6044 PRINT '27"&air6C"'27&
      "&dC Introduzca clave para seleccionar carrera, plan y curso      &
      "'27"&d@"
6046 PRINT LIN(2);TAB(11);"Carrera:";TAB(33);"Plan:";TAB(54);"Curso:"
6048 PRINT TAB(11);"-----";TAB(33);"-----";TAB(54)&
; "-----"
6050 PRINT TAB(11);"1.  " ;C1#[ 1 ];TAB(33);"1.  " ;C3#[ 1 ];TAB(54);&
"0.  TODOS"
6052 PRINT TAB(11);"2.  " ;C1#[ 2 ];TAB(33);"2.  " ;C3#[ 2 ];TAB(54);"1.  " ;C2#[
[ 1 ]
6054 PRINT TAB(11);"3.  " ;C1#[ 3 ];TAB(54);"2.  " ;C2#[ 2 ]

```

```

6056 PRINT TAB(11);"4. ";C1#[4];TAB(54);"3. ";C2#[3]
6058 PRINT TAB(11);"5. ";C1#[5];TAB(54);"4. ";C2#[4]
6060 PRINT '27"&a14r6C";"CLAVE?";
6062 PRINT '27"&a14r12C''27"&dC "'27"&d0''27"&a14r12C";
6064 ENTER 255,A,Z#
6066 IF Z#[1;1]>"5" OR Z#[1;1]<"1" THEN 6062
6068 IF Z#[2;1]>"2" OR Z#[2;1]<"1" THEN 6062
6070 IF Z#[3;1]>"4" OR Z#[3;1]<"0" THEN 6060
6072 IF Z#[2;1]="1" AND Z#[3;1]="4" THEN 6062
6074 PRINT '27"&a22r6C''27&
      "&dC Para visualizar pagina siguiente pulse 'RETURN'.
      "'27"&d0"
6076 LINPUT #9;L#
6078 REM.....
6080 REM   FORMATO PAGINA 2. APERTURA FICHERO 'C'. LISTADO TABULAR
6082 REM   CIERRE FICHERO 'C'. DEVOLUCION DE CONTROL Y CIERRE
6084 REM   FICHERO #9
6086 REM.....
6088 PRINT '27"h''27"J"
6090 CONVERT Z#[3;1] TO D
6092 IF D=0 AND Z#[2;1]="2" THEN L=1,H=4
6094 IF D=0 AND Z#[2;1]="1" THEN L=1,H=3
6096 IF D<>0 THEN L=H=D
6098 FOR K=L TO H
6100   CONVERT K TO Z#[3;1]
6102   ASSIGN "C"+Z#[1;3]+W#,K,X1
6104   READ #K,1;N#,N1,N4,N2,D7#[1],D7#[3],D7#[2]
6106   PRINT LIN(2);'27"U"
6108   PRINT TAB(4);"Carrera: ";D7#[1];TAB(33);"Plan: ";D7#[2];TAB(55)&
      ";Curso: ";D7#[3];LIN(1)
6110   FOR K1=1 TO 14
6112     J1[K1,1]=J1[K1,2]=J1[K1,3]=J1[K1,4]=J1[K1,5]=J1[K1,6]=0
6114   NEXT K1
6116   FOR J=N2+2 TO N1
6118     READ #K,J;Q,(FOR K1=1 TO 22,M#[K1]),T,T1#[1]

```

```

6120     CONVERT M$(12) TO M1
6122     M1=M1-16
6124     IF M1>13 THEN M1=14
6126     IF M1<1 THEN M1=1
6128     IF M$(4,1:1)="V" THEN DO
6130         J1(M1,5)=J1(M1,5)+1
6132         IF T1$(1,6:1)="0" THEN J1(M1,1)=J1(M1,1)+1
6134         ELSE J1(M1,2)=J1(M1,2)+1
6136     DOEND
6138     ELSE DO
6140         J1(M1,6)=J1(M1,6)+1
6142         IF T1$(1,6:1)="0" THEN J1(M1,3)=J1(M1,3)+1
6144         ELSE J1(M1,4)=J1(M1,4)+1
6146     DOEND
6148     NEXT J
6150     C(7)=0,F5$(7)=&
    "
    "
6152     FOR K1=1 TO 6
6154         C(K1)=0,F5$(K1)=&
    "
    "
6156     FOR K2=1 TO 14
6158         C(K1)=C(K1)+J1(K2,K1)
6160         CONVERT J1(K2,K1) TO N$
6162         F5$(K1,9+4*K2-LEN(N$);LEN(N$)]=N$
6164     NEXT K2
6166     F5$(K1,65:1)=":"
6168     CONVERT C(K1) TO N$
6170     F5$(K1,72-LEN(N$);LEN(N$)]=N$
6172     NEXT K1
6174     F5$(1,1:8)="V. NVO.:"
6176     F5$(2,1:8)="V. REP.:"
6178     F5$(3,1:8)="H. NVO.:"
6180     F5$(4,1:8)="H. REP.:"

```

&

&

```

6182 F5#[5,1;8]="TOT. V.:"
6184 F5#[6,1;8]="TOT. H.:"
6186 F5#[7,1;14]="TOTAL ALUMNOS:"
6188 C[7]=C[5]+C[6]
6190 CONVERT C[7] TO N#
6192 F5#[7,72-LEN(N#);LEN(N#)]=N#
6194 PRINT "SEXO Y :                                EDAD ";&
"
: TOTAL"
6196 PRINT "MATRIC.: 17 18 19 20 21 22 23 24 25 26 27";&
" 28 29 >29:"
6198 PRINT "-----";&
"-----"
6200 PRINT F5#[1]
6202 PRINT "-----";&
"-----"
6204 PRINT F5#[2]
6206 PRINT "-----";&
"-----"
6208 PRINT LIN(1);F5#[5];LIN(2)
6210 PRINT F5#[3]
6212 PRINT "-----";&
"-----"
6214 PRINT F5#[4]
6216 PRINT "-----";&
"-----"
6218 PRINT LIN(1);F5#[6];LIN(2)
6220 PRINT F5#[7]
6222 ASSIGN *,K
6224 LINPUT #9;L#
6226 NEXT K
6228 PRINT TAB(6);'27&
"&dC Para devolver control pulse 'RETURN'. Para repetir pulse'R' &
"'27"&d@"
6230 PRINT '27"A'
6232 LINPUT #9;L#

```

```
6234 IF L$<>" " AND L$<>"R" THEN 6230
6236 IF L$="R" THEN 6042
6238 ASSIGN *,9
6240 CHAIN "ARCHALUM"+W$
>
```

DELTA10

```
6500 REM*****
6502 REM *
6504 REM     MODULO 11.  OBTENCION DE LISTADO DE ASIGNATURAS *
6506 REM     EXISTENTES INDICANDO ESPECIALIDAD *
6508 REM     (EN SU CASO) POR CURSO PLAN Y CARRERA *
6510 REM *
6512 REM*****
6514 COM W[*]
6516 DIM B#[ 4, 12 ], C1#[ 5, 15 ], C2#[ 4, 10 ], C3#[ 2, 10 ], D5#[ 3, 30 ], D6#[ 2, 30 ]
6518 DIM D7#[ 3, 30 ], F2#[ 15, 65 ], N#[ 5 ], Z#[ 4 ]
6520 B#[ 1 ]="CARRERA", B#[ 3 ]="CURSO", B#[ 2 ]="PLAN"
6522 C1#[ 1 ]="INDUSTRIAL", C1#[ 2 ]="TELECOMUNICACION", C1#[ 5 ]="NAVAL"
6524 C1#[ 4 ]="OBRAS PUBLICAS", C1#[ 3 ]="TOPOGRAFIA"
6526 C2#[ 1 ]="PRIMERO", C2#[ 2 ]="SEGUNDO", C2#[ 3 ]="TERCERO", C2#[ 4 ]=&
    "CUARTO"
6528 C3#[ 1 ]="TRIENAL", C3#[ 2 ]="CUATRIENAL"
6530 FILES *,*,*,*,*,*,*,*,*
6532 SYSTEM X,"file qqqq=#stdin;rec=-1"
6534 ASSIGN "QQQQ",9,X
6536 REM.....
6538 REM     FORMATO PAGINA 1. INTRODUCCION DE CLAVE *
6540 REM.....
6542 PRINT '27"h''27"J"
6544 PRINT '27"&a1r6C''27&
    "&dC Introduzca clave para seleccionar carrera, plan y curso *
    "'27"&d@"
6546 PRINT LIN(2);TAB(11);"Carrera:";TAB(33);"Plan:";TAB(54);"Curso:"
6548 PRINT TAB(11);"-----";TAB(33);"-----";TAB(54)&
    ;"-----"
6550 PRINT TAB(11);"1.  ";C1#[ 1 ];TAB(33);"1.  ";C3#[ 1 ];TAB(54);&
    "0. TODOS"
6552 PRINT TAB(11);"2.  ";C1#[ 2 ];TAB(33);"2.  ";C3#[ 2 ];TAB(54);"1.  ";C2#[ &
    [ 1 ]
6554 PRINT TAB(11);"3.  ";C1#[ 3 ];TAB(54);"2.  ";C2#[ 2 ]
```

```

6556 PRINT TAB(11);"4. ";C1#[4];TAB(54);"3. ";C2#[3]
6558 PRINT TAB(11);"5. ";C1#[5];TAB(54);"4. ";C2#[4]
6560 PRINT '27"&a14r6C";"CLAVE?";
6562 PRINT '27"&a14r12C"'27"&dC "'27"&d0"'27"&a14r12C";
6564 ENTER 255,A,Z#
6566 IF Z#[1;1]>"5" OR Z#[1;1]<"1" THEN 6562
6568 IF Z#[2;1]>"2" OR Z#[2;1]<"1" THEN 6562
6570 IF Z#[3;1]>"4" OR Z#[3;1]<"0" THEN 6562
6572 IF Z#[2;1]="1" AND Z#[3;1]="4" THEN 6562
6574 IF Z#[3;1]="0" THEN DO
6576 PRINT '27"&a22r6C"'27&
      "&dC Para visualizar pagina siguiente pulse 'RETURN'.
      "'27"&d0"
6578 LINPUT #9;L#
6580 DOEND
6582 CONVERT Z#[3;1] TO D
6584 IF D=0 AND Z#[2;1]="2" THEN L=1,H=4
6586 IF D=0 AND Z#[2;1]="1" THEN L=1,H=3
6588 IF D<>0 THEN L=H=D
6590 REM,.....
6592 REM FORMATO DE PAGINA 2. APERTURA FICHERO C. LISTADO DE
6594 REM ASIGNATURAS. CIERRE FICHERO C. DEVOLUCION DE CONTROL Y
6596 REM CIERRE FICHERO #9.
6598 REM.....
6600 PRINT '27"h"'27"J"
6602 FOR K=L TO H
6604 CONVERT K TO Z#[3;1]
6606 ASSIGN "C"+Z#[1;3]+W#,K,X1
6608 READ #K,1;N#,N1,N4,N2,D7#[1],D7#[3],D7#[2]
6610 PRINT TAB(6);"Carrera: ";D7#[1];TAB(35);"Plan: ";D7#[2];TAB(57)&
      ";Curso: ";D7#[3];LIN(1)
6612 FOR J=2 TO N2+1
6614 READ #K,J;D6#[1],D6#[2],A1,B,(FOR K1=1 TO B,F2#[K1])
6616 IF D6#[1]<>" THEN PRINT TAB(6);"Especialidad: ";D6#[1];
6618 IF D6#[2]<>" THEN PRINT TAB(45);"Seccion: ";D6#[2];

```

```

6620     IF D6#[1]<>" THEN PRINT LIN(1)
6622     PRINT TAB(6);"Asignaturas:"
6624     PRINT TAB(5);(FOR A=1 TO 67,"=";)
6626     FOR K1=1 TO B
6628         PRINT TAB(6);F2#[K1]
6630     NEXT K1
6632     IF J<N2+1 THEN DO
6634         LINPUT #9;L$
6636         PRINT '27"&a2r0C"'27"J"
6638         PRINT '27"V"'27"&a2r0C"
6640     DOEND
6642     NEXT J
6644     ASSIGN *,K
6646     IF K<H THEN DO
6648         LINPUT #9;L$
6650         PRINT '27"&a0r0C"'27"U"
6652         PRINT LIN(4);'27"U"
6654     DOEND
6656     NEXT K
6658     PRINT LIN(4),'27"&a22r6C"'27&
        "&dC Para devolver control pulse 'RETURN'. Para repetir pulse 'R' &
        "'27"&d0"
6660     PRINT '27"A"':'27"&a22r70C";
6662     LINPUT #9;L$
6664     IF L#<>" AND L#<>"R" THEN 6660
6666     IF L#="R" THEN 6542
6668     ASSIGN *,9
6670     CHAIN "ARCHALUM"+W$

```

>

DELTA11

```
7000 REM*****
7002 REM *
7004 REM      MODULO 12.      VISUALIZACION DE LA FICHA DE *
7006 REM      EXPEDIENTE ACTUALIZADA DE UN ALUMNO *
7008 REM *
7010 REM*****
7012 COM W[* ]
7014 DIM A#[ 3, 30 ], C#[ 22, 30 ], D#[ 50 ], D5#[ 3, 30 ], D6#[ 2, 30 ], D7#[ 3, 30 ]
7016 DIM F2#[ 15, 70 ], F5#[ 10, 200 ], G#[ 8 ], H#[ 2 ], M#[ 23, 30 ], N#[ 5 ]
7018 DIM T1#[ 10, 6 ], T2#[ 10, 31 ], T3#[ 10, 45 ]
7020 C#[ 1 ]="PRIMER APELLIDO"
7022 C#[ 2 ]="SEGUNDO APELLIDO"
7024 C#[ 3 ]="NOMBRE"
7026 C#[ 4 ]="SEXO"
7028 C#[ 5 ]="NACIONALIDAD"
7030 C#[ 6 ]="NO. DOC. IDENT. O PASAPORTE"
7032 C#[ 7 ]="FECHA EXPEDICION"
7034 C#[ 8 ]="LUGAR DE EXPEDICION"
7036 C#[ 9 ]="LUGAR DE NACIMIENTO"
7038 C#[ 10 ]="PROVINCIA"
7040 C#[ 11 ]="FECHA"
7042 C#[ 12 ]="EDAD"
7044 C#[ 13 ]="ESTADO CIVIL"
7046 C#[ 14 ]="NOMBRE DEL PADRE"
7048 C#[ 15 ]="NOMBRE DE LA MADRE"
7050 C#[ 16 ]="REPRESENTANTE LEGAL"
7052 C#[ 17 ]="DOMICILIO (LOCALIDAD)"
7054 C#[ 18 ]="DOMICILIO (CALLE)"
7056 C#[ 19 ]="TELEFONO"
7058 C#[ 20 ]="PROCEDENCIA (COU O FP (ESPECIALIDAD))"
7060 C#[ 21 ]="TIPO DE ENSEANZA"
7062 C#[ 22 ]="NO. DE EXPEDIENTE"
7064 FILES *,*,*,*,*,*,*,*,*
7066 SYSTEM X,"FILE @@@@=$STDIN;REC=-1"
```

```

7068 ASSIGN "QQQQ",9,A
7070 ASSIGN "HELP"+W$,5,A
7072 READ #5,1:A,G$
7074 ASSIGN *,5
7076 ASSIGN G$+W$,6,A
7078 REM.....
7080 REM  FORMATO DE PAGINA 1. INTRODUCCION DE NOMBRE Y APELLIDOS .
7082 REM  DE UN ALUMNO Y LOCALIZACION EN EL ARCHIVO DE EXPEDIENTES .
7084 REM.....
7086 READ #6,1:H1
7088 PRINT '27"h"'27"J"
7090 PRINT '27"A";"Primer apellido";TAB(26);"Segundo apellido";TAB(51)&
; "Nombre"
7092 PRINT '27"&dC          "'27"&d@      "'27&
"&dC          "'27"&d@      "'27&
"&dC          "'27"&d@";
7094 PRINT '27"A"'27"A";
7096 PRINT '27"[          "'27" ]      "'27&
"C          "'27" ]      "'27"C          "'27" ]";
7098 FOR K=1 TO 3
7100 PRINT '27"W";
7102 PRINT (FOR K1=1 TO K-1,(FOR K2=1 TO 25,'27"C";));
7104 ENTER 255,A,A#[K]
7106 A#[K]=UPS$(DEB$(A#[K]))
7108 PRINT '27"A"'27"X"
7110 NEXT K
7112 IF H1=3 THEN H1=4
7114 ON END #6 THEN 7140
7116 L=2,H=H1
7118 M=INT((L+H)/2)
7120 IF M=1 THEN 7140
7122 READ #6,M:A,(FOR K=1 TO 3,M#[K])
7124 IF M#[1]+M#[2]+M#[3]>=A#[1]+A#[2]+A#[3] THEN 7132
7126 IF L>=H THEN 7140
7128 L=M+1

```

```

7130 GOTO 7118
7132 IF M#[1]+M#[2]+M#[3]=A#[1]+A#[2]+A#[3] THEN 7146
7134 IF L>=H THEN 7140
7136 H=M-1
7138 GOTO 7118
7140 PRINT '27"&a22r0C"'27&
      "&dC
          "'27"0", '27"A", '27"A"
7142 PRINT &
      "NO HAY NADIE MATRICULADO CON ESE NOMBRE. OTRO ALUMNO(SI/NO)?";
7144 GOTO 7436
7146 READ #6, M1Q1, (FOR K=1 TO 22, M#[K]), T, (FOR K=1 TO T, T1#[K], T2#[K], &
      F5#[K])
7148 REM.....
7150 REM LISTADO DE DATOS PERSONALES DEL ALUMNO
7152 REM.....
7154 PRINT '27"&a2r0C"i&
      "Sexo Nacionalidad DNI o Pasaporte Fecha exp. Lugar de expe&
dicion"
7156 PRINT '27"&a3r2C"i'27"&dC "'27"&d0 "'27"&dC "'27&
      "&d0 "'27"&dC "'27"&d0 "'27"&dC "'27&
      "&d0 "'27"&dC "'27"&d0"
7158 PRINT '27"&a3r0C"
7160 PRINT M#[4];TAB(5);M#[5];TAB(22);M#[6];TAB(40);M#[7];TAB(52);M#[8&
      ]
7162 PRINT '27"&a5r0C"i&
      "Lugar de nacimiento Provincia Fecha Edad Estad&
o civil"
7164 PRINT '27"&a7r0C"'27"&dC "'27"&d0 "'27&
      "&dC "'27"&d0 "'27"&dC "'27"&d0 "'27&
      "&dC "'27"&d0 "'27"&dC "'27"&d0"
7166 PRINT '27"&a6r0C"
7168 PRINT M#[9];TAB(21);M#[10];TAB(42);M#[11];TAB(54);M#[12];TAB(60);&
      M#[13]
7170 PRINT '27"&a8r0C"i&

```

```

"Nombre del padre      Nombre de la madre      Representante&
legal"
7172 PRINT '27"&a10r0C"'27"&dC      "'27"&d0      "'27&
"&dC      "'27"&d0      "'27&
"&dC      "'27"&d0"
7174 PRINT '27"&a9r0C"
7176 PRINT M#[14];TAB(26);M#[15];TAB(52);M#[16]
7178 PRINT '27"&a11r0C"i&
" Domicilio (Localidad)      Domicilio (Calle)      Telefon&
o"
7180 PRINT '27"&a13r0C"'27"&dC      "'27"&d0      "'27&
"&dC      "'27"&d0      "'27&
"&dC      "'27"&d0"
7182 PRINT '27"&a12r0C"
7184 PRINT M#[17];TAB(25);M#[18];TAB(58);M#[19]
7186 PRINT '27"&a15r0C"i&
" Procedencia (COU/FP)      Tipo ensenanza      No. expediente"
7188 PRINT '27"&a16r0C"'27"&dC      "'27"&d0      "'27&
"&dC      "'27"&d0      "'27"&dC      "'27"&d0"
7190 PRINT '27"&a15r0C"
7192 PRINT M#[20];TAB(29);M#[21];TAB(43);M#[22]
7194 PRINT '27"&a22r6C"'27&
"&dC Para visualizar pagina siguiente pulse 'RETURN'.      "&
'27"&d0"
7196 PRINT '27"&a22r65C";
7198 LINPUT #9;L$
7200 IF L$<>" " THEN 7196
7202 REM.....
7204 REM   FORMATO PAGINA 2. LISTADO DE CALIFICACIONES Y CONVOCATORIAS.
7206 REM.....
7208 PRINT '27"U"
7210 PRINT TAB(17);"CALIFICACIONES Y CONVOCATORIAS";LIN(1)
7212 PRINT '27"1"
7214 FOR K=1 TO T
7216   ASSIGN T1#[K,1;4]+W$,1,X1

```

```

7218 READ #1,1;N#,N1,N4,N2,D7#[1],D7#[3],D7#[2]
7220 CONVERT T1#[K,5;1] TO A
7222 READ #1,1+A;D6#[1],D6#[2],A1,B,(FOR K1=1 TO B,F2#[K1])
7224 PRINT TAB(3);"Carrera: ";D7#[1];TAB(32);"Plan: ";D7#[2];TAB(54)&
;"Curso: ";D7#[3];LIN(1);
7226 IF D6#[1]<>" " THEN PRINT LIN(1);TAB(3);"ESPECIALIDAD: ";D6#[1];
7228 IF D6#[2]<>" " THEN PRINT TAB(42);"SECCION: ";D6#[2];
7230 IF D6#[1]<>" " AND D6#[2]=" " THEN PRINT
7232 PRINT (FOR K1=1 TO 72,"=";)
7234 REM.....
7236 REM LOCALIZACION DEL ALUMNO EN FICHERO TIPO 'C'
7238 REM.....
7240 IF N1>N2+1 THEN DO
7242 L=N2+2,H=N1
7244 M=INT((L+H)/2)
7246 READ #1,M;Q,D5#[1],D5#[2],D5#[3]
7248 IF D5#[1]+D5#[2]+D5#[3]>=M#[1]+M#[2]+M#[3] THEN 7256
7250 IF L>=H THEN 7266
7252 L=M+1
7254 GOTO 7244
7256 IF D5#[1]+D5#[2]+D5#[3]=M#[1]+M#[2]+M#[3] THEN 7268
7258 IF L>=H THEN 7266
7260 H=M-1
7262 GOTO 7244
7264 DOEND
7266 M=0
7268 REM.....
7270 REM BUSQUEDA DE CALIFICACIONES Y CONV. EN FICHERO TIPO 'C'
7272 REM Y LISTADO DE DATOS
7274 REM.....
7276 FOR K1=1 TO B
7278 IF T2#[K,K1*2;1]<"H" THEN 7384
7280 ELSE DO
7282 IF M<>0 THEN DO
7284 READ #1,M;Q,(FOR K2=1 TO 22,M#[K2]),E,T1#[K],T2#[K],T3#[K&

```

```

]
7286     E$="M"
7288     FOR K2=1 TO 3
7290         IF T3$(K,K1*3-K2+1;1)<>"0" THEN DO
7292             IF T3$(K,K1*3-K2+1;1)<E$ THEN E$=T3$(K,K1*3-K2+1;1)
7294         DOEND
7296     NEXT K2
7298     IF E$<"H" THEN DO
7300         T2$(K,K1*2;1)=E$
7302         GOTO 7384
7304     DOEND
7306 DOEND
7308     IF T2$(K,K1*2;1)>"T" OR T2$(K,K1*2;1)="S" THEN DO
7310         D$="MTDO."
7312         IF T2$(K,K1*2;1)="Z" OR T2$(K,K1*2;1)="V" OR T2$(K,K1*2;1&
J="U" THEN DO
7314             IF T2$(K,K1*2;1)="Z" THEN D$=D$+" JUN Y SEP"
7316             IF T2$(K,K1*2;1)="V" THEN D$=D$+" FEB Y SEP"
7318             IF T2$(K,K1*2;1)="U" THEN D$=D$+" FEB Y JUN"
7320             CONVERT T2$(K,K1*2-1;1) TO H
7322             H=H-1
7324             CONVERT H TO H$
7326             D$=D$+" (" +H$+" Y "+T2$(K,K1*2-1;1)+")"
7328         DOEND
7330     ELSE DO
7332         IF T2$(K,K1*2;1)="X" THEN D$=D$+" SEP (" +T2$(K,K1*2-1;1&
J+)" )"
7334         IF T2$(K,K1*2;1)="Y" THEN D$=D$+" JUN (" +T2$(K,K1*2-1;1&
J+)" )"
7336         IF T2$(K,K1*2;1)="W" THEN D$=D$+" FEB (" +T2$(K,K1*2-1;1&
J+)" )"
7338         IF T2$(K,K1*2;1)="S" THEN D$="MATRICULADO"
7340     DOEND
7342 DOEND
7344 ELSE DO

```

```

7346         D$="NO MATRICULADO"
7348         IF T2#[K,K1*2-1;1]<>"0" THEN DO
7350             D$="NO MTD0. (" +T2#[K,K1*2-1;1]+ " CONV. AGT. )"
7352         DOEND
7354     DOEND
7356     PRINT TAB(1);F2#[K1];TAB(50);D$
7358     IF T2#[K,B*2;1]<>"S" THEN DO
7360         I2=LEN(F5#[K])
7362         FOR I3=1 TO INT(I2/4)
7364             CONVERT K1 TO E$
7366             IF F5#[K,I3*4-2;1]=E$ THEN DO
7368                 IF F5#[K,I3*4-1;1]="1" THEN PRINT TAB(58);"FEB "+F5#[&
K,I3*4;2]+": NP"
7370                 IF F5#[K,I3*4-1;1]="2" THEN PRINT TAB(58);"JUN "+F5#[&
K,I3*4;2]+": NP"
7372                 IF F5#[K,I3*4-1;1]="3" THEN PRINT TAB(58);"SEP "+F5#[&
K,I3*4;2]+": NP"
7374             DOEND
7376         NEXT I3
7378     DOEND
7380     GOTO 7402
7382     DOEND
7384     IF T2#[K,K1*2;1]="A" THEN D$="MATRIC. H. "
7386     IF T2#[K,K1*2;1]="B" THEN D$="SOBRES. 10"
7388     IF T2#[K,K1*2;1]="C" THEN D$="SOBRES. 9"
7390     IF T2#[K,K1*2;1]="D" THEN D$="NOTABLE 8"
7392     IF T2#[K,K1*2;1]="E" THEN D$="NOTABLE 7"
7394     IF T2#[K,K1*2;1]="F" THEN D$="APROBADO 6"
7396     IF T2#[K,K1*2;1]="G" THEN D$="APROBADO 5"
7398     D$=D$+" (CONVOC. "+T2#[K,K1*2-1;1]+")"
7400     PRINT TAB(1);F2#[K1,1,48];TAB(50);D$
7402     NEXT K1
7404     IF T1#[K,3;2]="13" OR T1#[K,3;2]="24" THEN DO
7406         IF Q1<1 AND T2#[K,2*B;1]<>"T" THEN DO
7408             PRINT LIN(1);TAB(1);"TITULO DEL PROYECTO:"

```

```

7410      PRINT TAB(1);F5*[K]
7412      DOEND
7414      DOEND
7416      PRINT
7418      ASSIGN *,1
7420      NEXT K
7422      REM.....
7424      REM  DEVOLUCION DE CONTROL Y CIERRE DE FICHEROS #5, #6 Y #9
7426      REM.....
7428      PRINT '27"&a21r-5C"
7430      PRINT '27&
          "&dC
          " '27"&d@" '27"A";
7432      PRINT (FOR K4=1 TO 71," ");". ";'27"G";'27"A";
7434      PRINT " OTRO ALUMNO? (SI/NO)";
7436      ENTER 255,A,H#
7438      IF H#="SI" OR H#="S" THEN DO
7440      PRINT '27"m"
7442      GOTO 7088
7444      DOEND
7446      IF H#="NO" OR H#="N" THEN DO
7448      PRINT '27"m"
7450      ASSIGN *,6
7452      ASSIGN *,9
7454      DOEND
7456      ELSE GOTO 7432
7458      CHAIN "ARCHALUM"+W#

```

>

DELTA12

```
7500 REM*****
7502 REM
7504 REM      MODULO 13.      OBTENCION DE LISTADO POR NACIONALIDAD
7506 REM
7508 REM*****
7510 COM W[*]
7512 DIM C[50],D#[50],F3#[45,65],G#[8],H2#[31],H3#[45],M#[23,30]
7514 FILES *,*,*,*,*,*,*,*,*
7516 ASSIGN "HELP"+W#,5,X1
7518 READ #5,1;N3,G#
7520 ASSIGN *,5
7522 ASSIGN G#+W#,6,X1
7524 READ #6,1;N5,N6
7526 SYSTEM X,"FILE QQQQ=#STDIN;REC=-16"
7528 ASSIGN "QQQQ",9,X
7530 REM.....
7532 REM      FORMATO PAGINA 1. SELECCION DE NACIONALIDAD
7534 REM.....
7536 PRINT '27"h"'27"J"
7538 PRINT '27"&a9r4C"'27&
      "&dC Indique nacionalidad que desea listar. Para listar todas puls&
e 'RETURN' "'27"&d@"
7540 PRINT '27"&a13r22C";"Nacionalidad? "'27"&dC      "'27&
      "&d@"
7542 PRINT '27"&a13r36C";
7544 LINPUT #9;D#
7546 REM.....
7548 REM      FORMATO PAGINA 2.
7550 REM      OPCION A: LISTADO ALFABETICO POR NACIONALIDAD.
7552 REM      OPCION B: LISTADO DE TODAS LAS NACIONALIDADES.
7554 REM      DEVOLUCION DE CONTROL Y CIERRE DE FICHEROS.
7556 REM.....
7558 PRINT '27"h"'27"J"
7560 IF D#="" THEN 7598
```

```

7562 D#=UPS$(DEB$(D#))
7564 IF D#<>" " THEN DO
7566 PRINT TAB(9);"NACIONALIDAD ";D#
7568 PRINT TAB(6):(FOR K=1 TO 66,"=");LIN(1)
7570 PRINT '27*1"
7572 J=0
7574 FOR K=2 TO N5
7576 READ #6,K;Q,M#[1],M#[2],M#[3],M#[4],M#[5]
7578 H2#=M#[5,1,LEN(M#[5])-1]
7580 H3#=D#[1,LEN(D#)-1]
7582 IF H2#=D# OR H2#=H3# OR M#[5]=H3# THEN DO
7584 J=J+1
7586 PRINT TAB(9);M#[1]+" "+M#[2]+" "+M#[3]
7588 DOEND
7590 NEXT K
7592 PRINT LIN(2);TAB(9);"Total alumnos de nacionalidad ";D#;" ";J
7594 DOEND
7596 ELSE DO
7598 F3#[1]="ESPANOLA",C[1]=0,I=1
7600 FOR K=2 TO N5
7602 READ #6,K;Q,M#[1],M#[2],M#[3],M#[4],M#[5]
7604 M#[5]=DEB$(M#[5])
7606 FOR J=1 TO I
7608 A=LEN(M#[5])-LEN(F3#[J])
7610 IF A=1 OR A=0 OR A=-1 THEN DO
7612 IF A=1 THEN DO
7614 IF M#[5,1,LEN(M#[5])-1]=F3#[J] THEN DO
7616 C[J]=C[J]+1
7618 GOTO 7654
7620 DOEND
7622 DOEND
7624 IF A=0 THEN DO
7626 IF M#[5,1,LEN(M#[5])-1]=F3#[J,1,LEN(F3#[J])-1] THEN DO
7628 C[J]=C[J]+1
7630 GOTO 7654

```

```

7632         DOEND
7634         DOEND
7636         IF A=-1 THEN DO
7638             IF M#[5]=F3#[J,1,LEN(F3#[J])-1] THEN DO
7640                 C[J]=C[J]+1
7642                 GOTO 7654
7644             DOEND
7646         DOEND
7648         DOEND
7650         NEXT J
7652         I=I+1,F3#[I]=M#[5],C[I]=1
7654     NEXT K
7656     PRINT TAB(9);"NACIONALIDAD";TAB(30);"No. ALUMNOS"
7658     PRINT TAB(6);(FOR K=1 TO 66,"=");LIN(1)
7660     PRINT '27"1"
7662     FOR K=1 TO I
7664         PRINT TAB(9);F3#[K];TAB(30);C[K]
7666     NEXT K
7668 DOEND
7670 ASSIGN *,9
7672 SYSTEM X,"FILE QQQQ=$STDIN;REC=-1"
7674 ASSIGN "QQQQ",9,X
7676 PRINT LIN(4),'27"&a23r6C"'27&
    "&dC Para devolver control pulse 'RETURN'. Para repetir pulse'R' &
    "'27"&d@"
7678 PRINT '27"A";'27"&a23r70C";
7680 LINPUT #9;L$
7682 IF L$<>" " AND L$<>"R" THEN 7678
7684 PRINT '27"m"
7686 ASSIGN *,9
7688 IF L$="R" THEN 7526
7690 ASSIGN *,6
7692 CHAIN "ARCHALUM"+W$

```

>

DELTA13

```
8000 REM*****
8002 REM *
8004 REM      MODULO 14.      OBTENCION DE LISTADO ALFABETICO *
8006 REM      POR CARRERA TERMINADA *
8008 REM *
8010 REM*****
8012 COM W[*]
8014 DIM G#[8],L1#[2],L2#[2],M#[23,30]
8016 FILES *,*,*,*,*,*,*,*,*
8018 SYSTEM X,"FILE @@@@=#STDINX;REC=-1"
8020 ASSIGN "QQQQ",9,X
8022 ASSIGN "HELP"+W#,5,X1
8024 READ #5,1;N3,G#
8026 ASSIGN *,5
8028 ASSIGN G#+W#,6,X1
8030 READ #6,1;N5
8032 REM.....
8034 REM      FORMATO PAGINA 1. INTRODUCCION DE LIMITES DE LISTADO. .
8036 REM.....
8038 PRINT '27"h" '27"J"
8040 PRINT '27"&a8r5C" '27&
      "&dC Introduzca limites de listado alfabetico si los desea &
      "'27"&d@"
8042 PRINT '27"&a11r13C";"Limite inferior? "'27"&dC "'27"&d@" '27&
      "&a11r45C";"Limite superior? "'27"&dC "'27"&d@"
8044 PRINT '27"&a11r30C";
8046 LINPUT #9;L1#
8048 L1#=UPS$(L1#)
8050 IF L1#<>" " AND (L1#<"A" OR L1#>"Z") THEN 8044
8052 PRINT '27"&a11r62C";
8054 LINPUT #9;L2#
8056 L2#=UPS$(L2#)
8058 IF L2#<>" " AND (L2#<"A" OR L2#>"Z") THEN 8052
```

```

8060 IF L2$="" THEN L2$="ZZ"
8062 REM.....
8064 REM   FORMATO PAGINA 2. LISTADO DE ALUMNOS. DEVOLUCION DE
8066 REM   CONTROL Y CIERRE DE FICHEROS.
8068 REM.....
8070 PRINT '27"h"'27"J"
8072 PRINT TAB(9),"ALUMNOS CON CARRERA TERMINADA"
8074 PRINT TAB(6);(FOR K=1 TO 66,"=";)
8076 PRINT '27"1"
8078 L=2,H=N5
8080 M=INT((L+H)/2)
8082 IF M=1 THEN M=2
8084 READ #6,M;Q,M#[1],M#[2],M#[3]
8086 IF M#[1]>=L1$ THEN B100
8088 IF L>=H THEN DO
8090   M=M+1
8092   GOTO B106
8094 DOEND
8096 L=M+1
8098 GOTO B080
8100 IF L>=H THEN B106
8102 H=M-1
8104 GOTO B080
8106 ON END #6 THEN B136
8108 FOR K=M TO N5
8110   READ #6,K;Q
8112   IF Q<0 THEN DO
8114     READ #6;M#[1],M#[2],M#[3]
8116     IF M#[1]<=L2$ THEN PRINT TAB(9);M#[1]+ "   "+M#[2]+ "   "+M#[3]
8118     ELSE GOTO B124
8120   DOEND
8122 NEXT K
8124 PRINT LIN(4);'27"&a23r6C"'27&
      "&dC Para devolver control pulse 'RETURN'. Para repetir pulse 'R' &
      "'27"&d0"

```

```
8126 PRINT '27"A"'27"&a23r70C";
8128 LINPUT #9:L$
8130 IF L$<>" " AND L$<>"R" THEN 8126
8132 PRINT '27"m"
8134 IF L$="R" THEN 8038
8136 ASSIGN *,6
8138 ASSIGN *,9
8140 CHAIN "ARCHALUM"+W$
```

>

```

9000 REM*****
9002 REM
9004 REM          SUBROUTINA 1. (MOD 2.)      LOCALIZACION E INICIALIZACION *
9006 REM          DE LA POSICION EN FICHERO DEL *
9008 REM          DEL ALUMNO A ANADIR.      *
9010 REM
9012 REM*****
9014 IF P1=P3-1 THEN DO
9016     R0=P3
9018     RETURN
9020 DOEND
9022 READ #P2,P1;0,(FOR K1=1 TO 3,D5#[K1])
9024 IF D4#>D5#[1]+D5#[2]+D5#[3] THEN DO
9026     IF P5=5 THEN DO
9028         FOR K3=P3 TO N1
9030             READ #P2,K3;0,(FOR K1=1 TO 22,F#[K1]),T,(FOR K1=1 TO T,T1#[&
                K1],T2#[K1],T3#[K1])
9032             PRINT #P5,K3;0,(FOR K1=1 TO 22,F#[K1]),T,(FOR K1=1 TO T,T1#[&
                [K1],T2#[K1],T3#[K1])
9034         NEXT K3
9036     DOEND
9038     R0=P1+1
9040     RETURN
9042 DOEND
9044 READ #P2,P3;0,(FOR K1=1 TO 3,D5#[K1])
9046 IF D4#<D5#[1]+D5#[2]+D5#[3] THEN DO
9048     FOR K3=P1 TO P3 STEP -1
9050         READ #P2,K3;0,(FOR K1=1 TO 22,F#[K1]),T,(FOR K1=1 TO T,T1#[K1&
                ],T2#[K1],T3#[K1])
9052         PRINT #P5,K3+1;0,(FOR K1=1 TO 22,F#[K1]),T,(FOR K1=1 TO T,T1#[&
                [K1],T2#[K1],T3#[K1])
9054     NEXT K3
9056     R0=P3
9058     RETURN
9060 DOEND

```

```

9062 L=P4,H=P1
9064 M=INT((L+H)/2)
9066 READ #P2,M;Q,(FOR K1=1 TO 3,D5#[K1])
9068 IF D5#[1]+D5#[2]+D5#[3]>=D4# THEN 9082
9070 IF L>=H THEN DO
9072     M=M+1
9074     GOTO 9098
9076 DOEND
9078 L=M+1
9080 GOTO 9064
9082 IF D5#[1]+D5#[2]+D5#[3]=D4# THEN DO
9084     READ #P2,M;B1,(FOR K1=1 TO 22,F#[K1]),T,H1#,H2#,H3#
9086     R0=M,R3=1
9088     RETURN
9090 DOEND
9092 IF L>=H THEN 9098
9094 H=M-1
9096 GOTO 9064
9098 FOR K3=P1 TO M STEP -1
9100     READ #P2,K3;Q,(FOR K1=1 TO 22,F#[K1]),T,(FOR K1=1 TO T,T1#[K1],&
        T2#[K1],T3#[K1])
9102     PRINT #P5,K3+1;Q,(FOR K1=1 TO 22,F#[K1]),T,(FOR K1=1 TO T,T1#[&
        K1],T2#[K1],T3#[K1])
9104 NEXT K3
9106 R0=M
9108 IF P5=5 THEN DO
9110     FOR K3=M-1 TO P3 STEP -1
9112         READ #P2,K3;Q,(FOR K1=1 TO 22,F#[K1]),T,(FOR K1=1 TO T,T1#[K1]&
            ],T2#[K1],T3#[K1])
9114         PRINT #P5,K3;Q,(FOR K1=1 TO 22,F#[K1]),T,(FOR K1=1 TO T,T1#[&
            K1],T2#[K1],T3#[K1])
9116     NEXT K3
9118 DOEND
9120 RETURN

```

```

9300 REM*****
9302 REM
9304 REM      SUBROUTINA 2. (MOD 2.)      ACTUALIZACION DE LOS DATOS *
9306 REM      ACADEMICOS DEL ALUMNO      *
9308 REM
9310 REM*****
9312 REM.....
9314 REM      LISTADO Y SELECCION DE ASIGNATURAS A MATRICULAR .
9316 REM.....
9318 PRINT LIN(2);TAB(6);"Asignaturas: ";LIN(1);TAB(5);&
      "-----&
- "
9320 FOR K1=1 TO B
9322   CONVERT K1 TO L$
9324   PRINT TAB(6);L$;"  ";F2$[K1]
9326 NEXT K1
9328 PRINT LIN(1);'27"&a22r5C"'27&
      "&dC Introduzca numero(s) correspond. a asignatura(s) de matricula&
cion"'27"&d0"
9330 FOR W0=1 TO 9999
9332 NEXT W0
9334 PRINT '27"A";(FOR K4=1 TO 70," ");". ";'27"G";'27"A";
9336 INPUT "      Numero(s) correspondiente(s) ?":
9338 IF BUF(0)=4 THEN 9636
9340 FOR W2=1 TO 15
9342   IF BUF(0)=4 THEN 9370
9344   IF BUF(0)<>5 THEN DO
9346     PRINT '27"M";'27"A"
9348     GOTO 9334
9350   DOEND
9352   INPUT :W3[W2]:
9354   IF W3[W2]<1 OR W3[W2]>B THEN DO
9356     D$=&
      "      No encuentro asignatura que corresponda a este numero      &
!"

```

```

9358     PRINT '27"A";'7;D$
9360     ENTER 3,A,D
9362     PRINT '27"A",'27"A"
9364     GOTO 9328
9366     DOEND
9368     NEXT W2
9370     PRINT '27"h''27"J"
9372     FOR W4=1 TO W2-1
9374     C3=W3[W4]
9376     REM.....
9378     REM  MATRICULACION DE PROYECTO. VERIFICACION DE HABER APRO-
9380     REM  BADO PREVIAMENTE TODAS LAS ASIGNATURAS DE LA CARRERA
9382     REM.....
9384     IF C3=B AND (D2$[K2,3;2]="13" OR D2$[K2,3;2]="24") THEN DO
9386     CONVERT D2$[K2,4;1] TO D2
9388     E2$=D2$[K2]
9390     I9=2
9392     FOR I1=1 TO T
9394     IF I9=2 THEN I9=0
9396     FOR I3=1 TO D2
9398     CONVERT I3 TO E2$[4;1]
9400     IF F3$[I1,1;4]=E2$ THEN DO
9402     CONVERT F3$[I1,5;1] TO A3
9404     I8=0
9406     READ #I3,1;N$,N1,N4,N2
9408     L=N2+2,H=N1
9410     IF L<=H THEN DO
9412     M=INT((L+H)/2)
9414     ON END #I3 THEN 9436
9416     READ #I3,M;D1,D5$[1],D5$[2],D5$[3]
9418     IF D5$[1]+D5$[2]+D5$[3]>=F$[1]+F$[2]+F$[3] THEN 9432
9420     IF L>=H THEN DO
9422     I8=1
9424     GOTO 9448
9426     DOEND

```

```

9428          L=M+1
9430          GOTO 9412
9432          IF D5#[ 1 ]+D5#[ 2 ]+D5#[ 3 ]=F#[ 1 ]+F#[ 2 ]+F#[ 3 ] THEN 9446
9434          IF L>=H THEN DO
9436              IB=1
9438              GOTO 9448
9440          DOEND
9442          H=M-1
9444          GOTO 9412
9446          READ #I3,M;A3,(FOR I5=1 TO 22,M#[ I5 ]),E,T1#[ 1 ],T2#[ 1 ]&
, T3#[ 1 ]
9448          READ #I3,1+A3;D6#[ 1 ],D6#[ 2 ],A,B2,(FOR I5=1 TO B2,F2#[ &
I5 ])
9450          I2=LEN(F4#[ I1 ])/2
9452          IF F3#[ I1,1;4 ]=D2#[ K2 ] THEN I2=I2-1
9454          FOR I4=1 TO I2
9456              IF F4#[ I1,2*I4;1 ]>"G" THEN DO
9458                  IF IB=1 OR ((T3#[ 1,3*I4-2;1 ]>"G" OR T3#[ 1,3*I4-2;
&
1 ]="0") AND (T3#[ 1,3*I4-1;1 ]>"G" OR T3#[ 1,3*I4-1;1 ]="0")) AND (T3#[ &
1,3*I4;1 ]>"G" OR T3#[ 1,3*I4;1 ]="0")) THEN DO
9460                      IF I9=0 THEN DO
9462                          I9=1
9464                          PRINT LIN(1);&
"Para matricular de proyecto quedan por aprobar las asignaturas:"
9466                              PRINT (FOR W5=1 TO 62,"-";)
9468                              DOEND
9470                              PRINT TAB(2);F2#[ I4 ]
9472                              DOEND
9474                              DOEND
9476                              NEXT I4
9478                              DOEND
9480                              DOEND
9482                              NEXT I3
9484                              NEXT I1
9486                              PRINT

```

```

9488     IF I9=0 THEN DO
9490         Q=0
9492         FOR I1=1 TO T
9494             IF F3#[ I1,1;4]=D2#[K2] THEN DO
9496                 PRINT "TITULO DEL PROYECTO:"
9498                 ENTER 255,X1,F5#[ I1 ]
9500                 PRINT
9502                 I2=LEN(F4#[ I1 ])
9504                 F4#[ I1,I2-1;2]="3S"
9506             DOEND
9508         NEXT I1
9510     DOEND
9512 DOEND
9513 ELSE DO
9514     REM.....
9516     REM  MATRICULACION DE ASIGNATURAS. PROTECCION PARA CASOS DE
9518     REM  5a,6a Y 7a CONVOCATORIA, Y PARA ASIGNATURAS PREVIAMENTE
9520     REM  APROBADAS O MATRICULADAS
9522     REM.....
9526     IF F4#[K3,C3*2;1]<"U" THEN DO
9528         IF F4#[K3,C3*2;1]<"H" THEN DO
9530             PRINT F2#[ C3,1,50 ];" ":";TAB(52);"ESTA APROBADA ! !"
9532             GOTO 9622
9534         DOEND
9536     CONVERT F4#[K3,1+(C3-1)*2;1] TO T2
9538     IF T2>4 THEN DO
9540         REM.....
9542         REM  MATRICULACION DE 7a CONVOCATORIA
9544         REM.....
9546         SYSTEM X,"FILE @@@@=#STDINX;REC=-5"
9548         ASSIGN "@@@@",9,X
9550         IF T2>6 THEN PRINT F2#[ C3 ];" ":";LIN(1);&
"AGOTADA CONVOCATORIA COMPULSADA ! ! !","'7" "'7" "'7"
9552         IF T2=6 THEN DO
9554             PRINT F2#[ C3,1,50 ];" ":";TAB(52);"6 CONV. AGOTADAS ! !"

```

```

9556          PRINT LIN(1);TAB(6);"CONVOCATORIAS NO PRESENTADAS:"
9558          I2=LEN(F5$(K3))
9560          I4=1
9562          FOR I3=1 TO INT(I2/4)
9564              CONVERT C3 TO E2$
9566              IF F5$(K3,I3*4-2;1)=E2$ THEN DO
9568                  I4=0
9570                  IF F5$(K3,I3*4-1;1)="1" THEN PRINT TAB(15);"FEB  ";&
F5$(K3,I3*4;2]
9572                  IF F5$(K3,I3*4-1;1)="2" THEN PRINT TAB(15);"JUN  ";&
F5$(K3,I3*4;2]
9574                  IF F5$(K3,I3*4-1;1)="3" THEN PRINT TAB(15);"SEP  ";&
F5$(K3,I3*4;2]
9576              DOEND
9578          NEXT I3
9580          IF I4=1 THEN PRINT TAB(15);"NINGUNA"
9582          PRINT LIN(1);TAB(6);&
"SE ACEPTA CONVOCATORIA COMPULSADA? (SI/NO)";
9584          LINPUT #9;E2$
9586          IF E2$="SI" THEN DO
9588              T2=T2+1
9590              F4$(K3,C3*2;1)="Y"
9592          DOEND
9594          IF E2$<>"NO" AND E2$<>"SI" THEN 9582
9596          DOEND
9598          ASSIGN *,9
9600          REM.....
9602          REM  FIN DE MATRICULACION DE 7a CONVOCATORIA
9604          REM.....
9606          IF T2=5 THEN PRINT F2$(C3,1,50);";";TAB(52);&
"5 CONV. AGOTADAS !"
9608          IF T2=5 THEN F4$(K3,C3*2;1)="Y",T2=T2+1
9610          DOEND
9612          ELSE F4$(K3,C3*2;1)="Z",T2=T2+2
9614          CONVERT T2 TO F4$(K3,1+(C3-1)*2;1]

```

```
9616      DOEND
9618      ELSE PRINT F2#[C3,1,50];":":TAB(52);"ESTA MATRICULADA ! !"
9620      DOEND
9622      NEXT W4
9624      IF K2<>J-1 THEN DO
9626          PRINT '27"1"
9628          PRINT LIN(1);'27"&a22r0C"'27&
              "&dC Para visualizar pagina siguiente pulse 'RETURN'
              "'27"&d@"
9630      ENTER 255,W6,R#
9632      PRINT '27"m"
9634      DOEND
9636      RETURN
```

&

```

9700 REM*****
9702 REM *
9704 REM      SUBROUTINA 3. (MOD 2.)      VOLCADO DE FICHERO TEMPO- *
9706 REM                                RAL 'TEMP' EN FICHERO DE *
9708 REM      EXPEDIENTES Y AMPLIACION DE ESTE SI ES NECESARIO *
9710 REM *
9712 REM*****
9714 IF N5+N>N6 THEN DO
9716   CONVERT G#[2,LEN(G#)] TO G
9718   G=G+1
9720   CONVERT G TO G1$
9722   G1$="G"+G1$
9724   N6=N6+1000
9726   CREATE X9,G1#+W#,N6,319
9728   ASSIGN G1#+W#,8,XB
9730   IF X9<>0 OR XB<>0 THEN 9716
9732   P6=8
9734 DOEND
9736 IF N>0 THEN DO
9738   N7=N5+N
9740   IF N5=1 THEN 9788
9742   READ #6,N5;Q,D5#[1],D5#[2],D5#[3]
9744   READ #7,N;Q,D7#[1],D7#[2],D7#[3]
9746   IF D7#[1]+D7#[2]+D7#[3]>D5#[1]+D5#[2]+D5#[3] THEN DO
9748     READ #7,N;Q,(FOR K1=1 TO 22,M#[K1]),T,(FOR K1=1 TO T,F3#[K1],&
      F4#[K1],F5#[K1])
9750     PRINT #P6,N5+N;Q,(FOR K1=1 TO 22,M#[K1]),T,(FOR K1=1 TO T,F3#&
      [K1],F4#[K1],"0")
9752     N=N-1
9754     IF N=0 THEN DO
9756       IF P6=8 THEN DO
9758         FOR K=N5 TO 2 STEP -1
9760         READ #6,K;Q,(FOR K1=1 TO 22,M#[K1]),T,(FOR K1=1 TO T,&
      F3#[K1],F4#[K1],F5#[K1])
9762         PRINT #8,K;Q,(FOR K1=1 TO 22,M#[K1]),T,(FOR K1=1 TO T,&

```

```

          F3#[K1 ],F4#[K1 ],F5#[K1 ])
9764         NEXT K
9766         DOEND
9768         PRINT #P6,1;N7,N6
9770         RETURN
9772         DOEND
9774         READ #7,N;Q,D7#[ 1 ],D7#[ 2 ],D7#[ 3 ]
9776         GOTO 9746
9778         DOEND
9780         ELSE DO
9782         READ #6,N5;Q,(FOR K1=1 TO 22,M#[K1 ]),T,(FOR K1=1 TO T,F3#[K1 ]&
          ,F4#[K1 ],F5#[K1 ])
9784         PRINT #P6,N5+N;Q,(FOR K1=1 TO 22,M#[K1 ]),T,(FOR K1=1 TO T,F3#[
          [K1 ],F4#[K1 ],F5#[K1 ])
9786         N5=N5-1
9788         IF N5=1 THEN DO
9790         FOR K=N TO 1 STEP -1
9792         READ #7,K;Q,(FOR K1=1 TO 22,M#[K1 ]),T,(FOR K1=1 TO T,F3#[&
          K1 ],F4#[K1 ],F5#[K1 ])
9794         PRINT #P6,K+1;Q,(FOR K1=1 TO 22,M#[K1 ]),T,(FOR K1=1 TO T,&
          F3#[K1 ],F4#[K1 ],"0")
9796         NEXT K
9798         PRINT #P6,1;N7,N6
9800         RETURN
9802         DOEND
9804         READ #6,N5;Q,D5#[ 1 ],D5#[ 2 ],D5#[ 3 ]
9806         GOTO 9746
9808         DOEND
9810         DOEND
9812         RETURN

```

>

```

9900 REM*****
9902 REM *
9904 REM      SUBROUTINA 4. (MOD. 5)      LISTADO DE ASIGNATURAS *
9906 REM                                POR CONVOCATORIA, CARRERA, *
9908 REM      PLAN Y CURSO QUE ESTAN SIN CALIFICAR. *
9910 REM *
9912 REM*****
9914 IF J=0 THEN DO
9916     J=1
9918     PRINT '27"&a1r4C"'27&
          "&dC Quedan por calificar las siguientes asignaturas: &
          "'27"&d0"
9920     PRINT '27"1"
9922 DOEND
9924 IF K=0 THEN DO
9926     K=1
9928     PRINT
9930     PRINT TAB(4);"Carrera: ";D7#[1];TAB(33);"Plan: ";D7#[2];TAB(55)&
          ;"Curso: ";D7#[3]
9932     PRINT LIN(1);TAB(1);"ASIGNATURA";TAB(59);"CONVOCATORIA"
9934     PRINT (FOR K3=1 TO 72,"=";)
9936 DOEND
9938 READ #1,Q+1;D6#[1];D6#[2],A1,B,(FOR K4=1 TO B,F2#[K4])
9940 FOR I2=1 TO I
9942     IF F2#[K2]=F3#[I2] THEN DO
9944         FOR I3=1 TO 3
9946             IF V[I2,I3]=A THEN 9962
9948         NEXT I3
9950         V[I2,A]=A
9952         GOTO 9960
9954     DOEND
9956 NEXT I2
9958 I=I+1,F3#[I]=F2#[K2],V[I,A]=A,V[I+1,1]=V[I+1,2]=V[I+1,3]=0
9960 PRINT TAB(1);F2#[K2];TAB(66);C4#[A,1;3]
9962 RETURN

```

>

BIBLIOGRAFIA

1. Pressman, Roger S. : "Software engineering: A practitioner's approach", Mc Graw-Hill, New York, 1982.
2. Shooman, Martin L. : "Software engineering: Design/Reliability/Management", Mc Graw-Hill, New York 1983.
3. Manual de uso del BASIC/3000, Hewlett Packard Inc., New York, 1978.