

---

# UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

---

FACULTAD DE INGENIERIA



## SOLUCION DE ALGORITMOS DE CONTROL DIGITAL POR COMPUTADORA

**T E S I S I N A**

QUE PARA OBTENER EL TITULO DE:  
INGENIERO MECANICO ELECTRICISTA  
P R E S E N T A N

OCTAVIO ALEJANDRO IRIS AGUILAR  
CELERINO PEREZ GUERRA

D I R E C T O R E S :

DR. ROBERTO CANALES RUIZ

DR. ANDRES BUZO DE LA PEÑA

MEXICO, D. F.

1982



Universidad Nacional  
Autónoma de México



**UNAM – Dirección General de Bibliotecas**  
**Tesis Digitales**  
**Restricciones de uso**

**DERECHOS RESERVADOS ©**  
**PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL**

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

INDICE

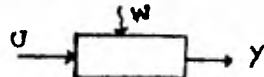
PROGRAMA NO.I ... SERIE EXPONENCIAL .....	1
PROGRAMA NO.II... VARIABLES DE ESTADO .....	5
PROGRAMA NO.III . RAICES DE POLINOMIO .....	7
PROGRAMA NO.IV .. GRAFICA .....	9
PROGRAMA NO.V ... VALORES CARACTERISTICOS .....	11
PROGRAMA NO.VI .. ACKERMANN .....	13
PROGRAMA NO.VII . ROOT LOCUS .....	16
LISTADO DEL PROGRAMA SERIE EXPONENCIAL .....	19
LISTADO DEL PROGRAMA VARIABLES DE ESTADO .....	24
LISTADO DEL PROGRAMA RAICES DE POLINOMIO .....	29
LISTADO DEL PROGRAMA GRAFICA .....	33
LISTADO DEL PROGRAMA VALORES CARACTERISTICOS .....	37
LISTADO DEL PROGRAMA ACKERMANN .....	43
LISTADO DEL PROGRAMA ROOT LOCUS .....	51
BIBLIOGRAFIA .....	62

PROGRAMA NO. I

1.- NOMBRE: "SERIE EXPONENCIAL".

2.- DESCRIPCION DEL PROBLEMA.

La representación de un sistema de ecuaciones diferenciales, con coeficientes constantes, lineal, continuo e invariante en el tiempo es:

$$\dot{X} = FX + GU + G_1W$$


The diagram shows a rectangular block representing a system. An arrow labeled 'U' enters the block from the left. An arrow labeled 'W' enters the block from the top. An arrow labeled 'Y' exits the block to the right.

Y lo denominamos sistema de variables de estado.

Donde: U: Entrada de control al sistema

W: Entrada perturbatoria

X: Vector de variables de estado

F: Matriz del sistema

G, G<sub>1</sub>: Vectores del sistema.

Podemos expresar la salida como:

$$Y = HX + JU$$

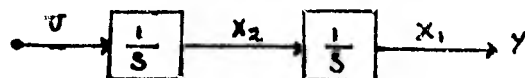
Donde: Y: Salida del sistema

H: Vector que determina la relación de Y con los estados

J: Constante de proporcionalidad entrada-salida.

En este programa se asume que G<sub>1</sub>, J es igual a 0, ya que en el parámetro J, en problemas de cálculo de control comúnmente es 0, y consideramos que no existen perturbaciones por lo cual G<sub>1</sub> es 0.

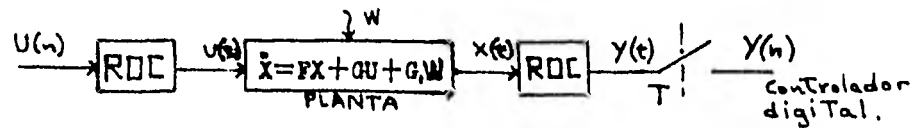
A manera de ejemplo tengamos el siguiente sistema:



cuya representación; el sistema de variables de estado:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} U \quad Y = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

En control digital, el sistema continuo se convierte, en un sistema muestreado como lo ilustra la siguiente figura



Donde: T: Periodo de muestreo

ROC: Retén de orden Cero

Observe que el controlador digital trabaja con ecuaciones de diferencias.

Para cerrar la malla es preciso resolver el sistema de variables de estado.

Donde la solución está dada por: \*

$$X_h(t) = \text{EXP}(F(t-t_0))X(t_0)$$

Donde: H: Se refiere a homogenea

$$\text{Donde: } \text{EXP}(F(t-t_0)) \triangleq I + F(t-t_0) + \frac{F^2(t-t_0)^2}{2!} + \frac{F^3(t-t_0)^3}{3!} + \dots \\ = \sum_{k=0}^{\infty} \frac{F^k(t-t_0)^k}{k!}$$

Y la solución particular:  $X_p(t) = \text{EXP}(F(t-t_0))B(t)$

$$\text{Donde: } B(t) = \int_{t_0}^t \text{EXP}(-F(t-\tau))GU(\tau)d\tau$$

$$\text{Llegando a: } X_p(t) = \int_{t_0}^t \text{EXP}(F(t-\tau))GU(\tau)d\tau$$

$$\text{Y como: } X(t) = X_h(t) + X_p(t)$$

$$X(t) = \text{EXP}(F(t-t_0))X(t_0) + \int_{t_0}^t \text{EXP}(F(t-\tau))GU(\tau)d\tau$$

Considerando el sistema muestreado, tenemos:

$$X(nT+T) = \text{EXP}(FT)X(nT) + \int_{nT}^{nT+T} \text{EXP}(F(nT+T-\tau))d\tau GU(\tau), \quad \bar{n} = nT+T-\tau$$

Definiendo  $FI = \text{EXP}(FT)$

$$GAMA = \int_0^T \text{EXP}(F\bar{n})d\bar{n}G$$

Reduciendose el sistema a:

$$X(n+1) = F IX(n) + GAMA U(n) \quad Y(n) = H X(n)$$

$$\text{Donde: } FI = \text{EXP}(FT) = I + FT + \frac{F^2 T^2}{2!} + \frac{F^3 T^3}{3!} + \dots$$

$$\text{Si: } FI = I + FT + PSI \quad PSI = I + \frac{FT}{2!} + \frac{F^2 T^2}{3!} + \dots$$

$$\text{Para evaluar GAMA: } GAMA = \sum_{k=0}^{\infty} \frac{F^k T^{k+1}}{(k+1)!} G \\ = \sum_{k=0}^{\infty} \frac{F^k T^k}{(k+1)!} TG \\ = PSITO$$

\* Consulte. Franklin & Powell. "Digital control of dynamic systems" Cap. VI

Ahora para evaluar la serie exponencial podemos considerar un número de términos estimado. Nos preguntamos que tan buena será nuestra estimación, para ello podríamos calcular el residuo.

Para obtener

$$\sum_{k=0}^{\infty} \frac{F^k T^k}{(k+1)!} = \lim_{N \rightarrow \infty} \sum_{k=0}^N \frac{F^k T^k}{(k+1)!} \quad *$$

(Sobre este punto trataremos con detalles en el siguiente programa "VARIABLES DE ESTADO").

### 3.- Descripción y uso del programa

(Suponemos que Ud. no conoce la Apple).

Este programa está diseñado para cualquier usuario.

a). Su computadora Apple está apagada.

Introduzca el disco de control digital en el drive 1 (cierre la puerta del drive)

b). Prenda la televisión y la Apple, y espere a que el drive sea "enganchado", esto sucederá cuando se apague el LED, en caso de que continúe por más de un minuto, oprima Crt Reset y oprima PR#6 "Return".

c). Corra el programa así:

RUN SERIE EXPONENCIAL "Return"

d). El programa tiene las instrucciones suficientes para continuar.

e). Si desea hacer correcciones, liste el programa. List A,B "Return", donde A, es el principio de listado y B el final.

f). La lógica del programa es la siguiente y el listado se anexa al final con su nombre respectivo.

#### LOGICA DEL PROGRAMA

- 1.- Selección de período de muestreo, obtención de las matrices F y G.
- 2.- Formación de la matriz identidad.
- 3.- A la matriz PSI se le asigna la matriz identidad.
- 4.- Selección del número de términos TE.
- 5.- Proceso iterativo desde TE hasta uno en forma de creciente-

6.- A la matriz PSI le asignamos  $I+FP\text{SI}$ .

7.- Regreso iterativo.

8.- A la matriz Gama le asignamos  $PSITG$ .

9.- A la matriz FI le asignamos  $I+FP\text{SI}$ .

Nota: Después de listado el programa anexamos un ejemplo de aplicación.

PROGRAMA NO. II

1.- NOMBRE: "VARIABLES DE ESTADO".

2.- DESCRIPCION DEL PROBLEMA.

El presente programa es semejante al anterior con la diferencia que aquí buscamos un número de términos automáticamente para la evaluación de la serie exponencial a un valor satisfactorio.

Para ello utilizamos el siguiente algoritmo:

Cuando FT es grande, se toman bastantes términos de la serie exponencial, para obtener un valor apropiado, entonces buscamos que, término a término  $\frac{(FT)^N}{N!}$  se haga pequeño, una vez hecho esto vamos doblando la serie exponencial hasta obtener el valor de la serie, sin escalamiento usando la fórmula siguiente:

$$(EXP(FT))^2 = EXP(2T)$$

Por ejemplo podremos calcular la serie en T/2 y luego elevar el resultado al cuadrado, en caso de que T/2 vuelve a ser muy grande, calculamos la serie para T/4 y elevamos a la cuarta potencia, en caso de que vuelva a ser muy grande, se vuelve a hacer el mismo procedimiento y así sucesivamente hasta que encontremos un T/2<sup>k</sup>, que sea lo suficientemente pequeño, para esto utilizamos el siguiente criterio.

Sea  $FT/2^k < 1$  donde las series de FT/2<sup>k</sup>, seguramente convergerán.

La regla para seleccionar K es:

$$2^k > ||FT||$$

Donde:  $||FT|| \triangleq \max_j \sum_{i=1}^n |F_{ij}| T$   
 $\log_2 K > \log_2 ||FT||$

De donde:  $K = \max((\log_2 ||FT||, 0))$

Ahora ya podremos duplicar PSI con la fórmula:

$$PSI(2T) = (I + \frac{T_1}{2} PSI(T)) PSI(T)$$

Para  $T_1 = T/2^k$



3.- DESCRIPCION Y USO DEL PROGRAMA.

- a). Siga las instrucciones a y b del programa anterior.
- b). Corra el programa así: RUN VARIABLES DE ESTADO "Return"
- c). El programa tiene las instrucciones que se requieren para continuar.
- d). La lógica del programa es la siguiente y el listado lo anexamos al final --  
con el nombre del programa.

- 1.- Seleccione F y T .
- 2.- Comentario: Obtención de la norma de FT.
- 2a. A V le asignamos  $\max_j (\sum_i |F_{ij}|) \times T$ .
- 2b. K el mas pequeño no negativo entero mayor que  $\log_2 V$ .
- 3.- Obtención de  $PSI(T/2^K)$ .
- 4.-  $T1 = T/2^K$
- 5.- Formación de la matriz identidad.
- 6.- A PSI le asignamos la identidad.
- 7.-  $j = 1$ .
- 8.- If  $j = 1$  GOTO 12.
- 9.-  $PSI = I + \frac{FT}{j} PSI$ .
10. A j le asignamos  $j-1$ .
11. GOTO 8.
12. Comentario: Duplicación de PSI, K veces.
13. If  $K = 0$  STOP.
14. A PSI le asignamos  $(I + \frac{FT}{2^{K+1}} PSI) PSI$ .
15. A K le asignamos  $K-1$ .
16. GOTO 13.

PROGRAMA NO. III

- 1.- NOMBRE: "RAICES DE POLINOMIO".
- 2.- DESCRIPCION DEL PROBLEMA.

La solución de las raíces de un polinomio de  $n$ ésimo grado, ha sido ampliamente tratado y existen muchos algoritmos; retomando uno de ellos; el método iterativo Lin-Bairstow consiste en el cálculo de las raíces reales o complejas de un polinomio de coeficientes reales. El método está basado en la sucesiva extracción de factores cuadráticos del polinomio original de grado  $N$  y de los subsecuentes polinomios de grado  $N-2m$  ( $m=1,2,\dots$ ). Cada factor cuadrático es determinado por un proceso iterativo de corrección diferencial. El principio es:

$$Dn: Pn(x) = F(x)Pn-2(x) + Rx + S$$

Observamos que si  $R$  y  $S$  son Cero, efectivamente  $F(x)$  será un polinomio cuadrático del polinomio reducido  $Pn-2(x)$ . Para la obtención de  $R$  y  $S$  es suficiente dar originalmente dos valores iniciales y posteriormente irlos reduciendo por medio del método de corrección diferencial de Newton; (consultar la nota al final de esta página).

- 3.- DESCRIPCION Y USO DEL PROGRAMA:

Como este programa es auxiliar a los demás programas de control digital no están protegidos contra descuidos. Por lo que le recomendamos tenga cuidado al correr el programa y coloque los datos coherentes.

- a). Para correr el programa siga las instrucciones a y b del programa número I.
- b). Corra el programa así:

RUN RAICES DE POLINOMIO "Return"

- c). Los datos que le piden son grado del polinomio, aproximación requerida, (Si usted coloca una aproximación que salga de las capacidades numéricas de la máquina seguramente nunca va a obtener un resultado y no convergerá el programa a la solución), valores iniciales de aproximación a la raíz  $R$  y  $S$ ; que son los coeficientes de la raíz cuadrática a obtener, después le piden los coeficientes, término a término del polinomio, indicándole en que formato debe hacerlo.

d). La lógica del programa es la siguiente:

- 1.- Lectura del grado del polinomio, del error permitido, de los valores iniciales del factor cuadrático, es decir las raíces originales.
- 2.- Primero obtenemos la división del polinomio enésimo entre el polinomio cuadrático dado:
- 3.- De donde obtendremos un error  $Rx + S$ , el cual es necesario eliminar, o sea tenemos que hacer  $R$  y  $S$  Cero, por lo que utilizamos el método de Newton para obtener las raíces correctas del factor cuadrático.
- 4.- Y así sucesivamente hasta tener el polinomio enésimo factorizado en polinomios cuadráticos.

PROGRAMA NO. IV

1.- NOMBRE: "GRAFICA".

2.- DESCRIPCION DEL PROBLEMA.

Este algoritmo está dividido en dos partes, una de ellas calcula; de un sistema de variables de estado la respuesta a escalón, y la otra; de una función de transferencia obtenemos la forma canónica de sus variables de estado de la función de transferencia y obtenemos la respuesta a escalón.

Dado el sistema de variables de estado de la siguiente forma:

$$Y(n) = HX(n)$$

$$X(n+1) = FI X(n) + GAMA U(n)$$

O la función de transferencia:

$$H(Z) = \frac{B(1) + B(2) Z^{-1} + \dots + B(n)}{Z^n + A(1) Z^{n-1} + A(2) Z^{n-2} + \dots + A(n)}$$

Que en su forma canónica es:

$$FI = \begin{bmatrix} -A(1) & -A(2) & \dots & -A(n) \\ 1 & 0 & \dots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 1 & & 0 \end{bmatrix} \quad GAMA = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad H = [B(1) \quad B(2) \quad \dots \quad B(n)]$$

3.- DESCRIPCION Y USO DEL PROGRAMA:

a).Siga las instrucciones a y b del programa número 1 "SERIE EXPONENCIAL".

b).Cerra el programa así:

RUN "GRAFICA"

c).El programa tiene las instrucciones suficientes para continuar.

d).La lógica del programa es:

1.- Selección del tipo del algoritmo que se usará.

2.- Transforma la función de transferencia a un sistema de variables de estado.

En caso de que se inicie, con una función de transferencia, se transformará a su forma canónica, en variables de estado.

3.- Hace el cálculo partiendo desde Cero (el vector  $X$ ), resolviendo para la próxima  $X$  y calculando  $Y$  (la salida del sistema) en el valor anterior de la  $X$  actual (aunque podría calcularse la  $Y$  para un tiempo igual a la actual  $X$ ).

4.- Se grafican 46 puntos en toda la pantalla en la cual tenemos una resolución de 276 puntos, por lo tanto cada punto está espaciado 6 puntos.

5.- Se calcula una escala en base a la  $Y$  máxima.

6.- En caso de la existencia de una  $Y$  negativa, se coloca el eje del tiempo a la mitad de la pantalla.

7.- Grafíquese la función escalón (en caso de que la escala sea bastante grande dibuje la función escalón en cero).

PROGRAMA NO. V

- 1.- NOMBRE : " VALORES CARACTERISTICOS "
- 2.- DESCRIPCION DEL PROBLEMA.

El teorema de Cayley-Hamilton establece que toda matriz cuadrada satisface su ecuación característica.

Si se tiene una matriz A, nxn, cuyo polinomio característico es :

$$f(\lambda) = \det(\lambda I - A) = \lambda^n + c_{n-1}\lambda^{n-1} + \dots + c_1\lambda + c_0 \quad \dots 1$$

entonces A satisface la ecuación:

$$A^n + c_{n-1}A^{n-1} + \dots + c_1A + c_0I = 0 \quad \dots 2$$

La potencia n-ésima de cualquier matriz A, nxn, puede ser expresada como una combinación lineal de las potencias inferiores I, A, A<sup>2</sup>, ..., A<sup>n-1</sup>. Puede deducirse de esto que cada una de las potencias superiores A<sup>n+1</sup>, A<sup>n+2</sup>, ... pueden expresarse como combinación lineal de I, A, A<sup>2</sup>, ..., A<sup>n-1</sup>. (1)

Multiplicando la ecuación 2 por un vector V, nx1, cualquiera diferente de cero obtenemos:

$$A^n V + c_{n-1}A^{n-1}V + \dots + c_1AV + c_0V = 0$$

pasando la potencia superior al otro lado de la igualdad se llega a :

$$c_{n-1}A^{n-1}V + \dots + c_1AV + c_0V = -A^n V$$

arreglando en forma matricial :

$$\begin{bmatrix} A^{n-1}V & A^{n-2}V & \dots & AV & V \end{bmatrix} \begin{bmatrix} c_{n-1} \\ \vdots \\ c_1 \\ c_0 \end{bmatrix} = -A^n V \quad \dots 3$$

De donde pueden calcularse, mediante un programa, los coeficientes  $c_{n-1}, c_{n-2}, \dots, c_1, c_0$  y llevandolos a la ecuación 1, obtener los valores característicos

(1) Véase. Calculus, Tom II. Apostol, Vol.2

3.- DESCRIPCIÓN Y USO DEL PROGRAMA

a). Siga las instrucciones a y b del programa I.

b). Corra el programa así:

RUN VALORES CARACTERISTICOS (oprime Return)

c). Al final se tiene el listado del programa y un ejemplo de aplicación .

e). Lógica del programa.

1.- Entrada de datos : orden de la matriz , elementos de  $\pm$  la matriz A.

2.- Arreglo matricial de acuerdo a la expresión 3.

3.- Transposición y triangularización.

4.- Obtención de los coeficientes  $U_n$ 's.

5.- Cálculo de los valores característicos.

PROGRAMA NO. VI

1.- NOMBRE : " ACHEMERA "

2.- DESCRIPCION DEL PROBLEMA.

El calculo de la ley de control es el primer paso para el dise-  
no por el metodo de espacio de estados, suponiendo que todos los -  
estados son aprovechables.

El paso subsecuente es el diseo de un estimador. la combina-  
ción de la ley de control y el estimador forman lo que se conoce e-  
como controlador, donde el calculo de la ley de control es basada  
sobre los estados estimados , además de el estado actual.

La ley de control es la realimentación de una combinación lineal  
de todos los estados .Expresado matemáticamente seria:

$$u = -K \chi = - \begin{bmatrix} k_1, k_2, \dots \end{bmatrix} \begin{bmatrix} \chi_1 \\ \chi_2 \\ \vdots \end{bmatrix} \text{ --- 1}$$

Utilizando la ecuación diferencial en su forma estandar

$$\chi(n+1) = \phi \chi(n) + \Gamma u(n) + \Gamma_1 \omega(n) \text{ --- 2}$$

es posible sustituir el valor de la ecuación 1 , y se tiene:

$$\chi(n+1) = \phi \chi(n) - \Gamma K \chi(n) + \Gamma_1 \omega \text{ --- 3}$$

Si suponemos  $\omega = \omega_0 \delta_0$  (un pulso en cero) , la transformada zeta  
de 3 sera:

$$(zI - \phi - \Gamma K) X(z) = \Gamma_1 \omega_0$$

en donde el  $\det |zI - \phi + \Gamma K| = 0 \text{ --- 4}$

es la ecuación característica del sistema controlado(lazo cerrado).

Por lo tanto el diseo de la ley de control consiste en esco-  
ger los elementos de K de tal forma que las raices de la ecuación  
4 se encuentren en el lugar deseado.

Dada la localización de las raices deseadas se tiene:

$$z_i = \beta_1, \beta_2, \beta_3, \dots$$



donde la ecuación característica deseada de control sera:

$$d_c(z) = (z - \beta_1)(z - \beta_2)(z - \beta_3) \dots = 0 \quad \text{--- 5}$$

Las  $\beta$ 's son obtenidas por igualación de coeficientes con las ecuaciones 4 y 5 forzando el polinomio característico del sistema a ser idéntico a la expresión 5.

Sin embargo para poder utilizar un programa de computadora es necesario arreglar las expresiones a una formula más conveniente. Esta formula es proporcionada por Ackermann de la siguiente manera:

$$K = [0 \dots 0 \ 1] [\Gamma \ \phi \Gamma \dots \phi^{n-1} \Gamma]^{-1} d_c(z) \quad \text{--- 6}$$

donde  $\mathcal{L} = [\Gamma \ \phi \Gamma \dots \phi^{n-1} \Gamma]$

es la matriz de controlabilidad.

n = orden del sistema o numero de estados.

Si sustituimos  $\phi$  por z en  $d_c(z)$  se llega a:

$$d_c(\phi) = \phi^n - \alpha_1 \phi^{n-1} - \dots - \alpha_n I$$

donde las  $\alpha$ 's son los coeficientes de la ecuación característica -- deseada, esto es :

$$d_c(z) = |zI - \phi + \Gamma K| = z^n - \alpha_1 z^{n-1} - \dots - \alpha_n$$

El programa de ACKERMANN resuelve la expresión 6 para el calculo de las  $\alpha$ 's.

### 3.- DESCRIPCIÓN Y USO DEL PROGRAMA.

a). Sigue las instrucciones a y b del programa I.

b). Corra el programa así :

RUN ACKERMANN (oprime Return)

c). Al final se tiene un programa de aplicación y el listado, para su consulta.

d). Lógica del programa .

treo T;  $\Gamma$  - Lectura de datos: orden del sistema N; tiempo de muestreo

- 2.- Formación de la matriz identidad I.
- 3.- Colocación de indicadores necesarios, al programa.
- 4.- Si el número de polos dado es mayor que N, entonces va al paso 1.
- 5.- Lectura de polos (parte real A, parte imaginaria B).
- 6.- El polo es complejo ? si, paso 8.
- 7.- Calcula  $A1 = e^{AT}$  y  $\Delta ALPHA = ALPHA \times (\phi - A1 \times I)$ .
- 7.a Regresa al paso 4.
- 8.- Calcula  $A1 = -2 e^{AT} \cos BT$  ;  $A2 = e^{2AT}$   
 $\Delta ALPHA = ALPHA \times (\phi \times \phi + A1 \phi + A2 I)$ .
- 8a.- Regresa al paso 4.
- 9.- Construye la matriz de controlabilidad,  $\mathcal{C}$ .
- 10.- Presenta en la pantalla los datos, y los cálculos realizados hasta el momento.
- 11.- Calcula las K's. e imprime sus valores en la pantalla.

PROGRAMA NO. VII

- 1.- NOMBRE DEL PROGRAMA: "ROOT LOCUS "
- 2.- DESCRIPCION DEL PROBLEMA

El lugar de las raíces es el lugar geométrico de los puntos de donde las raíces de una ecuación característica pueden localizarse al variar un parametro real desde cero hasta valores muy grandes.

Generalmente el parámetro que varia es la ganancia, pero es posible utilizar cualquier otra variable de la función de transferencia de lazo abierto.

La ecuación característica de un sistema de lazo sencillo en el dominio zeta es :

$$1 + D(z) G(z) = 0$$

ecuación que es semejante al del dominio s.

El mecanismo de trazado de las raíces en el plano zeta es el mismo que en el plano s, excepto que la localización de polos tiene diferente significado cuando es interpretada la estabilidad del sistema y la respuesta dinámica.

Es conveniente contar con un programa que trace el lugar geométrico de las raíces ya que permite el diseño de controladores por el método de técnicas transformadas.

En el diseño de sistemas de lazo cerrado se desea ajustar los polos y ceros de lazo abierto, de manera de que los polos y ceros de lazo cerrado se situen en las posiciones deseadas en el plano s.

- 3.- DESCRIPCION Y USO DEL PROGRAMA.

- a). Siga las instrucciones a y b del programa I.
- b). Corra el programa así :

RUN ROOT LOCUS ( oprima Return).

c). El programa está hecho para aceptar un rango de ganancia positivo, con un límite inferior KI y un límite superior KS, donde  $KS - KI$  es siempre positivo. Sin embargo puede aceptar un rango negativo de ganancia realizando las siguientes operaciones:

1.- Carga el programa con :

**LOAD ROOT LOCUS** (oprime Return).

2.- Borre la instrucción 420 así:

420 (oprime Return). Pero:

3.- Debe cuidarse que KI sea mayor que KS

Ejemplo :  $KI = -3$  ;  $KS = -6$

d). Puede dibujar en el plano S o Z. En el plano S aparece el círculo unitario pero debe tomarse solo como referencia .

e). En el programa aparecen una serie de indicaciones que guían al usuario.

f). Lógica del programa.

1.- Entrada de datos: grado del numerador PM; grado del denominador PN ; coeficientes del numerador BM ; coeficientes del denominador AN ; se dibujará en el plano S o Z , si es en Z dar el período de muestreo T ; rango de la ganancia KI, KS y número de -- puntos a evaluar NE.

2.- Se calcula el  $INC = (KS - KI) / NE$

3.- Pregunta si se ha llegado a KS , si: va a 10.

4.- Arreglo del polinomio para poder calcular las raíces.

5.- Se calculan las raíces .

6.- Se imprimen en la pantalla.

7.- Se guardan en un arreglo.

8.- Se incrementa el valor de la ganancia.

9.- Se regresa al paso 3.

10.- Se determina la raíz más grande para efecto de escalamiento.

- 11.- Se traza en la pantalla el lugar geométrico de las raíces .
- 12.- Se quiere cambiar de escala ? no: paso 14
- 13.- Pregunta por el factor de escalamiento FE y regresa a 11.
- 14.- Pregunta por otro rango de ganancia NO ; 16.
- 15.- Regresa al paso 1.
- 16.- Fin .

PROGRAMA # I

SERIE EXPONENCIAL

```

3?""
3LIST
5 HOME
10 PRINT " SOLUCION DE UN SISTEMA DE VARIABLES DE ESTADO
"
15 PRINT " SELECCIONANDO EL NUMERO DE TERMINOS EN LA OBTENCION DE LA MATRIZ EXPONENCIAL
"
20 PRINT " AL INTERRUPIRSE UN LISTADO CON EL SIMBOLO ( ? )
OPRIMA 'RETURN'
"
25 REM INTRODUCCION DE DATOS
30 PRINT " CUAL ES EL ORDEN DEL SISTEMA ?
"
35 INPUT N
40 IF N <= 0 THEN 395
45 IF N < > INT (N) THEN 405
50 N = INT (N)
55 IF N > 20 THEN 420
60 PRINT " CUAL ES EL PERIODO DE MUESTREO ?
"
65 INPUT T
70 IF T <= 0 THEN 430
75 DIM F(N,N),G(N),FI(N,N),GA(N),OP(N,N),M(N,N),N(N,N),P(N,N),C(N,N),I(N,N)
80 PRINT " DADO EL SISTEMA DE LA FORMA : X(K+1)=FX(K)+GU
"
85 FOR I = 1 TO N
90 PRINT " INTRODUZCA EL RENGLON "I" DE LA "MATRIZ F
"
95 FOR J = 1 TO N
100 INPUT F(I,J)
105 NEXT J,I
110 PRINT " INTRODUZCA EL VECTOR G
"
115 FOR I = 1 TO N
120 INPUT G(I)
125 NEXT I
130 HOME
135 PRINT " SUS DATOS SON :
"
140 REM SE APLICAN CONTADORES PARA PARAR LA IMPRESION
145 FOR I = 1 TO N
150 FOR J = 1 TO N
155 CZ = CZ + 1: IF CZ = 40 THEN CZ = 0: INPUT B#
160 PRINT "F("I;"J;")= "F(I,J),
165 NEXT J,I
170 FOR I = 1 TO N
175 CZ = CZ + 1: IF CZ = 40 THEN CZ = 0: INPUT B#
180 PRINT "G("I;")= "G(I),
185 NEXT I
190 REM CORRECCIONES Y PROTECCIONES PARA SER USADO POR CUALQUIER USUARIO
195 PRINT " DESEA HACER ALGUNA CORRECCION (S,N)
"
200 INPUT B#
205 IF B# = "N" THEN 450
210 IF B# = "S" THEN 225
215 PRINT " SOLAMENTE SE ACEPTA (S) O (N)
"
" VUELVA A INDICAR LO QUE SE PIDE
"
220 GOTO 200
225 PRINT " INDIQUE EN DONDE HARA SU CORRECCION :
EN : ORDEN DEL SISTEMA ( N ) , PERIODO DE MUESTREO ( T ) , ( F ) O ( G )
"
230 INPUT B#
235 IF B# = "N" THEN RUN
240 IF B# = "T" THEN 305
245 IF B# = "F" THEN 335
250 IF B# = "G" THEN 275
255 PRINT " NO SE RECONOCE TAL DATO
"
" INTRODUZCA SOLAMENTE UN DATO N, T, F, G
"
260 GOTO 220

```

```

265 PRINT "           ESE VALOR EXCEDE EL ORDEN DADO           " 21
      DE OTRA VEZ LOS DATOS
270 GOTO 335
275 PRINT "           DE EL RENGLON , DESPUES EL NUEVO DATO "
280 INPUT I
285 IF (I > N) OR (I <= 0) THEN PRINT "           NO EXISTE ESE VALOR DELO NUEVAMENTE
      " : GOTO 280
290 PRINT "G(I)= ";
295 INPUT G(I)
300 GOTO 365
305 PRINT "           DE EL NUEVO TIEMPO DE MUESTREO           "
310 INPUT T
315 IF T <= 0 THEN PRINT "           EL TIEMPO DE MUESTREO NO PUEDE SER CERO SI SE DA N
      NEGATIVO CONTINUA EL PROCESO CONSIDERANDOLO POSITIVO "
320 T = - T
325 IF T = 0 THEN PRINT " DE OTRO TIEMPO DIFERENTE DE CERO " : GOTO 310
330 GOTO 365
335 PRINT "           INTRODUCZA PRIMERO EL RENGLON LUEGO LA COLUMNA , DESPUES EL NUEVO DAT
      O
340 INPUT I,J
345 IF (I > N) OR (J > N) THEN 265
350 IF (I <= 0) OR (J <= 0) THEN PRINT "           VALORES ILOGICOS VUELVA A DARLOS
      " : GOTO 340
355 PRINT "F(I; ,J,)= ";
360 INPUT F(I,J)
365 PRINT "           ALGUN OTRO CAMBIO ? ( S , N )           "
370 INPUT B$
375 IF B$ = "S" THEN 235
380 IF B$ = "N" THEN 450
385 PRINT "           UNICAMENTE OPRIMA (S) O (N)           "
390 GOTO 370
395 PRINT "           NO ACEPTAMOS TAL VALOR
      "
      VUELVA A DAR EL ORDEN
400 GOTO 35
405 PRINT "           CONTINUA EL PROCESO CONSIDERANDO SOLO LA PARTE ENTERA
      "
410 N = INT (N)
415 GOTO 40
420 PRINT "           ESTA COMPUTADORA NO TIENE TAL CAPACIDAD
      "
      NUEVAMENTE DE EL ORDEN
425 GOTO 35
430 PRINT "           EL TIEMPO DE MUESTREO NO PUEDE SER CERO NI NEGATIVO
      "
435 GOTO 65
440 REM OBTENCION DE LA NORMA PARA CONSIDERAR UN NUMERO RAZONABLE DE TERMINOS
445 REM COMENZAMOS EL PROGRAMA
450 PRINT "           INDIQUE CUANTOS TERMINOS DESEA
      TERMINO:0 1 2
      F1=1+F*T+(F*T)^2/2!+(F*T)^2/3!+...
455 INPUT TE
460 IF F(1,1) < 0 AND N = 1 THEN I0 = 1.F(1,1) = - F(1,1)
465 IF TE < 0 THEN PRINT " CONTINUA EL PROCESO CONSIDERANDO EL NUMERO DE TERMINOS POSITIVOS " : TE = - TE
470 IF TE = 0 THEN 635
475 IF TE = 1 THEN 795
480 REM COMIENZA EL PROGRAMA
485 H = 0
490 FOR A = TE TO 2 STEP - 1
495 D1 = T / (TE - (H - 1))
500 H = H + 1
505 FOR I = 1 TO N
510 FOR J = 1 TO N
515 IF I = J THEN I(I,J) = 1
520 OP(I,J) = I(I,J) + D1 + F(I,J)
525 NEXT J,I
530 IF H > 2 THEN GOTO 595
535 IF H = 2 THEN GOTO 565
540 FOR I = 1 TO N

```



```

545 FOR J = 1 TO N
550 M(I,J) = OP(I,J)
555 NEXT J,I
560 GOTO 495
565 FOR I = 1 TO N
570 FOR J = 1 TO N
575 M(I,J) = OP(I,J) - I(I,J)
580 NEXT J,I
585 GOSUB 765
590 GOTO 630
595 FOR I = 1 TO N
600 FOR J = 1 TO N
605 M(I,J) = OP(I,J) - I(I,J)
610 N(I,J) = P(I,J) + I(I,J)
615 P(I,J) = 0
620 NEXT J,I
625 GOSUB 765
630 NEXT I
635 FOR I = 1 TO N
640 FOR J = 1 TO N
645 IF I = J THEN I(I,J) = 1
650 M(I,J) = F(I,J) * T
655 N(I,J) = P(I,J) + I(I,J)
660 P(I,J) = 0
665 NEXT J,I
670 GOSUB 765
675 PRINT " ESTA MATRIZ ES FI "
680 FOR I = 1 TO N
685 FOR J = 1 TO N
690 FI(I,J) = I(I,J) + P(I,J)
695 IF I8 = 1 THEN FI(I,1) = 1 / FI(I,1)
700 PRINT "FI("I","J")= "FI(I,J)
705 NEXT J,I
710 IF I8 = 1 THEN 755
715 PRINT " ESTA MATRIZ ES GAMMA "
720 FOR I = 1 TO N
725 FOR J = 1 TO N
730 C(I,J) = N(I,J) * T
735 GA(I) = C(I,J) + G(J) + GA(I)
740 NEXT J
745 PRINT "GAMMA("I")= "GA(I)
750 NEXT I
755 PRINT "
TA DEL SISTEMA V(Z)/U(Z) UTILICE LA FORMULA
NOTA : PARA OBTENER LA FUNCION DE TRANSFERENCIA EN EL DOMINIO DE ZE
H*(Z*I-FI)^-1*GAMA CONSULTE FRANKLIN & POWELL PAGINAS 131,137 "
760 END
765 FOR I = 1 TO N
770 FOR L = 1 TO N
775 FOR J = 1 TO N
780 P(I,L) = P(I,L) + M(I,J) * N(J,L)
785 NEXT J,L,I
790 RETURN
795 FOR I = 1 TO N
800 FOR J = 1 TO N
805 IF I = J THEN I(I,J) = 1
810 P(I,J) = F(I,J) * T / 2
815 NEXT J,I
820 GOTO 635
JPR#0

```

13" EJEMPLO DE APLICACION"  
EJEMPLO DE APLICACION

23

IKUN

SOLUCION DE UN SISTEMA DE VARIABLES DE ESTADO

SELECCIONANDO EL NUMERO DE TERMINOS EN LA OBTENCION DE LA MATRIZ

EXPONENCIAL

AL INTERRUPIRSE UN LISTADO CON EL SIMBOLO ( ? )

O P P I Y

R E T U R N

CUAL ES EL ORDEN DEL SISTEMA ?

?2

CUAL ES EL PERIODO DE MUESTREO ?

0.001

DADO EL SISTEMA DE LA FORMA

$X(K+1) = FX(K) + GU$

INTRODUZCA EL RENGLON 1 DE LA MATRIZ F

?1

?2

INTRODUZCA EL RENGLON 2 DE LA MATRIZ F

?1

?2

INTRODUZCA EL VECTOR G

?1

?1

S U S D A T O S S O N

F(1,1)= 1    F(1,2)= 2    F(2,1)= 1    F(2,2)= 3    G(1)= 1    G(2)= 1

DESEA HACER ALGUNA CORRECCION (S/N)

?N

INDIQUE CUANTOS TERMINOS DESEA

FI=I+F\*1

F\*1)^2/2!+(F\*1)^3/3!+

TERMINO:0

1

2

?6

ESTA MATRIZ ES F1

F1(1,1)= 1.0010015

F1(1,2)= 2.004005E-02

F1(2,1)= 1.0020025E-02

F1(2,2)= 1.00300551

ESTA MATRIZ ES GAMA

GAMA(1)= 1.00150184E-03

GAMA(2)= 1.0020025E-03

NOTA : PARA OBTENER LA FUNCION DE TRANSFERENCIA EN EL DOMINIO DE ZETA DEL SISTEMA Y

X(Z) UTILICE LA FORMULA  $H(Z) = (Z-1)^{-1} * GAMA$  CONSULTE FRANKLIN & POWELL PAGINAS 131, 137

PROGRAMA # II

VARIABLES DE ESTADO

```

5 HOME
10 PRINT " SOLUCION DE UN SISTEMA DE VARIABLES DE ESTADO
15 PRINT " ( UTILIZANDO TIEMPO DE ESCALAMIENTO AUTOMATICO PARA LA OBTENCION DE LA MATRIZ EXPONENCIAL )
20 PRINT " AL INTERRUPIRSE UN LISTADO CON EL SIMBOLO ( ? )
   O P A I M A ' P E T U R N "
25 REM INTRODUCCION DE DATOS
30 PRINT "          CUAL ES EL ORDEN DEL SISTEMA ?
35 INPUT N
40 IF N <= 0 THEN 400
45 IF N < 20 THEN INT (N) THEN 410
50 N = INT (N)
55 IF N > 20 THEN 425
60 PRINT "          CUAL ES EL PERIODO DE MUESTREO ?
65 INPUT T
70 IF T <= 0 THEN 425
75 DIM F(N,N),G(N),F1(N,N),G1(N),OP(N,N),M(N,N),N1(N,N),P(N,N),C(N,N),I(N,N),V(N)
80 PRINT " DADO EL SISTEMA DE LA FORMA :          X(K+1)=FX(K)+GU

85 FOR I = 1 TO N
90 PRINT " INTRODUZA EL RENGLON "I" DE LA "MATRIZ F
95 FOR J = 1 TO N
100 INPUT F(I,J)
105 NEXT J,I
110 PRINT "          INTRODUCZA EL VECTOR G

115 FOR I = 1 TO N
120 INPUT G(I)
125 NEXT I
130 HOME
135 PRINT " S U S D A T O S E O N "
140 REM SE APLICAN CONTADORES PARA PARAR LA IMPRESION
145 FOR I = 1 TO N
150 FOR J = 1 TO N
155 C2 = C2 + 1: IF C2 = 40 THEN C2 = 0: INPUT B$
160 PRINT "F("I,";J;")= "F(I,J).
165 NEXT J,I
170 FOR I = 1 TO N
175 C2 = C2 + 1: IF C2 = 40 THEN C2 = 0: INPUT B$
180 PRINT "G("I;")= "G(I).
185 NEXT I
190 INPUT B$
195 REM CORRECCIONES Y PROTECCIONES PARA SER USADO POR CUALQUIER USUARIO
200 PRINT "          DESEA HACER ALGUNA CORRECCION (S/N)
205 INPUT B$
210 IF B$ = "N" THEN 155
215 IF B$ = "S" THEN 230
220 PRINT "          SOLAMENTE SE ACEPTA (S) O (N)
          VUELVA A INDICAR LO QUE SE PIDE
225 GOTO 205
230 PRINT "          INDIQUE EN DONDE HARA SU CORRECCION :
          EN ORDEN DEL SISTEMA ( N ) , PERIODO DE MUESTREO ( T ) , ( F ) O ( G )
235 INPUT B$
240 IF B$ = "N" THEN 155
245 IF B$ = "T" THEN 105
250 IF B$ = "F" THEN 325
255 IF B$ = "G" THEN 280
260 PRINT "          NO SE RECONOCE TAL DATO
          INTRODUCZA SOLAMENTE UN DATO N, T, F, G
265 GOTO 235
270 PRINT "          ESE VALOR TIENE EL ORDEN DADO
          DE OTRA VEZ LOS DATOS
275 GOTO 235
280 PRINT "          DE EL RENGLON DESPUES EL NUEVO DATO "
285 INPUT I
290 IF I < 1 OR I > N THEN PRINT "          NO EXISTE ESE VALOR DELO NUEVAMENTE
          GOTO 235

```

```

290 PRINT "G. I" = "
295 INPUT G(I)
300 GOTO 365
305 PRINT "
DE EL NUEVO TIEMPO DE MUESTREO
310 INPUT T
315 IF T <= 0 THEN PRINT "
EL TIEMPO DE MUESTREO NO PUEDE SER CERO SI SE DA UN
NEGATIVO CONTINUA EL PROCESO CONSIDERANDOLO POSITIVO
320 T = - T
325 IF T = 0 THEN PRINT " DE OTRO TIEMPO DIFERENTE DE CERO " : GOTO 310
330 GOTO 365
335 PRINT "
INTRODUZCA PRIMERO EL RENGLON LUEGO LA COLUMNA , DESPUES EL NUEVO DAT
0
340 INPUT I, J
345 IF (I > N) OR (J > N) THEN 270
350 IF (I <= 0) OR (J <= 0) THEN PRINT "
VALORES ILOGICOS VUELVA A DARLOS
" : GOTO 340
355 PRINT "F("I"; "J; ") = "
360 INPUT F(I, J)
365 PRINT "
ALGUN OTRO CAMBIO ? ( S / N )
370 INPUT B$
375 IF B$ = "S" THEN 230
380 IF B$ = "N" THEN 455
385 PRINT "
UNICAMENTE OPRIMA (S) O (N)
390 GOTO 370
400 PRINT "
NO ACEPTAMOS TAL VALOR
VUELVA A DAR EL ORDEN
405 GOTO 35
410 PRINT "
CONTINUA EL PROCESO CONSIDERANDO SOLO LA PARTE ENTERA
415 N = INT (N)
420 GOTO 10
425 PRINT "
ESTA COMPUTADORA NO TIENE TAL CAPACIDAD
NUEVAMENTE DE EL ORDEN
430 GOTO 35
435 PRINT "
EL TIEMPO DE MUESTREO NO PUEDE SER CERO NI NEGATIVO
440 GOTO 65
445 REM OBTENCION DE LA NORMA PARA CONSIDERAR UN NUMERO RAZONABLE DE TERMINOS
450 REM COMENZAMOS EL PROGRAMA
455 IF N = 1 THEN Y = F(1,1) * T : I(1,1) = 1 : FI(1,1) = 1 : GOTO 515
460 FOR J = 1 TO N
465 V(J) = 0
470 FOR I = 1 TO N
475 IF J = I THEN I(I, J) = 1
480 FI(I, J) = I(I, J)
485 V(J) = ABS (F(I, J)) + V(J)
490 NEXT I
495 IF V(J) >= VM THEN VM = V(J)
500 NEXT J
505 REM NOTE QUE SE BUSCA EL TERMINO MAYOR PARA CALCULAR LA NORMA
510 V = VM * T
515 IF V < 4 THEN K = 1 : GOTO 545
520 REM TRANSFORMACION DE BASE LOGARITMICA (SE TOMA ESE VALOR QUE ES UN POCO MAYOR QUE EL EXACTO POR RAZONES DE REDONDEO)
525 K = 1.4427 * LOG (V)
530 K = ABS (K)
535 K = K + 1
540 K = INT (K)
545 T1 = T / 2 ^ K
550 FOR JO = 11 TO 2 STEP - 1
555 FOR I = 1 TO N
560 FOR J = 1 TO N
565 P(I, J) = 0
570 M(I, J) = F(I, J) + (T1 / JO)
575 N(I, J) = FI(I, J)
580 NEXT J, I
585 REM LLAMA UNA SUBROUTINA DE MULTIPLICACION DE MATRICES

```

```

550 GOSUB 715
595 FOR I = 1 TO N
600 FOR J = 1 TO N
605 F(I,J) = I(I,J) + P(I,J)
610 NEXT J,I,J0
615 FOR K = K TO 1 STEP - 1
620 E = T / (2 ^ (K + 1))
625 FOR I = 1 TO N
630 FOR J = 1 TO N
635 IF B > (N + N - 1) THEN F(I,J) = P(I,J)
640 B = B + 1
645 P(I,J) = 0
650 M(I,J) = F(I,J) + E
655 N(I,J) = F(I,J)
660 NEXT J,I
665 GOSUB 715
670 FOR I = 1 TO N
675 FOR J = 1 TO N
680 M(I,J) = I(I,J) + P(I,J)
685 N(I,J) = F(I,J)
690 P(I,J) = 0
695 NEXT J,I
700 GOSUB 715
705 NEXT K
710 GOTO 745
712 REM SUBROUTINA : PRODUCTO DE MATRICES
715 FOR I = 1 TO N
720 FOR L = 1 TO N
725 FOR J = 1 TO N
730 P(I,L) = P(I,L) + M(I,J) * N(J,L)
735 NEXT J,L,I
740 RETURN
745 FOR I = 1 TO N
750 FOR J = 1 TO N
755 M(I,J) = F(I,J) * T
760 N(I,J) = P(I,J)
765 P(I,J) = 0
770 NEXT J,I
775 GOSUB 715
777 C2 = 0
780 FOR I = 1 TO N
785 FOR J = 1 TO N
790 F(I,J) = P(I,J) + I(I,J)
795 C2 = C2 + 1: IF C2 = 40 THEN C2 = 0: INPUT B *
800 PRINT "F(I,";I;",";J;")= "F(I,J)
805 NEXT J,I
807 INPUT B#
810 FOR I = 1 TO N
815 FOR J = 1 TO N
820 OP(I,J) = N(I,J) + T
825 GA(I) = OP(I,J) + G(J) + GA(I)
830 NEXT J
835 C2 = C2 + 1: IF C2 >= 20 THEN C2 = 0: INPUT B#
840 PRINT "GAMA(I,";I;")= "GA(I)
845 NEXT I
850 PRINT *
      TA DEL SISTEMA Y(Z);U(Z) UTILICE LA
855 PRINT * CONSULTE FRANKLYN & POWELL PAGINAS
      AUTOMATICO"
860 END

```

NOTA: PARA OBTENER LA FUNCION DE  
FORMULA  $M(ZI-FI)^{-1}G$   
131-137, 175-177 (EL APENDICE CONTIENE

TRANSFERENCIA EN EL DOMINIO DE ZE  
\*  
LA PARTE TEORICA DEL ESCALAMIENTO

PROGRAMA EJEMPLO  
POR EJEMPLO

DE UN

SOLUCION DE UN SISTEMA DE VARIABLES DE ESTADO  
( UTILIZANDO TIEMPO DE ESCALAMIENTO AUTOMATICO PARA LA OBTENCION DE LA MATRIZ EXPONENCIAL )  
AL INTERRUPTIRSE UN LISTADO CON EL SIMBOLO ( ? )

O P R I I

RETORN

CUAL ES EL ORDEN DEL SISTEMA ?

?2

CUAL ES EL PERIODO DE MUESTREO ?

?1

DADO EL SISTEMA DE LA FORMA :

$$X(K+1) = F X(K) + G U$$

INTRODUZCA EL RENGLON 1 DE LA MATRIZ F

?1

?1

INTRODUZCA EL RENGLON 2 DE LA MATRIZ F

?0

?0

INTRODUZCA EL VECTOR G

?0

?1

SUS DATOS SON :

F(1,1)= 1    F(1,2)= 1    F(2,1)= 0    F(2,2)= 0    G(1)= 0    G(2)= 1    ?

DESEA HACER ALGUNA CORRECCION (S,N)

?5

INDIQUE EN DONDE HARA SU CORRECCION :

EN : ORD

DEL SISTEMA ( N ) ,

PERIODO DE MUESTREO ( T ) ,

( F ) 0 ( G )

?F

INTRODUZCA PRIMERO EL RENGLON LUEGO LA COLUMNA , DESPUES EL NUEVO DATO

?1

??1

F(1,1)= ?0

ALGUN OTRO CAMBIO ? ( S , N )

?N

F(1,1)= 1

F(1,2)= .1

F(2,1)= 0

F(2,2)= 1

?

GAMA(1)= 5E-03

GAMA(2)= .1

NOTA: PARA OBTENER LA FUNCION DE TRANSFERENCIA EN EL DOMINIO DE ZETA DEL SIST.

Y(Z), U(Z) UTILICE LA FORMULA  $H(Z) = (ZI - F)^{-1}G$

CONSULTE FRANKLYN & POWELL PAGINAS 131-137, 175-177 (EL APENDICE CONTIENE LA PARTE TEORICA DEL ESCALAMIENTO AUTOMATI)

PROGRAMA # III

RAICES DE POLINOMIO



```

115F
5 HOME
10 REM NC ES EL GRADO ORIGINAL DEL POLINOMIO R2,S2 SON LOS VALORES INICIALES PARA LA SOLUCION DE LA ECUACION NO LINEAL EPS T
   ERMINO DE APROXIMACION O DE TOLERANCIA DE ERROR
15 PRINT " IMPRIMA EL GRADO DEL POLINOMIO "
20 INPUT NC
25 DIM A(NC),B(NC),P(NC),Q(NC),RL(NC),RG(NC)
30 PRINT " IMPRIMA EL TERMINO DE CONVERGENCIA ( ERROR PERMITIDO ) "
35 INPUT EP
40 PRINT " IMPRIMA LOS VALORES INICIALES DE APROXIMACION A LA RAZ R Y S "
45 INPUT R2,S2
50 PRINT " COLOQUE LOS COEFICIENTES DEL POLINOMIO TERMINO A TERMINO CON EL
   L FORMATO SIGUIENTE
   X^N+A(L)*X^(N-1)+...+A(N)=0 "
55 FOR I = 1 TO NC
60 PRINT
65 PRINT "A(I)=" ;
70 INPUT A(I)
75 NEXT I
80 M = 0 : J = 1
85 REM ENTRADA PARA VALORES INICIALES
90 R = R2
95 S = S2
100 L = 0
105 H = NC - 2 + M
110 IF (N - 2) < 0 THEN 130
115 IF (N - 2) = 0 THEN 145
120 IF (N - 2) > 0 THEN 255
125 REM CALCULA LA RAZ DEL FACTOR LINEAL (SI EXISTE)
130 RL(J) = - A(1)
135 GOTO 655
140 REM CALCULA EL PAR DE RAICES DEL ULTIMO TERMINO CUADRATICO
145 H = A(1) + A(1) - 4 * A(2)
150 IF H < 0 THEN 165
155 IF H = 0 THEN 200
160 IF H > 0 THEN 225
165 RH = - H
170 RA = SQR (RH)
175 RL(J) = - A(1) / 2
180 RL(J + 1) = RL(J)
185 RG(J) = RA / 2
190 RG(J + 1) = - RA / 2
195 GOTO 655
200 RL(J) = - A(1) / 2
205 RL(J + 1) = RL(J)
210 RG(J) = 0
215 RG(J + 1) = 0
220 GOTO 655
225 RA = SQR (H)
230 RL(J) = ( - A(1) + RA) / 2
235 RL(J + 1) = ( - A(1) - RA) / 2
240 RG(J) = 0
245 RG(J + 1) = 0
250 GOTO 655
255 REM ENTRADA PARA REDEFINIR R Y S
260 B(1) = A(1) - R
265 B(2) = A(2) - R + B(1) - S
270 FOR K = 3 TO N
275 B(K) = A(K) - R + B(K - 1) - S + B(K - 2)
280 NEXT K
285 P0 = B(N - 1)
290 S0 = B(N) + P + B(N - 1)
295 REM CALCULA LAS PARCIALES DE B(K) W.R.T R
300 P(1) = - 1
305 P(2) = R - B(1)
310 FOR K = 3 TO N

```

```

315 P(K) = - B(K - 1) - R + P(K - 1) - S + P(K - 2)
320 NEXT K
325 PR = P(N - 1)
330 SR = P(N) + R + P(N - 1) + B(N - 1)
335 REM CALCULA LAS DERIVADAS PARCIALES DE B(K) W.R.TS
340 Q(1) = 0
345 Q(2) = - 1
350 FOR K = 3 TO N
355 Q(K) = - B(K - 2) - R + Q(K - 1) - S + Q(K - 2)
360 NEXT K
365 RS = Q(N - 1)
370 SS = Q(N) + R + Q(N - 1)
375 REM RESOLVIENDO LA ECUACION PARA LA ECUACION RECURSIVA
380 DE = RR + SS - RS + SR
385 RN = - RC + SS + SC + RS
390 SN = - RR + SC + SR + RC
395 IF DE = 0 THEN 485
400 DR = RN / DE
405 DS = SN / DE
410 REM CALCULA LOS NUEVOS VALORES R,S
415 R = R + DR
420 S = S + DS
425 REM PRUEBA DE CONVERGENCIA PARA LA CORRECCION
430 G = ABS (DR) - EPS
435 IF G < = 0 THEN 445
440 IF G > 0 THEN 465
445 G2 = ABS (DS) - EPS
450 IF G2 < = 0 THEN 505
455 IF G2 > 0 THEN 465
460 REM PRUEBA DE ITERACION
465 IF L - 100 < = 0 THEN 475
470 IF L - 100 > 0 THEN 490
475 L = L + 1
480 GOTO 260
485 RZ = RZ + .5:SZ = SZ + .2: GOTO 80
490 PRINT "
      "" ""
495 GOTO 670
500 REM CALCULA EL PAR DE RAICES
505 G3 = R + R - 4 * S
510 IF G3 < 0 THEN 525
515 IF G3 = 0 THEN 560
520 IF G3 > 0 THEN 585
525 RM = - G3
530 RA = SQR (RM)
535 RL(J) = - R / 2
540 RL(J + 1) = - R / 2
545 RG(J) = RA / 2
550 RG(J + 1) = - RA / 2
555 GOTO 620
560 RL(J) = - R / 2
565 RL(J + 1) = - R / 2
570 RG(J) = 0
575 RG(J + 1) = 0
580 GOTO 620
585 RA = SQR (G3)
590 RL(J) = (- R + RA) / 2
595 RL(J + 1) = (- R - RA) / 2
600 RG(J) = 0
605 RG(J + 1) = 0
610 GOTO 620
615 REM CAMBIO DEL POLINOMIO ORIGINAL AL POLINOMIO REDUCIDO
620 M = M + 1
625 J = J + 2
630 IE = NC - 2 + M
635 FOR K = 1 TO MF

```

NO CONVERGE EN EL POLINOMIO REDUCIDO

```

640 A(K) = B(K)
645 NEXT K
650 GOTO 90
655 FOR K = 1 TO NC
660 PRINT "RREAL("K")= "R(K); " RIMAG("K")= "RG(K)
665 NEXT K IF G=0 STOP
??"EJEMPLO DE APLICACION"

```

EJEMPLO DE APLICACION

TRUN

IMPRIMA EL GRADO DEL POLINOMIO

??

IMPRIMA EL TERMINO DE CONVERGENCIA ( ERROR PERMITIDO )

?? 0001

IMPRIMA LOS VALORES INICIALES DE APROXIMACION A LA RAIZ R Y S

??

??

COLOQUE LOS COEFICIENTES DEL POLINOMIO TERMINO A TERMINO CON EL FORMATO S  
 $X^N + A(1)X^{N-1} + \dots + A(N) = 0$

UENTE :

A(1)= 232

A(2)= 9-12

A(3)= 26

RREAL(1)= .188182344 RIMAG(1)= .08717926

RREAL(2)= .188182344 RIMAG(2)= -.08717926

RREAL(3)= -32.3763647 RIMAG(3)= 0

PROGRAMA # IV

GRAFICA

```

5 TEXT
10 HOME
15 PRINT " DIBUJA LA RESPUESTA A ESCALON DE UN SISTEMA DE VARIABLES DE ESTADO DISCRETAS
"
20 PRINT " O DE UNA FUNCION DE TRANSFERENCIA EN EL DOMINIO DE ZETA "
25 PRINT " NOTA: DESPUES DE INTRODUCIR CADA DATO Y CUANDO SE DETENGA LA IMPRESION
O P R I M A ' R E T U R N ' "
30 PRINT " PARA UN SISTEMA DE VARIABLES OPRIMA ' S I S T E M A '
"
35 PRINT " PARA UNA FUNCION DE TRANSFERENCIA ' T R A N S F E R '
"
40 INPUT A$
45 IF A$ = "SISTEMA" THEN 65
50 IF A$ = "TRANSFER" THEN S8 = 1: GOTO 65
55 PRINT " SOLO OPRIMA (SISTEMA) O (TRANSFER) "
60 GOTO 30
65 IF S8 = 1 THEN 80
70 PRINT " DE EL ORDEN DEL SISTEMA "
75 GOTO 85
80 PRINT " EL GRADO DEL NUMERADOR DEBE SER MENOR DEL GRADO DEL DENOMINADOR
DE EL GRADO DEL DENOMINADOR "
85 INPUT N
90 IF N > INT (N) THEN 195
95 IF N > 20 THEN 300
100 IF N <= 0 THEN 310
105 DIM F(N,N), G(N), X(N), H(N), Y0(276), Y(276)
110 IF S8 = 1 THEN 210
115 PRINT " EL SISTEMA DE VARIABLES DE ESTADO DEBE SER DE LA FORMA :
X(N+1)=F1*X(N)+GAMA*U(N) Y(N)=H*X(N) "
120 PRINT " DE POR RENGLONES LA MATRIZ F1 "
125 FOR I = 1 TO N
130 PRINT " DE EL RENGLON "I" DE LA MATRIZ F1 "
135 FOR J = 1 TO N
140 INPUT F(I,J)
145 NEXT J,I
150 PRINT " DE EL VECTOR GAMA "
155 FOR I = 1 TO N
160 INPUT G(I)
165 NEXT I
170 PRINT " DE EL VECTOR H "
175 FOR I = 1 TO N
180 INPUT H(I)
185 NEXT I
190 GOTO 305
195 PRINT " CONTINUA EL PROCESO CONSIDERANDO SOLAMENTE LA PARTE ENTERA "
200 N = INT (N)
205 GOTO 95
210 PRINT " INTRODUCZA LOS COEFICIENTES DEL NUMERADOR EN EL FORMATO SIGUIENTE :
B(1)*Z^N+B(2)*Z^(N-1)+...+B(N-1) "
215 FOR I = 1 TO N
220 PRINT "B("I")= ";
225 INPUT H(I)
230 NEXT I
235 PRINT " INTRODUCZA LOS COEFICIENTES DEL DENOMINADOR EN EL SIGUIENTE FORMATO :
Z^N+A(1)*Z^(N-1)+...+A(N) "
240 FOR I = 1 TO N
245 PRINT "A("I")= ";
250 INPUT F(I,I)
255 F(I,I) = - F(I,I)
260 NEXT I
265 REM FORMACION DE LAS MATRICES DE LAS VARIABLES DE ESTADO
270 G(I) = 1
275 FOR I = 1 TO N
280 FOR J = 1 TO N
285 IF J = (I - 1) THEN F(I,J) = 1
290 NEXT J,I
295 GOTO 320
300 PRINT " ESTA COMPUTADORA NO TIENE SUFICIENTE CAPACIDAD PARA DICHS VALORES
DE UN NUEVO VALOR MAS CHICO "

```

```

395 0010 25
210 PRINT " EL ORDEN DEL SISTEMA DEBE SER MAYOR QUE CERO " 35 "
215 GOTO 40
220 PRINT " SUS DATOS SON " "
225 FOR I = 1 TO N
230 PRINT "B("I")= "H(I),
235 A4 = - FI(L,I)
240 PRINT "R("I")= "A4,
245 IF I > 16 THEN INPUT A#
250 NEXT I
255 PRINT " DESEA HACER CORRECCIONES (S,N) "
260 INPUT A#
265 IF A# = "S" THEN 395
270 IF A# = "N" THEN 645
275 PRINT " INTRODUCZA SOLAMENTE S O N "
280 GOTO 360
285 PRINT " QUIERE CORREGIR UN DATO (S,N) "
290 GOTO 430
295 PRINT " DESEA CORREGIR COEFICIENTES DEL NUMERADOR O DEL DENOMINADOR (N,D) "
300 INPUT A#
405 IF A# = "N" THEN 450
410 IF A# = "D" THEN 495
415 PRINT " OPRIMA CORRECTAMENTE N O D "
420 GOTO 400
425 PRINT " QUIERE CORREGIR OTRO DATO (S,N) "
430 INPUT A#
435 IF A# = "N" THEN 645
440 IF A# = "S" THEN 525
445 PRINT " NO SE RECONOCE TAL DATO SOLO (N) O (S) , VUELVA A DARLO NUEVAMENTE " : GOTO 425
450 PRINT " OPRIMA EL INDICE DEL COEFICIENTE QUE CAMBIARA , POSTERIORMENTE OPRIMA EL NUEVO VALOR
.
455 INPUT I
460 IF I > N THEN PRINT " NO EXISTE TAL INDICE " : GOTO 450
465 INPUT H(I)
470 PRINT " ALGUN OTRO CAMBIO (S,N) "
475 INPUT A#
480 IF A# = "S" THEN 395
485 IF A# = "N" THEN 645
490 PRINT " SOLAMENTE SOLO (S) O (N) PORFAVOR " : GOTO 475
495 PRINT " OPRIMA EL INDICE DEL COEFICIENTE QUE CAMBIARA , POSTERIORMENTE OPRIMA EL NUEVO VALOR
.
500 INPUT I
505 IF I > N THEN PRINT " NO EXISTE DICHO ELEMENTO VUELVA A METER EL VALOR CORRECTO " : GOTO 500
510 INPUT FI(L,I)
515 FI(L,I) = - FI(L,I)
520 GOTO 470
525 PRINT " INDIQUE QUE MATRIZ DESEA CAMBIAR ( FI ) , ( GAMA ) O ( H ) "
530 INPUT A#
535 IF A# = "FI" THEN 555
540 IF A# = "GAMA" THEN 595
545 IF A# = "H" THEN 615
550 PRINT " NO EXISTE EN ESTE PROGRAMA " : GOTO 525
555 PRINT " DE PRIMERO EL RENGLON , DESPUES LA COLUMNA Y FINALMENTE EL NUEVO VALOR "
560 INPUT I,J
565 IF I > N OR J > N THEN 590
570 INPUT FI(I,J)
575 GOTO 425
580 PRINT " NO EXISTE TAL ELEMENTO " : GOTO 555
585 PRINT " DE PRIMERO EL RENGLON DESPUES EL NUEVO VALOR "
590 INPUT I
595 IF I > N THEN 610
600 INPUT GA(I)
605 GOTO 425
610 PRINT " NO EXISTE TAL ELEMENTO " : GOTO 595
615 PRINT " DE PRIMERO EL RENGLON DESPUES EL DATO "
620 INPUT I
625 IF I > N THEN PRINT " NO EXISTE TAL ELEMENTO " : GOTO 615

```

```

630 INPUT H,I
635 GOTO 425
640 REM EMPIEZAN EL PROGRAMA
645 HGR
650 HCOLOR= 7
655 HPLOT 0,0 TO 279,0 TO 279,155 TO 0,155 TO 0,0
660 REM COMENZAMOS OPERACIONES PARA LA OBTENCION DE LA RESPUESTA A ESCALON
665 FOR I = 1 TO N
670 X(I) = 0
675 NEXT I
680 FOR K = 0 TO 276 STEP 6
685 Y = 0
690 FOR I = 1 TO N
695 AN(I) = 0
700 FOR J = 1 TO N
705 AN(I) = F1(I,J) * X(J) + AN(I)
710 NEXT J
715 Y = H(I) * X(I) + Y
720 NEXT I
725 FOR I = 1 TO N
730 X(I) = AN(I) + GA(I)
735 NEXT I
740 IF Y < 0 THEN 835
745 YG(K) = Y
750 IF YM < YG(K) THEN YM = YG(K)
755 IF YN < YG(K) THEN YN = YG(K)
760 NEXT K
765 IF YM = 0 THEN 850
770 IF C4 > 1 THEN 860
775 ES = 155 / YM
780 HCOLOR= 0
785 HPLOT 0,154 TO 0,155
790 FOR K = 6 TO 276 STEP 6
795 HCOLOR= 3
800 IF ES > 155 THEN 810
805 HPLOT K,155 - ES
810 YD = - YG(K) + ES + 155
815 HPLOT K,YD
820 HPLOT K + 1,YD
825 NEXT K
830 END
835 YG(K) = - Y
840 C4 = C4 + 1
845 GOTO 755
850 PRINT " LA GRAFICA ES CERO
855 GOTO 830
860 HPLOT 0,78 TO 279,78
865 ES = 77 / YM
870 HCOLOR= 3
875 FOR K = 6 TO 276 STEP 6
880 IF YG(K) = 0 THEN 895
885 YD = - YG(K) + ES + 77
890 HPLOT K,YD
895 IF YG(K) = 0 THEN 910
900 YD = YG(K) + ES + 78
905 HPLOT K,YD
910 NEXT K
915 GOTO 830

```

PROGRAMA # V

VALORES CARACTERISTICOS



```

LIST
5 HOME PRINT
10 PRINT " ESTE PROGRAMA ES PARA OBTENER UN ARREGLO MATRICIAL PARA LA OBTENCION DE LOS VALORES CARACTERISTICOS
15 PRINT PRINT " DE EL ORDEN DE LA MATRIZ A "
20 PRINT PRINT " N= ".
25 INPUT N
30 IF N = 0 GOTO 265
40 DIM A(N,N),V(N),AA(N,N),C(N,N),X(N),AI(N,N),D(N),CO(N,N+1),TRI(N,N+1),X(N),K(L,N),B(N)
45 FOR I = 1 TO N
50 PRINT PRINT " DE EL RENGLON "I" DE LA MATRIZ A": PRINT
55 FOR J = 1 TO N "A("I","J")= "
60 INPUT A(I,J)
70 NEXT J,I
75 GOTO 275
80 V(1) = 1
85 FOR I = 1 TO N
90 V(I) = V(I)
95 FOR J = 1 TO N
100 IF J = I THEN AA(I,J) = 1
105 NEXT J,I
110 K1 = 1:K2 = N
115 IF K1 > N THEN 240
120 FOR I = 1 TO N
125 FOR J = K2 TO K2
130 C(I,J) = V(I)
135 NEXT J,I
140 FOR I = 1 TO N
145 V(I) = 0
150 NEXT I
155 K1 = K1 + 1
160 K2 = K2 - 1
165 FOR I = 1 TO N
170 FOR L = 1 TO N
175 FOR J = 1 TO N
180 AI(I,L) = AI(I,L) + AA(I,J) * A(J,L)
185 NEXT J,L,I
190 FOR I = 1 TO N
195 FOR J = 1 TO N
200 AA(I,J) = AI(I,J)
205 AI(I,J) = 0
210 NEXT J,I
215 FOR I = 1 TO N
220 FOR J = 1 TO N
225 V(I) = V(I) + AA(I,J) * V(J)
230 NEXT J,I
235 GOTO 115
240 FOR I = 1 TO N
245 D(I) = - V(I)
250 NEXT I
255 PRINT PRINT
260 GOTO 245
265 PRINT PRINT " EL ORDEN DEL SISTEMA DEBE SER UN NUMERO ENTERO MAYOR QUE CERO"
270 GOTO 15
275 PRINT PRINT " QUIERE CAMBIAR ALGUN DATO DE ESTA MATRIZ ? (S/N)"
280 PRINT INPUT A$
285 IF A$ = "N" THEN 80
290 IF A$ = "S" THEN 200
295 GOTO 275
300 PRINT PRINT " DE PRIMERO EL RENGLON: DESPUES LA COLUMNA Y FINALMENTE EL DATO"
305 PRINT INPUT I,J
310 IF I > N THEN 325
315 IF J > N THEN 325
320 PRINT PRINT " A("I","J")= "
325 INPUT A(I,J)
330 GOTO 275
335 PRINT PRINT " NO EXISTE TAL ELEMENTO "
340 GOTO 200
345 FOR I = 1 TO N

```

```

350 B(I) = D(I)
355 NEXT I
360 REM DEFINIMOS EU COMO B(N)
365 REM TRANSPOSICION Y TRIANGULARIZACION
370 FOR I = 1 TO N
375 FOR J = 1 TO N
380 CO(L,J) = C(L,J)
385 NEXT J
390 CO(L,N + 1) = B(I)
395 NEXT I
400 FOR I = 1 TO N
405 DIV = CO(I,I)
410 IF DIV = 0 THEN 490
415 FOR J = 1 TO N + 1
420 TRI(L,J) = CO(L,J) / DIV
425 NEXT J
430 FOR K = 1 TO N
435 IF K <= I THEN 465
440 PRO = CO(K,I)
445 FOR J = 1 TO N + 1
450 TRI(K,J) = CO(K,J) - TRI(L,J) * PRO
455 CO(K,J) = TRI(K,J)
460 NEXT J
465 NEXT K
470 B(I) = TRI(L,N + 1)
475 NEXT I
480 REM TERMINAMOS DE TRIANGULARIZAR
485 GOTO 620
490 AM = 0
495 FOR L = 1 TO N
500 IF I = 1 AND L = N THEN 655
505 IF CO(L,I) = 0 THEN 520
510 AM = CO(L,I)
515 GOTO 530
520 NEXT L
525 IF AM = 0 THEN 560
530 FOR J = 1 TO N + 1
535 CO(L,J) = CO(L,J) + CO(L,I)
540 NEXT J
545 D8 = D8 + 1
550 IF D8 = > N ^ 2 THEN 655
555 GOTO 400
560 FOR L = 1 + 1 TO N
565 IF CO(L,I) = 0 THEN 580
570 AM = CO(L,I)
575 GOTO 530
580 NEXT L
585 FOR L = I - 1 TO 1 STEP - 1
590 IF CO(L,I) = 0 THEN 605
595 AM = CO(L,I)
600 GOTO 530
605 NEXT L
610 GOTO 655
615 NEXT L
620 FOR I = 1 TO N
625 L = N - I + 1
630 SP = 0
635 IF L + 1 > N THEN 665
640 FOR J = L + 1 TO N
645 SP = SP + TRI(L,J) * X(J)
650 GOTO 675
655 PRINT "SISTEMA INCOMPATIBLE"
660 END
665 X(L) = B(L)
670 GOTO 685
675 NEXT J

```

```

690 K(L,I) = K(L,I) + ALP(J,I) + K(L,I)
695 NEXT I
690 FOR I = 1 TO N
695 FOR J = 1 TO N
700 K(L,I) = K(J) + ALP(J,I) + K(L,I)
705 NEXT J
710 F8 = F8 + 1. IF F8 > 4 THEN F8 = 0: INPUT F8#
715 NEXT I
720 REM NC ES EL GRADO ORIGINAL DEL POLINOMIO R2,S2 SON LOS VALORES INICIALES PARA LA SOLUCION DE LA ECUACION NO LINEAL EPS
    TERMINO DE APROXIMACION O DE TOLERANCIA DE ERROR
725 NC = N
730 N = 0
735 DIM AS(NC), P(NC), Q(NC), RL(NC), RG(NC)
740 REM SE TOMA UN ERROR DE .0001
745 EP = .0001
750 REM INICIAMOS CON R2,S2 ARBITRARIOS
755 R2 = 1.52 = - 1
760 REM IGUALACION DE COEFICIENTES
765 FOR I = 1 TO NC
770 AS(I) = X(I)
775 NEXT I
780 M = 0: J = 1
785 REM ENTRADA PARA VALORES INICIALES
790 R = R2
795 S = S2
800 L = 0
805 N = NC - 2 * M
810 IF (N - 2) < 0 THEN 830
815 IF (N - 2) = 0 THEN 845
820 IF (N - 2) > 0 THEN 955
825 REM CALCULA LA RAIZ DEL FACTOR LINEAL (SI EXISTE)
830 RL(J) = - AS(1)
835 GOTO 1255
840 REM CALCULA EL PAR DE RAICES DEL ULTIMO TERMINO CUADRATICO
845 H = AS(1) * AS(1) - 4 * AS(2)
850 IF H < 0 THEN 865
855 IF H = 0 THEN 900
860 IF H > 0 THEN 925
865 RM = - H
870 RA = SQR (RM)
875 RL(J) = - AS(1) / 2
880 RL(J + 1) = RL(J)
885 RG(J) = RA / 2
890 RG(J - 1) = - RA / 2
895 GOTO 1255
900 RL(J) = - AS(1) / 2
905 RL(J + 1) = RL(J)
910 RG(J) = 0
915 RG(J + 1) = 0
920 GOTO 1255
925 RA = SQR (H)
930 RL(J) = (- AS(1) + RA) / 2
935 RL(J + 1) = (- AS(1) - RA) / 2
940 RG(J) = 0
945 RG(J + 1) = 0
950 GOTO 1255
955 REM ENTRADA PARA REDEFINIR R Y S
960 B(1) = AS(1) - R
965 B(2) = AS(2) - R + B(1) - S
970 FOR K = 3 TO N
975 B(K) = AS(K) - R + B(K - 1) - S + B(K - 2)
980 NEXT K
985 RC = B(N - 1)
990 SC = B(N) + R + B(N - 1)
995 REM CALCULA LAS PARCIALES DE B(K) PARA
1000 B(1) = - 1

```

```

1005 P(2) = R + B(1)
1010 FOR K = 3 TO N
1015 P(K) = - B(K - 1) - R * P(K - 1) - S * P(K - 2)
1020 NEXT K
1025 PR = P(N - 1)
1030 SR = P(N) + R * P(N - 1) + B(N - 1)
1035 REM CALCULA LAS DERIVADAS PARCIALES DE B(K) W.R.T.S
1040 Q(1) = 0
1045 Q(2) = - 1
1050 FOR K = 3 TO N
1055 Q(K) = - B(K - 2) - R * Q(K - 1) - S * Q(K - 2)
1060 NEXT K
1065 RS = Q(N - 1)
1070 SS = Q(N) + R * Q(N - 1)
1075 REM RESOLVIENDO LA ECUACION PARA LA ECUACION RECURSIVA
1080 DE = RR + SS - RS * SR
1085 RN = - RC + SS + SC + RS
1090 SN = - RR + SC + SR + RC
1095 IF DE = 0 THEN 1185
1100 DR = RN / DE
1105 DS = SN / DE
1110 REM CALCULA LOS NUEVOS VALORES R, S
1115 R = R + DR
1120 S = S + DS
1125 REM PRUEBA DE CONVERGENCIA PARA LA CORRECCION
1130 G = ABS(DR) - EPS
1135 IF G < = 0 THEN 1145
1140 IF G > 0 THEN 1165
1145 G2 = ABS(DS) - EPS
1150 IF G2 < = 0 THEN 1205
1155 IF G2 > 0 THEN 1165
1160 REM PRUEBA DE ITERACION
1165 IF L - 100 < = 0 THEN 1175
1170 IF L - 100 > 0 THEN 1190
1175 L = L + 1
1180 GOTO 960
1185 R2 = R2 + .5: S2 = S2 + .2: GOTO 780
1190 PRINT "
      N= "N
1195 GOTO 1300
1200 REM CALCULA EL PAR DE RAICES
1205 G3 = R + R - 4 * S
1210 IF G3 < 0 THEN 1225
1215 IF G3 = 0 THEN 1260
1220 IF G3 > 0 THEN 1285
1225 RM = - G3
1230 RA = SQR(RM)
1235 RL(J) = - R / 2
1240 RL(J + 1) = - R / 2
1245 RG(J) = RA / 2
1250 RG(J + 1) = - RA / 2
1255 GOTO 1320
1260 RL(J) = - R / 2
1265 RL(J + 1) = - R / 2
1270 RG(J) = 0
1275 RG(J + 1) = 0
1280 GOTO 1320
1285 RA = SQR(G3)
1290 RL(J) = (- R + RA) / 2
1295 RL(J + 1) = (- R - RA) / 2
1300 RG(J) = 0
1305 RG(J + 1) = 0
1310 GOTO 1320
1315 REM CAMBIO DEL POLINOMIO ORIGINAL AL POLINOMIO REDUCIDO
1320 M = M + 1
1325 J = J + 2

```

NO CONVERGE EN EL POLINOMIO REDUCIDO

```

1330 NE = NC - 2 + N
1335 FOR K = 1 TO NE
1340 AS(K) = B(K)
1345 NEXT K
1350 GOTO 790
1355 PRINT : PRINT "  LOS  VALORES
      C O S  S O N
1360 FOR K = 1 TO NC
1365 PRINT
1370 PRINT "AREAL("K")= "RL(K);"  RIMAG("K")= "RG(K)
1375 NEXT K
1380 END
??"EJEMPLO"

```

CARACTERISTI  
"

EJEMPLO  
??"EJEMPLO"

ESTE PROGRAMA ES PARA OBTENER UN ARREGLO MATRICIAL PARA LA OBTENCION DE LOS VALORES CARACTERISTICOS DE EL ORDEN DE LA MATRIZ A

```

N= 03
DE EL RENGLON 1 DE LA MATRIZ A
A(1,1)= 01
A(1,2)= 02
A(1,3)= 03
DE EL RENGLON 2 DE LA MATRIZ A
A(2,1)= 04
A(2,2)= 05
A(2,3)= 06
DE EL RENGLON 3 DE LA MATRIZ A
A(3,1)= ?

```

??"ENTER  
03

```

A(3,2)= 09
A(3,3)= 01

```

QUIERE CAMBIAR ALGUN DATO DE ESTA MATRIZ ?

(5,N)

LOS VALORES

CARACTERISTICOS SON:

```

AREAL(1)= - 373732438  RIMAG(1)= 0
AREAL(2)= -5.57796943  RIMAG(2)= 0
AREAL(3)= 12.9517049  RIMAG(3)= 0

```

PROGRAMA # VI

ACKERMANN

```

1030
5  TEXT HOME
10  INVERSE
15  PRINT "          ** ACKERMANN **          " PRINT
20  NORMAL
25  PRINT "          PROGRAMA PARA CALCULAR          LA LEY DE CONTROL POR          LA FORMULA DE ACKERMANN"
30  PRINT "          * NOTA: DESPUES DE INTRODUCIR CADA          DATO Y CUANDO APAREZCA EL SIGNO ?          OPRIMA LA TECLA RETUR"
    "          "
35  PRINT "          DAR EL ORDEN DEL SISTEMA": PRINT
40  PRINT "          N = ": INPUT N
45  PRINT "          QUIERE CAMBIAR EL ORDEN DEL SISTEMA ?          (S,N)"
50  PRINT "          INPUT A$
55  IF A$ = "S" THEN 35
60  N = INT (N)
65  IF N <= 0 THEN 810
70  DIM F(N,N), GA(N,1), I(N,N), ALP(N,N), AMP(N,N), C(N,N), F(N,1), FO(N,N), ANP(N,N), DE(N,1), AOP(N,N), EU(N,N), CO(N,N + 1), TRI(N,N +
    1), X(N), K(1,N), SR(N), SI(N)
75  PRINT "          DAR EL PERIODO DE MUESTREO": PRINT
80  PRINT "          T = ": INPUT T
85  PRINT "          QUIERE CAMBIAR EL PERIODO DE MUESTREO ?          (S,N)"
90  PRINT "          INPUT A$
95  IF A$ = "S" THEN 75
100 IF T < 0 THEN 830
105 PRINT "          DE POR RENGLONES LA MATRIZ FI": PRINT
110 FOR I = 1 TO N
115 PRINT "          DE EL RENGLON "I" DE LA MATRIZ FI"
120 FOR J = 1 TO N
125 INPUT FI(I,J)
130 NEXT J,I
135 FOR I = 1 TO N
140 PRINT "          DE EL ELEMENTO DE LA MATRIZ GAMA("I",1)"
145 INPUT GA(I,1)
150 NEXT I
155 GOTO 880
160 REM FORMACION DE LA MATRIZ IDENTIDAD
165 FOR I = 1 TO N
170 FOR J = 1 TO N
175 IF J = I THEN I(I,J) = 1
180 ALPHA(I,J) = I(I,J)
185 NEXT J,I
190 K3 = N:K5 = 1
195 K1 = 1
200 IF K3 > N GOTO 260
205 PRINT "          UBICACION DE LOS POLOS DE LA ECUACION          CARACTERISTICA DESEADA, PLANO S": PRINT
210 PRINT "          PARTE REAL= ",
215 INPUT A
220 PRINT "          PARTE IMAGINARIA= ",
225 INPUT B
230 GOTO 1215
235 SR(K5) = A
240 SI(K5) = B
245 IF K3 = 1 THEN 850
250 IF B < > 0 THEN 660
255 GOTO 570
260 REM SE CONSTRUIRA LA MATRIZ DE CONTROLABILIDAD
265 FOR I = 1 TO N
270 FOR J = 1 TO N
275 C(I,J) = I(I,J)
280 FO(I,J) = 0
285 NEXT J,I
290 FOR I = 1 TO N
295 E(I,1) = GA(I,1)
300 NEXT I

```

```

305 K2 = 1
310 IF K2 > N GOTO 425
315 FOR I = 1 TO N
320 FOR J = K2 TO K2
325 C(I,J) = E(I,I)
330 NEXT J,I
335 FOR I = 1 TO N
340 E(I,1) = 0
345 NEXT I
350 FOR I = 1 TO N
355 FOR L = 1 TO N
360 FOR J = 1 TO N
365 F0(I,L) = F0(I,L) + F1(I,J) + I(J,L)
370 NEXT J,L,I
375 FOR I = 1 TO N
380 FOR J = 1 TO N
385 I(I,J) = F0(I,J)
390 F0(I,J) = 0
395 NEXT J,I
400 FOR I = 1 TO N
405 FOR J = 1 TO N
410 E(I,1) = E(I,1) + I(I,J) + G0(J,1)
415 NEXT J,I
420 K2 = K2 + 1: GOTO 310
425 SPEED = 170
430 PRINT "      *** D A T O S ***      "
435 PRINT : PRINT " ORDEN DEL SISTEMA= "N
440 PRINT : PRINT " PERIODO DE MUESTREO= "T
445 PRINT : PRINT "      ESTA ES LA MATRIZ F1". PRINT
450 FOR I = 1 TO N
455 FOR J = 1 TO N
460 PRINT "      F1("I","J")= "F1(I,J)
465 NEXT J,I
470 PRINT : PRINT "      ESTA ES LA MATRIZ GAMA": PRINT
475 FOR I = 1 TO N
480 PRINT "      G0("I",1)= "G0(I,1)
485 NEXT I
490 PRINT : PRINT "      ESTOS SON LOS POLOS "": PRINT
495 FOR I = 1 TO N
500 PRINT "PR= "SR(I), "PI= "SI(I)
505 NEXT I
510 PRINT : SPEED = 255
515 INPUT A#
520 FOR I = 1 TO N
525 FOR J = 1 TO N
530 PRINT "      ALP("I","J")= "ALP(I,J)
535 NEXT J,I
540 PRINT
545 FOR I = 1 TO N
550 FOR J = 1 TO N
555 PRINT "      C("I","J")= "C(I,J)
560 NEXT J,I
565 GOTO 1235
570 A1 = EXP (R + T)
575 FOR I = 1 TO N
580 FOR J = 1 TO N
585 F0(I,J) = F1(I,J) - A1 * I(I,J)
590 NEXT J,I
595 FOR I = 1 TO N
600 FOR L = 1 TO N
605 FOR J = 1 TO N
610 W0(I,L) = W0(I,L) + ALP(I,J) * F0(J,L)
615 NEXT J,L,I
620 FOR I = 1 TO N
625 FOR L = 1 TO N
630 ALP(I,L) = W0(I,L)

```



```
635 AMP(I,J) = 0
640 NEXT J,I
645 K3 = K3 - 1
650 K1 = K1 + 1
655 K5 = K5 + 1: GOTO 200
660 A1 = - 2 * EXP (A * T) * COS (B * T)
665 A2 = EXP (2 * A * T)
670 FOR I = 1 TO N
675 FOR J = 1 TO N
680 FO(I,J) = FI(I,J)
685 NEXT J,I
690 FOR I = 1 TO N
695 FOR L = 1 TO N
700 FOR J = 1 TO N
705 AMP(I,L) = AMP(I,L) + FI(I,J) * FO(J,L)
710 NEXT J,L,I
715 FOR I = 1 TO N
720 FOR L = 1 TO N
725 AMP(I,L) = AMP(I,L) + A1 * FI(I,L) + A2 * I(I,L)
730 NEXT L,I
735 FOR I = 1 TO N
740 FOR L = 1 TO N
745 FOR J = 1 TO N
750 AOP(I,L) = AOP(I,L) + ALP(I,J) * AMP(J,L)
755 NEXT J,L,I
760 FOR I = 1 TO N
765 FOR J = 1 TO N
770 AMP(I,J) = 0
775 ALP(I,J) = AOP(I,J)
780 AOP(I,J) = 0
785 NEXT J,I
790 K3 = K3 - 2
795 K1 = K1 + 2
800 K5 = K5 + 2: GOTO 200
805 REM PROTECCION AL PROGRAMA
810 FLASH
815 PRINT : PRINT "EL ORDEN DEL SISTEMA DEBE SER UN NUMERO ENTERO MAYOR QUE CERO": PRINT
820 NORMAL
825 GOTO 35
830 FLASH
835 PRINT : PRINT " EL PERIODO DE MUESTREO DEBE SER POSITIVO": PRINT
840 NORMAL
845 GOTO 75
850 IF B < > 0 THEN 860
855 GOTO 570
860 FLASH
865 PRINT : PRINT "EL NUMERO DE POLOS DADO ES MAYOR QUE EL ORDEN DEL SISTEMA": PRINT
870 NORMAL
875 GOTO 205
880 PRINT : PRINT " ** SUS DATOS SON **"
885 PRINT : PRINT "ORDEN DEL SISTEMA= "N
890 PRINT : PRINT " PERIODO DE MUESTREO= "T
895 PRINT : PRINT " MATRIZ FI"
900 INPUT A#
905 K4 = 1
910 FOR I = 1 TO N
915 FOR J = 1 TO N
920 IF K4 = 10 THEN 955
925 IF K4 = 20 THEN 955
930 IF K4 = 30 THEN 955
935 PRINT : PRINT " FI("I","J")= "FI(I,J)
940 K4 = K4 + 1
945 NEXT J,I
950 GOTO 965
955 PRINT : INPUT A#
960 GOTO 935
```

```

820 INPUT A#
830 PRINT : PRINT "      MATRIZ GAMMA": PRINT
840 FOR I = 1 TO N
850 IF I = 10 THEN 1005
860 IF I = 20 THEN 1005
870 PRINT : PRINT "      GAC(I,1)= GAC(I,1)"
880 NEXT I
890 GOTO 1015
1005 PRINT : INPUT A#
1010 GOTO 930
1015 PRINT : INPUT A#
1020 PRINT : PRINT "QUIERE CORREGIR ALGUN DATO ?" (S,N)
1025 GOTO 1035
1030 PRINT : PRINT "QUIERE CORREGIR OTRO DATO ?" (S,N)
1035 PRINT : INPUT A#
1040 IF A# = "N" THEN 1060
1045 IF A# = "S" THEN 1055
1050 GOTO 1030
1055 PRINT : PRINT " INDIQUE QUE MATRIZ DESEA CAMBIAR" (FI O GA): PRINT
1060 INPUT A#
1065 IF A# = "F:" THEN 1090
1070 IF A# = "GA" THEN 1140
1075 FLASH
1080 PRINT : PRINT " NO EXISTE EN ESTE PROGRAMA "
1085 NORMAL : GOTO 1030
1090 PRINT : PRINT "DE PRIMERO EL RENGLON,DESPUES LA COLUMNA Y FINALMENTE EL DATO": PRINT
1095 INPUT I,J
1100 IF I > N THEN 1125
1105 IF J > N THEN 1125
1110 PRINT : PRINT "      FI(I,J)= "
1115 INPUT FI(I,J)
1120 GOTO 1030
1125 FLASH
1130 PRINT : PRINT " NO EXISTE TAL ELEMENTO " : PRINT
1135 NORMAL : GOTO 1030
1140 PRINT : PRINT "DE PRIMERO EL RENGLON,DESPUES EL DATO" : PRINT
1145 INPUT I
1150 IF I > N THEN 1170
1155 PRINT : PRINT "      GAC(I,1)= "
1160 INPUT GAC(I,1) : PRINT
1165 GOTO 1030
1170 FLASH
1175 PRINT : PRINT " NO EXISTE TAL ELEMENTO"
1180 NORMAL : GOTO 1140
1185 PRINT : PRINT " UBICACION DEL NUEVO POLO" : PRINT
1190 PRINT "PARTE REAL= "
1195 INPUT A
1200 PRINT : PRINT "PARTE IMAGINARIA= "
1205 INPUT B
1210 GOTO 235
1215 PRINT : PRINT " QUIERE CAMBIAR LOS POLOS ?" (S,N)
1220 PRINT : INPUT A#
1225 IF A# = "S" THEN 1195
1230 GOTO 235
1235 B(N) = 1
1240 REM DEFINIMOS EN COMO B(N)
1245 REM TRANSPOSICION Y TRIANGULARIZACION
1250 FOR I = 1 TO N
1255 FOR J = 1 TO N
1260 C(I,J) = G(J,I)
1265 NEXT J
1270 C(I,N+1) = B(I)
1275 NEXT I
1280 FOR I = 1 TO N
1285 C(I) = C(I,I)
1290 IF C(I) = A THEN 1295

```

```

1305 FOR J = 1 TO N + 1
1310 TRICK(J) = CO(K, J) / DIV
1315 NEXT J
1320 FOR I = 1 TO N
1325 IF K(I) = 1 THEN 1345
1330 PRO = CO(K, I)
1335 FOR J = 1 TO N + 1
1340 TRICK(J) = CO(K, J) - TRI(L, J) + PRO
1345 CO(K, J) = TRICK(J)
1350 NEXT J
1355 NEXT I
1360 B(I) = TRI(L, N + 1)
1365 NEXT I
1370 REM TERMINAMOS DE TRIANGULARIZAR
1375 GOTO 1500
1380 AM = 0
1385 FOR L = 1 TO N
1390 IF I = 1 AND L = N THEN 1535
1395 IF CO(L, I) = 0 THEN 1400
1400 AM = CO(L, I)
1405 GOTO 1410
1410 NEXT L
1415 IF AM = 0 THEN 1440
1420 FOR J = 1 TO N + 1
1425 CO(L, J) = CO(L, J) + CO(L, I)
1430 NEXT J
1435 DB = DB + 1
1440 IF DB = > N ^ 2 THEN 1535
1445 GOTO 1200
1450 FOR L = I + 1 TO N
1455 IF CO(L, I) = 0 THEN 1460
1460 AM = CO(L, I)
1465 GOTO 1410
1470 NEXT L
1475 FOR L = I - 1 TO 1 STEP - 1
1480 IF CO(L, I) = 0 THEN 1485
1485 AM = CO(L, I)
1490 GOTO 1410
1495 NEXT L
1500 FOR I = 1 TO N
1505 L = N - I + 1
1510 SP = 0
1515 IF L + 1 > N THEN 1560
1520 FOR J = L + 1 TO N
1525 SP = SP + TRI(L, J) * X(J)
1530 GOTO 1570
1535 FLASH
1540 PRINT "SISTEMA INCOMPATIBLE ": PRINT
1545 NORMAL
1550 PRINT " ** FIN DEL PROGRAMA ** "
1555 END
1560 X(L) = B(L)
1565 GOTO 1580
1570 NEXT J
1575 X(L) = B(L) - SP
1580 NEXT I
1585 FOR I = 1 TO N
1590 FOR J = 1 TO N
1595 K(L, I) = X(J) + ALP(J, I) + K(L, I)
1600 NEXT J
1605 F8 = F8 + 1: IF F8 > 4 THEN F8 = 0: INPUT F8
1610 PRINT
1615 PRINT " K(L, I) = " X(L, I)
1620 NEXT I

```

JPRINT\* EJEMPLO DE APLICACION\*  
EJEMPLO DE APLICACION

JRUN

\*\* ACKERMANN \*\*

PROGRAMA PARA CALCULAR

\* NOTA: DESPUES DE INTRODUCIR CADA  
DAR EL ORDEN DEL SISTEMA

N = ?2

QUIERE CAMBIAR EL ORDEN DEL SISTEMA ?

(S, N)

?N

DAR EL PERIODO DE MUESTREO

T = ?1

QUIERE CAMBIAR EL PERIODO DE MUESTREO ?

(S, N)

?N

DE POR RENGONES LA MATRIZ FI

DE EL RENGON 1 DE LA MATRIZ FI

?1.2

?2

DE EL RENGON 2 DE LA MATRIZ FI

?1.5

?3

DE EL ELEMENTO DE LA MATRIZ GAMA(1,1)

? .005

DE EL ELEMENTO DE LA MATRIZ GAMA(2,1)

? .1

\*\* SUS DATOS SON \*\*

ORDEN DEL SISTEMA= 2

PERIODO DE MUESTREO= 1

MATRIZ FI

?

FI(1,1)= 1.2

FI(1,2)= 2

FI(2,1)= .5

FI(2,2)= 3

?

MATRIZ GAMA

GA(1,1)= 5E-03

GA(2,1)= .1

?

QUIERE CORREGIR ALGUN DATO ?

(S, N)

?N

UBICACION DE LOS POLOS DE LA ECUACION CARACTERISTICA DESEADA, PLANO S

PARTE REAL= ?-3

PARTE IMAGINARIA= ?0

QUIERE CAMBIAR LOS POLOS ?

(S, N)

?N

UBICACION DE LOS POLOS DE LA ECUACION CARACTERISTICA DESEADA, PLANO S

PARTE REAL= ?-2

PARTE IMAGINARIA= ?0

QUIERE CAMBIAR LOS POLOS ?

(S, N)

?N

\*\*\* D A T O S \*\*\*

ORDEN DEL SISTEMA= 2

PERIODO DE MUESTREO= 1

ESTA ES LA MATRIZ FI

FI(1,1)= 1.2

FI(1,2)= 2

FI(2,1)= .5

FI(2,2)= 3

ESTA ES LA MATRIZ GAMA

GA(1,1)= 5E-03

GA(2,1)= .1

ESTOS SON LOS POLOS

PR= -3

PI= 0

--

--

PK= -2  
?

PI= 0

50 -

ALP(1,1)= 2.22459113  
ALP(1,2)= 8.0297553  
ALP(2,1)= 2.00743883  
ALP(2,2)= 9.45127009  
C(1,1)= 5E-03  
C(1,2)= .206  
C(2,1)= .1  
C(2,2)= .3025  
K(1,1) = 11.1288497  
K(1,2) = 39.592334

J  
JOB#0  
SYNTAX ERROR

PROGRAMA # VII

ROOT LOCUS

```

J?"
LIST
10 REM PRESENTACION
15 TEXT : HOME : PRINT
20 INVERSE
25 PRINT "          ** ROOT LOCUS **          "
30 NORMAL
35 PRINT : PRINT "  PROGRAMA QUE CALCULA Y DIBUJA          (PLANO S Y Z) LAS RAICES DE          UNA FUNCION DE TRANSFE
RENCIA "
40 PRINT : PRINT " * NOTA: DESPUES DE INTRODUCIR CADA          DATO, Y CUANDO APAREZCA EL SIGNO ?          OPRIMA LA TECLA RETURN
"
45 PRINT : PRINT "  PARA CUALQUIER DUDA O ACLARACION          CONSULTE EL MANUAL DE PROGRAMAS"
50 PRINT : INPUT A$
55 PRINT : PRINT "  EL POLINOMIO DEL NUMERADOR ESTARA          EXPRESADO COMO :
      K*(B0*S^M + B1*S^M-1 + ... + BM^M"
60 PRINT : PRINT "  EL POLINOMIO DEL DENOMINADOR ESTARA          EXPRESADO COMO :
      A0*S^N + A1*S^N-1 + ... + AN^N"
65 REM ORDEN DEL NUMERADOR Y DENOMINADOR
70 PRINT : PRINT "  DE EL GRADO DEL NUMERADOR"
75 PRINT : PRINT "      PM= "; INPUT PM
80 PM = INT (PM)
85 PRINT : PRINT "  DE EL GRADO DEL DENOMINADOR"
90 PRINT : PRINT "      PN= "; INPUT PN
95 PN = INT (PN)
100 GOTO 575
105 DIM CB(PM + 1), CA(PN + 1), A(NC), B(NC), P(NC), Q(NC), RL(NC), RG(NC), RZ(PN), IZ(PN), BK(PN + 1), AK(PN + 1), ZR(250), ZI(250)
110 REM COEFICIENTES DEL NUM. Y DEN.
115 PRINT : PRINT "DE LOS COEFICIENTES DEL NUMERADOR EN  ORDEN ASCENDENTE B0,B1,...,BM": PRINT
120 FOR I = 1 TO PM + 1
125 INPUT CB(I)
130 NEXT I
135 PRINT : PRINT "DE LOS COEFICIENTES DEL DENOMINADOR EN  ORDEN ASCENDENTE AN,AN-1,...,A0": PRINT
140 FOR I = 1 TO PN + 1
145 INPUT CA(I)
150 NEXT I
155 GOTO 665
160 REM LIMITES DE LA GANANCIA K
165 PRINT : PRINT "QUIERE TRAZAR EL LUGAR GEOMETRICO EN  EL PLANO (S) O EL EL PLANO (Z) "
170 PRINT : INPUT A$
175 IF A$ = "Z" THEN 910
180 IF A$ < > "S" THEN 165
185 PRINT : PRINT "      RANGO DE LA GANANCIA K"
190 PRINT : PRINT "      LIMITE INFERIOR KI= ";
195 INPUT KI
200 PRINT : PRINT "      LIMITE SUPERIOR KS= ";
205 INPUT KS
210 PRINT : PRINT "      NUMERO DE PUNTOS A EVALUAR NE= ";
215 INPUT NE
220 GOTO 830
225 PRINT : PRINT "  SE ESTA CALCULANDO LAS RAICES "
230 INC = (KS - KI) / NE
235 NS = PN + NE + PN
240 IF NS > 235 + PN THEN 945
245 KA = KI
250 K1 = 1; K2 = PN
255 IF KA : KS + 00001 THEN 1575
260 REM  ARREGLO DEL POLINOMIO PARA EL METODO DE LIN-BAIRSTON
265 FOR I = 1 TO PM + 1
270 BK(I) = KA * CB(I)
275 NEXT I
280 FOR I = 1 TO PN + 1
285 AK(I) = BK(I) + CA(I)

```

```

290 NEXT I
295 IF PN = PM THEN 955
300 FOR I = PM + 2 TO PN + 1
305 AK(I) = CA(I)
310 NEXT I
315 J = PN
320 FOR I = 1 TO PN
325 A(I) = AK(I)
330 J = J - 1
335 NEXT I
340 GOTO 990
345 IF K5 < 1 THEN 430
350 REM CAMBIO DE LAS RAICES AL PLANO ZETA
355 FOR J = 1 TO PN
360 IF RG(J) < 0 THEN 380
365 RZ(J) = EXP (RL(J) * T)
370 NEXT J
375 GOTO 450
380 A1 = - 2 * EXP (RL(J) * T) * COS (RG(J) * T)
385 A2 = EXP (2 * RL(J) * T)
390 RZ(J) = A1 / 2
395 RZ(J + 1) = A1 / 2
400 RA = ABS ((A1 ^ 2) - 4 * A2)
405 RA = SQRT (RA)
410 IZ(J) = RA / 2
415 IZ(J + 1) = - RA / 2
420 J = J + 1
425 GOTO 370
430 FOR I = 1 TO PN
435 RZ(I) = RL(I)
440 IZ(I) = RG(I)
445 NEXT I
450 HOME : INPUT A$
455 PRINT "      *** RAICES ***"
460 PRINT : PRINT "      KA = "KA: PRINT
465 FOR I = 1 TO PN
470 PRINT "RZ("I")= "RZ(I);
475 PRINT "  IZ("I")= "IZ(I)
480 NEXT I
485 INPUT A$
490 L = 1
495 FOR I = K1 TO K2
500 ZR(I) = RZ(L)
505 ZI(I) = IZ(L)
510 L = L + 1
515 NEXT I
520 K1 = K2 + 1:K2 = K2 + PN
525 KA = KA + INC
530 GOTO 255
535 PRINT : PRINT "DESEA OTRO RANGO DE VALORES DE K CON  LOS MISMOS COEFICIENTES ?
      (S,N)"
540 PRINT : INPUT A$
545 IF A$ = "S" THEN 2160
550 IF A$ = "N" THEN 560
555 GOTO 535
560 PRINT : PRINT : PRINT "      *** FIN DEL PROGRAMA ***"
565 END
570 REM PRUEBA Y CORRECCION DEL GRADO DEL NUM. Y DEN.
575 IF PM > PN THEN 595
580 IF PN < = 0 THEN 605
585 IF PM < 0 THEN 615
590 GOTO 620
595 PRINT : PRINT " PM DEBE SER < 0 = A PN"
600 GOTO 620
605 PRINT : PRINT "EL GRADO DEL DENOMINADOR DEBE SER UN  NUMERO ENTERO MAYOR QUE CERO"
610 GOTO 620

```



```

615 PRINT PRINT "EL GRADO DEL NUMERADOR DEBE SER UN      NUMERO ENTERO > 0 = A CERO"
620 PRINT PRINT "DESEA CAMBIAR (PN) O (PM) ? (S/N)"
625 PRINT : INPUT A#
630 IF A# = "S" THEN 70
635 IF PN <= 0 THEN 605
640 IF PM > PM THEN 595
645 IF PM < 0 THEN 615
650 NO = PN
655 GOTO 185
660 REM CORRECCION DE COEFICIENTES
665 PRINT : PRINT " ESTOS SON LOS COEFICIENTES" : PRINT
670 FOR I = 1 TO PM + 1
675 PRINT "      CB("I")= "CB(I)
680 NEXT I
685 PRINT
690 FOR I = 1 TO PN + 1
695 PRINT "      CR("I")= "CR(I)
700 NEXT I
705 PRINT : PRINT "QUIERE CORREGIR ALGUN COEFICIENTE ? S/N"
710 INPUT A#
715 IF A# = "S" THEN 730
720 IF A# = "N" THEN 2150
725 GOTO 705
730 PRINT : PRINT "COEFICIENTE DEL NUMERADOR (CB) O DEL DENOMINADOR (CR)"
735 PRINT : INPUT A#
740 IF A# = "CB" THEN 755
745 IF A# = "CR" THEN 790
750 GOTO 705
755 PRINT : PRINT "DE PRIMERO EL INDICADOR DEL COEFICIENTE ,DESPUES SU NUEVO VALOR"
760 PRINT : INPUT I, CB(I)
765 IF I > PM + 1 THEN 780
770 PRINT : PRINT "      CB("I")= "CB(I)
775 GOTO 705
780 PRINT PRINT " NO EXISTE TAL COEFICIENTE"
785 GOTO 755
790 PRINT : PRINT "DE PRIMERO EL INDICADOR DEL COEFICIENTE ,DESPUES SU NUEVO VALOR"
795 PRINT : INPUT I, CR(I)
800 IF I > PN + 1 THEN 815
805 PRINT : PRINT "      CR("I")= "CR(I)
810 GOTO 705
815 PRINT : PRINT " NO EXISTE TAL COEFICIENTE"
820 GOTO 790
825 REM PRUEBA DE LOS LIMITES DE LA GANANCIA
830 IF KS - KI <= 0 THEN 855
835 IF KI < 0 THEN 855
840 IF KS < 0 THEN 865
845 IF ME <= 0 THEN 875
850 GOTO 895
855 PRINT : PRINT " KS DEBE SER MAYOR QUE KI"
860 GOTO 185
865 PRINT : PRINT " LOS LIMITES DE LA GANANCIA DEBEN SER      POSITIVOS"
870 GOTO 185
875 PRINT : PRINT " ME DEBE SER MAYOR QUE CERO"
880 GOTO 210
885 PRINT : PRINT " QUIERE CAMBIAR KI, KS O ME ?
890 PRINT : INPUT A#
895 IF A# = "S" THEN 185
900 IF A# = "N" THEN 895
905 GOTO 895
910 KS = 1
915 PRINT PRINT " QUE TIEMPO DE MUESTRO SE VA A USAR?"
920 PRINT : PRINT "      T= " : INPUT T
925 IF T < 0 THEN 905
930 GOTO 185
935 PRINT PRINT "EL TIEMPO DE MUESTRO DEBE SER POSITIVO"
940 GOTO 915

```

```

345 PRINT PRINT * EL PRODUCTO PAMME DEBE SER LE A 235 DE MENOS PUNTOS A EVALUAR *
950 GOTO 210
955 CU = AK(PN + 1)
960 FOR I = 1 TO PM + 1
965 AK(I) = AK(I) / CU
970 NEXT I
975 GOTO 215
980 REM CALCULO DE RAICES METODO DE LIN-BEIRSTON
985 REM VALORES INICIALES
990 EPS = .00001
995 IF PN > 5 THEN EPS = .0001
1000 RZ = .5:SZ = 1.5
1005 M = 0:J = 1
1010 R = RZ
1015 S = SZ
1020 L = 0
1025 N = NC - 2 + M
1030 IF (N - 2) < 0 THEN 1050
1035 IF (N - 2) = 0 THEN 1065
1040 IF (N - 2) > 0 THEN 1175
1045 REM CALCULA LA RAIZ DEL FACTOR LINEAL (SI EXISTE)
1050 RL(J) = -A(1)
1055 GOTO 345
1060 REM CALCULA EL PAR DE RAICES DEL ULTIMO TERMINO CUADRATICO
1065 H = A(1) * A(1) - 4 * A(2)
1070 IF H < 0 THEN 1085
1075 IF H = 0 THEN 1120
1080 IF H > 0 THEN 1145
1085 RM = -H
1090 RA = SQR(RM)
1095 RL(J) = -A(1) / 2
1100 RL(J + 1) = RL(J)
1105 RG(J) = RA / 2
1110 RG(J + 1) = -RA / 2
1115 GOTO 345
1120 RL(J) = -A(1) / 2
1125 RL(J + 1) = RL(J)
1130 RG(J) = 0
1135 RG(J + 1) = 0
1140 GOTO 345
1145 RA = SQR(H)
1150 RL(J) = (-A(1) + RA) / 2
1155 RL(J + 1) = (-A(1) - RA) / 2
1160 RG(J) = 0
1165 RG(J + 1) = 0
1170 GOTO 345
1175 REM ENTRADA PARA REDEFINIR R Y S
1180 B(1) = A(1) - R
1185 B(2) = A(2) - R * B(1) - S
1190 FOR K = 3 TO N
1195 B(K) = A(K) - R * B(K - 1) - S * B(K - 2)
1200 NEXT K
1205 RC = B(N - 1)
1210 SC = B(N) + R * B(N - 1)
1215 REM CALCULA LAS PARCIALES DE B(K) W.R.T.R
1220 P(1) = -1
1225 P(2) = R - B(1)
1230 FOR K = 3 TO N
1235 P(K) = -B(K - 1) - R * P(K - 1) - S * P(K - 2)
1240 NEXT K
1245 RR = P(N - 1)
1250 SR = P(N) + R * P(N - 1) + B(N - 1)
1255 REM CALCULA LAS DERIVADAS PARCIALES DE B(K) W.R.T.S
1260 Q(1) = 0
1265 Q(2) = -1
1270 FOR K = 3 TO N

```

```

1275 Q(K) = -B(K - 2) - R * Q(K - 1) - S * Q(K - 2)
1280 NEXT K
1285 RS = Q(N - 1)
1290 SS = Q(N) + R * Q(N - 1)
1295 REM RESOLVIENDO LA ECUACION PARA LA ECUACION RECURSIVA
1300 DE = RR * SS - RS * SR
1305 RN = -RC + SS + SC * RS
1310 SN = -RR * SC + SR * RC
1315 IF DE = 0 THEN 1405
1320 DR = RN / DE
1325 DS = SN / DE
1330 REM CALCULA LOS NUEVOS VALORES R, S
1335 R = R + DR
1340 S = S + DS
1345 REM PRUEBA DE CONVERGENCIA PARA LA CORRECCION
1350 G = ABS(DR) - EPS
1355 IF G <= 0 THEN 1365
1360 IF G > 0 THEN 1385
1365 G2 = ABS(DS) - EPS
1370 IF G2 <= 0 THEN 1425
1375 IF G2 > 0 THEN 1385
1380 REM PRUEBA DE ITERACION
1385 IF L - 100 <= 0 THEN 1395
1390 IF L - 100 > 0 THEN 1410
1395 L = L + 1
1400 GOTO 1180
1405 RZ = RZ + .5: S2 = S2 + .2: GOTO 1005
1410 PRINT "
      N= "N
1415 GOTO 560
1420 REM CALCULA EL PAR DE RAICES
1425 G3 = R * R - 4 * S
1430 IF G3 < 0 THEN 1445
1435 IF G3 = 0 THEN 1430
1440 IF G3 > 0 THEN 1505
1445 RM = -G3
1450 RA = SQR(RM)
1455 RL(J) = -R / 2
1460 RL(J + 1) = -R / 2
1465 RG(J) = RA / 2
1470 RG(J + 1) = -RA / 2
1475 GOTO 1540
1480 RL(J) = -R / 2
1485 RL(J + 1) = -R / 2
1490 RG(J) = 0
1495 RG(J + 1) = 0
1500 GOTO 1540
1505 RA = SQR(G3)
1510 RL(J) = (-R + RA) / 2
1515 RL(J + 1) = (-R - RA) / 2
1520 RG(J) = 0
1525 RG(J + 1) = 0
1530 GOTO 1540
1535 REM CAMBIO DEL POLINOMIO ORIGINAL AL POLINOMIO REDUCIDO
1540 N = N + 1
1545 J = J + 2
1550 NE = NC - 2 * M
1555 FOR K = 1 TO NE
1560 A(K) = E(K)
1565 NEXT K
1570 GOTO 1010
1575 REM DETERMINACION DE LA RAIZ MAS GRANDE PARA EFECTO DE ESCALAMIENTO
1580 T1 = ABS(ZR(1))
1585 FOR I = 1 TO N5
1590 IF T1 < ABS(ZR(I)) THEN T1 = ABS(ZR(I))
1595 NEXT I

```

NO CONVERGE EN EL POLINOMIO REDUCIDO

```

1500 FOR J = 1 TO NS
1505 IF T1 < ABS(ZI(J)) THEN T1 = ABS(ZI(J))
1510 NEXT J
1515 IF T1 = 0 THEN T1 = 1
1520 R = 75 / T1
1525 REM TRAZO DE LOS EJES COORDENADOS
1530 IF R > 75 THEN R = 75
1535 HGR2
1540 HCOLOR= 3
1545 HPLLOT 4,80 TO 275,80
1550 HPLLOT 140,4 TO 140,155
1555 REM ORIGEN (0,0)=(140,80)
1560 X0 = 140
1565 Y0 = 80
1570 REM TRAZO DEL CIRCULO UNITARIO
1575 FOR Y = 0 TO R
1580 X = SQR((R ^ 2) + 0.001) - Y ^ 2)
1585 X1 = X0 - X
1590 NX = X0 + X
1595 NY = Y0 + Y
1700 MY = Y0 - Y
1705 HPLLOT X1,NY
1710 HPLLOT NX,NY
1715 HPLLOT X1,MY
1720 HPLLOT NX,MY
1725 NEXT Y
1730 REM FACTOR DE ESCALAMIENTO = R , TRAZO DE RAICES
1735 FOR I = 1 TO NS
1740 IF ZR(I) = 0 THEN 1910
1745 IF ZI(I) < > 0 THEN 1860
1750 HCOLOR= 0
1755 REM TRAZO DE RAICES REALES PURAS
1760 XZ = ZR(I) * R + X0
1765 IF XZ > 275 THEN 1950
1770 IF XZ < 4 THEN 1955
1775 HPLLOT XZ,Y0
1780 NEXT I
1785 PRINT : INPUT A#
1790 TEXT : HOME
1795 PRINT : PRINT "SI EL CIRCULO UNITARIO NO APARECE INDICA QUE EXISTEN RAICES MUY GRANDES "
1800 PRINT : PRINT "LAS CRUCES INDICAN QUE LAS RAICES ESTAN FUERA DE ESCALA,ESCEPTO CUANDO SE HALLAN EN EL CIRCULO UNIT
ARIO"
1805 PRINT : PRINT " QUIERE CAMBIAR DE ESCALA ? (S/N)"
1810 PRINT : INPUT A#
1815 IF A# = "S" THEN 1830
1820 IF A# = "N" THEN 525
1825 GOTO 1805
1830 PRINT : PRINT "INDIQUE EL FACTOR DE ESCALAMIENTO PUEDE SER UN NUMERO FRACCIONARIO O ENTERO MAYOR QUE CERO"
1835 PRINT : PRINT " FE = ",
1840 INPUT FE
1845 IF FE < = 0 THEN 1830
1850 R = R * FE
1855 GOTO 1630
1860 HCOLOR= 3
1865 REM TRAZO DE RAICES COMPLEJAS
1870 XZ = ZR(I) * R + X0
1875 YZ = - ZI(I) * R + Y0
1880 IF YZ > 275 THEN 1995
1885 IF YZ < 4 THEN 2015
1890 IF YZ > 155 THEN 2025
1895 IF YZ < 4 THEN 2070
1900 HPLLOT XZ,YZ
1905 GOTO 1790
1910 HCOLOR= 0
1915 REM TRAZO DE RAICES IMAGINARIAS PURAS
1920 YZ = - ZI(I) * R + Y0

```

```

1905 IF YZ > 155 THEN 2110
1930 IF YZ < 4 THEN 2115
1925 HPLLOT X0,YZ
1940 GOTO 1780
1945 REM TRAZO DE RICES REALES PURAS QUE SALEN FUERA DE ESCALA
1950 XZ = 270: GOTO 1960
1955 XZ = 1
1960 HCOLOR= 3
1965 HPLLOT XZ + 1,Y0
1970 HPLLOT XZ - 1,Y0
1975 HPLLOT XZ,Y0 + 1
1980 HPLLOT XZ,Y0 - 1
1985 GOTO 1780
1990 REM TRAZO DE RAICES COMPLEJAS QUE SALEN FUERA DE ESCALA
1995 XZ = 270
2000 IF YZ > 155 THEN 2025
2005 IF YZ < 4 THEN 2030
2010 GOTO 2075
2015 XZ = 1
2020 GOTO 2000
2025 YZ = 150: GOTO 2075
2030 YZ = 1: GOTO 2075
2035 YZ = 155
2040 IF XZ > 255 THEN 2055
2045 IF XZ < 4 THEN 2065
2050 GOTO 2075
2055 XZ = 270: GOTO 2075
2060 YZ = 155: GOTO 2075
2065 XZ = 4: GOTO 2075
2070 YZ = 4: GOTO 2040
2075 HCOLOR= 3
2080 HPLLOT XZ + 1,YZ
2085 HPLLOT XZ - 1,YZ
2090 HPLLOT XZ,YZ + 1
2095 HPLLOT XZ,YZ - 1
2100 GOTO 1780
2105 REM TRAZO DE LAS RAICES IMAGINARIAS PURAS QUE SALEN FUERA DE ESCALA
2110 YZ = 150: GOTO 2120
2115 YZ = 1
2120 HCOLOR= 3
2125 HPLLOT X0,YZ + 1
2130 HPLLOT X0,YZ - 1
2135 HPLLOT X0 + 1,YZ
2140 HPLLOT X0 - 1,YZ
2145 GOTO 1780
2150 IF PM = 0 THEN CB(1) = 1
2155 GOTO 165
2160 K5 = 0: GOTO 165

```

```

)PRINT"EJEMPLO"

```

```

EJEMPLO
)RUN

```

```

** ROOT LOCUS **

```

```

PROGRAMA QUE CALCULA Y DIBUJA
* NOTA: DESPUES DE INTRODUCIR CADA
PARA CUALQUIER DUDA O ACLARACION

```

```

(PLANO S Y Z) LAS RAICES DE
DATO, Y CUANDO APAREZCA EL SIGNO ?
CONSULTE EL MANUAL DE PROGRAMAS

```

```

UNA FUNCION DE TRANSFERENCIA
OPRIMA LA TECLA RETURN

```

```

EL POLINOMIO DEL NUMERADOR ESTARA EXPRESADO COMO :
(B0*S^M + B1*S^M-1 + ... + Bn)

```

```

EL POLINOMIO DEL DENOMINADOR ESTARA EXPRESADO COMO :
(A0*S^N + A1*S^N-1 + ... + An)

```

```

DE EL GRADO DEL NUMERADOR

```

```

PM= 0

```

```

DE EL GRADO DEL DENOMINADOR

```

```

PN= 0

```

```

TECLA CAMBIA (M) O (N) O (P) O (S) O (Z)

```

?N  
 DE LOS COEFICIENTES DEL NUMERADOR EN ORDEN ASCENDENTE B1,BM-1,...,00  
 ?1  
 DE LOS COEFICIENTES DEL DENOMINADOR EN ORDEN ASCENDENTE AN,AN-1,...,A0  
 ?0  
 ?2  
 ?3  
 ?1

ESTOS SON LOS COEFICIENTES

CB(1)= 1  
 CR(1)= 0  
 CR(2)= 2  
 CR(3)= 3  
 CR(4)= 1

QUIERE CORREGIR ALGUN COEFICIENTE ? S/N

?N

QUIERE TRAZAR EL LUGAR GEOMETRICO EN EL PLANO (S) O EL EL PLANO (Z)

?S

RANGO DE LA GANANCIA K

LIMITE INFERIOR KI= ?0

LIMITE SUPERIOR KS= ?10

NUMERO DE PUNTOS A EVALUAR NE= ?10

QUIERE CAMBIAR KI,KS O NE ?

(S, N)

?N

SE ESTA CALCULANDO LAS RAICES

?

\*\*\* RAICES \*\*\*

KA = 0

RZ(1)= 0 IZ(1)= 0

RZ(2)= -1 IZ(2)= 0

RZ(3)= -2.00000338 IZ(3)= 0

?

?

\*\*\* RAICES \*\*\*

KA = 1

RZ(1)= -.337641021 IZ(1)= .562279512

RZ(2)= -.337641021 IZ(2)= -.562279512

RZ(3)= -2.32471907 IZ(3)= 0

?

?

\*\*\* RAICES \*\*\*

KA = 2

RZ(1)= -.239310147 IZ(1)= .057873627

RZ(2)= -.239310147 IZ(2)= -.057873627

RZ(3)= -2.5213834 IZ(3)= 0

?

?

\*\*\* RAICES \*\*\*

KA = 3

RZ(1)= -.164150059 IZ(1)= 1.04686932

RZ(2)= -.164150059 IZ(2)= -1.04686932

RZ(3)= -2.67169988 IZ(3)= 0

?

?

\*\*\* RAICES \*\*\*

KA = 4

RZ(1)= -.101839048 IZ(1)= 1.1916708

RZ(2)= -.101839048 IZ(2)= -1.1916708

RZ(3)= -2.79632191 IZ(3)= 0

?

?

\*\*\* RAICES \*\*\*

KA = 5

RZ(1)= -.0479195704 IZ(1)= 1.31124804

RZ(2)= -.0479195704 IZ(2)= -1.31124804

RZ(3)= -2.90416141 IZ(3)= 0

\*\*\* RAICES \*\*\*

KA = 6

RZ(1)= 1.91120405E-11 IZ(1)= 1.41421356  
RZ(2)= 1.91120405E-11 IZ(2)= -1.41421356  
RZ(3)= -7.000000528 IZ(3)= 0

\*\*\* RAICES \*\*\*

KA = 7

RZ(1)= .0433726699 IZ(1)= 1.50528389  
RZ(2)= .0433726699 IZ(2)= -1.50528389  
RZ(3)= -3.09674534 IZ(3)= 0

\*\*\* RAICES \*\*\*

KA = 8

RZ(1)= .0831563737 IZ(1)= 1.587351  
RZ(2)= .0831563737 IZ(2)= -1.587351  
RZ(3)= -3.16631275 IZ(3)= 0

\*\*\* RAICES \*\*\*

KA = 9

RZ(1)= .120820494 IZ(1)= 1.66232901  
RZ(2)= .120820494 IZ(2)= -1.66232901  
RZ(3)= -3.24084099 IZ(3)= 0

\*\*\* RAICES \*\*\*

KA = 10

RZ(1)= .15445366 IZ(1)= 1.73155703  
RZ(2)= .15445366 IZ(2)= -1.73155703  
RZ(3)= -3.30890734 IZ(3)= 0

?  
?  
? SI EL CIRCULO UNITARIO NO APARECE INDICA QUE EXISTEN RAICES MUY GRANDES  
LAS CRUCES INDICAN QUE LAS RAICES ESTAN FUERA DE ESCALA, EXCEPTO CUANDO SE QUIERE CAMBIAR DE ESCALA ?

HALLAN EN EL CIRCULO UNITARIO  
(S,N)

?N  
DESEA OTRO RANGO DE VALORES DE K CON LOS MISMOS COEFICIENTES ?  
(S,N)

?NS  
QUIERE TRAZAR EL LUGAR GEOMETRICO EN EL PLANO (S) O EL EL PLANO (2)  
?Z

QUE TIEMPO DE MUESTREO SE VA A USAR?

T= 21

RANGO DE LA GANANCIA K

LIMITE INFERIOR KI= 10

LIMITE SUPERIOR KS= 210

NUMERO DE PUNTOS A EVALUAR NE= 25

QUIERE CAMBIAR KI, KS O NE ?

(S,N)

?N  
SE ESTA CALCULANDO LAS RAICES  
?

\*\*\* RAICES \*\*\*

KA = 0

RZ(1)= 1 IZ(1)= 1.73155703  
RZ(2)= .267879441 IZ(2)= -1.73155703  
RZ(3)= .135334826 IZ(3)= 0

\*\*\* RAICES \*\*\*

KA = 2

RZ(1)= .514846991 IZ(1)= .59549096

RZ(2)= .514246391 IZ(2)= - 595459056  
RZ(3)= .0903463762 IZ(3)= 0

61

\*\*\* RAICES \*\*\*

KA = 4

RZ(1)= -.234272475 IZ(1)= 839039229  
RZ(2)= -.234272475 IZ(2)= - 839039229  
RZ(3)= .0610341395 IZ(3)= 0

\*\*\* RAICES \*\*\*

KA = 6

RZ(1)= -.155943695 IZ(1)= 987765946  
RZ(2)= -.155943695 IZ(2)= - 987765946  
RZ(3)= .0497368008 IZ(3)= 0

\*\*\* RAICES \*\*\*

KA = 8

RZ(1)= .017909333 IZ(1)= 1.09656282  
RZ(2)= .017909333 IZ(2)= -1.09656282  
RZ(3)= .0421587616 IZ(3)= 0

\*\*\* RAICES \*\*\*

KA = 10

RZ(1)= .106803932 IZ(1)= 1.15197241  
RZ(2)= .106803932 IZ(2)= -1.15197241  
RZ(3)= .0365560955 IZ(3)= 0

?  
?  
SI EL CIRCULO UNITARIO NO APARECE INDICA QUE EXISTEN RAICES MUY GRANDES  
LAS CRUCES INDICAN QUE LAS RAICES ESTAN FUERA DE ESCALA, ESCEPTO CUANDO SE HALLAN EN EL CIRCULO UNITARIO  
QUIERE CAMBIAR DE ESCALA ? (S, N)

?  
DESEA OTRO RANGO DE VALORES DE K CON LOS MISMOS COEFICIENTES ?  
(S, N)

?  
\*\*\* FIN DEL PROGRAMA \*\*\*



## BIBLIOGRAFIA

- 1.-"Digital Control of Dinamic Systems"  
Franklin & Powell
- 2.-"Digital Computers Process Control"  
Smith G . L.
- 3.-"Linear Systems"  
Thomas Kailath
- 4.-"Numerical Methods with Fortran"  
Richard MacCalla
- 5.-"Numerical Methods"  
Dahlquist & Björck
- 6.-"Discrete-Time and Computer Control Systems"  
James A. Cadzow & Hinrich R . Martens
- 7.-"Digital and Sampled Data Control "  
Tou Joulins T.
- 8.-"Sampled-Data Control Systems"  
John R . Ragassini.
- 9.-"Calculus"  
Tom M. Apostol.VOL. II

MEXICO 21 , CIUDAD UNIVERSITARIA 13 de Agosto de 1982