

Para los siguientes ejercicios diseñe los algoritmos en pseudocódigo, diagrama de flujo, diagrama NS y muestre el ambiente.

CE07A Analizar, hacer la prueba de escritorio para encontrar la raíz de la ecuación cúbica $y = x^3 + 6x^2 - x - 61.8$ la raíz debe estar en un intervalo $[a,b]$ donde el signo de $y(a)$ es distinto del signo de $y(b)$. Para nuestro ejemplo el intervalo $[a,b]$ puede ser $[0,8]$ y error deseado para la raíz menor a 0.1% .

Manual	Automático
<p><u>Algoritmo</u> Raiz</p> <p><u>Variables</u></p> <p>Real: a, b, ErA</p> <p>caracter: Op</p> <p><u>Inicio</u></p> <p><u>Repetir</u></p> <p><u>Escribir</u> 'Proponer valor de x:' // $[0,8]$</p> <p><u>Leer</u> x</p> <p>$ErA \leftarrow x^3 + 6x^2 - x - 61.8$</p> <p><u>Escribir</u> 'Error absoluto:', ErA</p> <p><u>Escribir</u> '¿Mas iteraciones? S/N:'</p> <p><u>Leer</u> Op</p> <p><u>Hasta que</u> Op='n'</p> <p><u>Escribir</u> 'Raíz aproximada = ', x</p> <p><u>Fin.</u></p>	<p><u>Algoritmo</u> Raiz</p> <p><u>Variables</u></p> <p>Real: a, b, m, m1, m2, ErRP, ErRC, fm</p> <p><u>Inicio</u></p> <p><u>Escribir</u> 'Ingrese intervalo a y b y error propuesto'</p> <p><u>Leer</u> a, b, ErRP</p> <p>$m \leftarrow (a + b) / 2$</p> <p><u>Repetir</u></p> <p>$fa \leftarrow a^3 + 6 * a^2 - a - 61.8$</p> <p>$fm \leftarrow m^3 + 6 * m^2 - m - 61.8$</p> <p><u>si</u> $fa * fm > 0$</p> <p><u>entonces</u></p> <p>$a \leftarrow m$</p> <p><u>sino</u></p> <p>$b \leftarrow m$</p> <p><u>fin</u></p> <p>$m1 \leftarrow m$</p> <p>$m \leftarrow (a + b) / 2$</p> <p>$m2 \leftarrow m$</p> <p>$ErRC \leftarrow \text{Abs}((m1 - m2)/m1) * 100$</p> <p><u>Hasta que</u> $ErRC < ErRP$</p> <p><u>Escribir</u> 'Raíz aproximada= ', m</p> <p><u>Fin.</u></p>

CE07B Analizar, hacer la prueba de escritorio y ejecutar el siguiente algoritmo para aproximar la solución de un sistema de ecuaciones mediante iteraciones sucesivas. Responder: ¿En cuántas iteraciones converge con un error relativo menor al 1%? ¿Qué ventajas tiene el método de Gauss Seidel respecto de otros métodos?

Sistema de ecuaciones	PSeInt
$3X_1 - 0.2X_2 - 0.5X_3 = 8$ $0.1X_1 + 7X_2 + 0.4X_3 = -19.5$ $0.4X_1 - 0.1X_2 + 10X_3 = 72.4$	<p>Proceso Sistema_lineal_Gauss_Seidel</p> <p>Escribir 'Ingrese el error relativo deseado';</p> <p>Leer ErD; // ingresar 1 que corresponde al 1%</p> <p>$X1 \leftarrow 0;$</p> <p>$X2 \leftarrow 0;$</p> <p>$X3 \leftarrow 0;$</p> <p><u>Repetir</u></p> <p>$X1a \leftarrow X1;$</p> <p>$X2a \leftarrow X2;$</p> <p>$X3a \leftarrow X3;$</p> <p>$X1 \leftarrow (8 + 0.2 * X2 + 0.5 * X3) / 3;$</p> <p>$X2 \leftarrow (-19.5 - 0.1 * X1 - 0.4 * X3) / 7;$</p> <p>$X3 \leftarrow (72.4 - 0.4 * X1 + 0.1 * X2) / 10;$</p> <p>$ErrRelX1 \leftarrow \text{Abs}(X1 - X1a) / X1 * 100;$</p> <p>$ErrRelX2 \leftarrow \text{Abs}(X2 - X2a) / X2 * 100;$</p> <p>$ErrRelX3 \leftarrow \text{Abs}(X3 - X3a) / X3 * 100;$</p> <p><u>Hasta Que</u> $(ErrRelX1 < ErD) \& (ErrRelX2 < ErD) \& (ErrRelX3 < ErD)$</p> <p>Escribir 'Soluciones';</p> <p>Escribir 'X1=', X1;</p> <p>Escribir 'X2=', X2;</p> <p>Escribir 'X3=', X3;</p> <p>FinProceso</p>

1. Analizar el algoritmo N_Primos que lee N números enteros X, que se eligen de un intervalo al azar entre (-100000,100000) y se valida que los números ingresados X sean mayores que cero. Calcular y mostrar si es primo o no lo es, también se desea saber cuántos números ingresados No son primos.

N_Primos	PruebaDeEscritorio1
<pre> Proceso N_Primos Escribir 'Ingrese N' Leer N CNOP <- 0 // contador de n° No primos Para Cont<-1 Hasta N Con Paso 1 Hacer Repetir //Escribir 'Ingrese un número X' //Leer X X <- (-1)^(Azar(2)+1)*Azar(100000) Hasta Que X>0 // Primo es_primo <- Verdadero i <- 2 AuxX <- RAIZ(X) // Eratóstenes Mientras (i < AuxX) Y es_primo Hacer Si (X mod i) = 0 Entonces es_primo <- Falso Sino i <- i + 1 Fin Si Fin Mientras Si es_primo Entonces Escribir X, ' es primo' Sino CNOP <- CNOP + 1 Fin Si Fin Para Escribir 'Cantidad de NO primos ', CNOP FinProceso </pre>	<pre> Proceso pruebaDeEscritorio1 Definir a como Real; Definir continua como Logico; Definir contador1, contador2 como Entero; a <- 100; contador1 <- 0; contador2 <- 0; continua <- Verdadero; Mientras continua Hacer Si a / 2 > 10 Entonces a <- a / 2; contador1 <- contador1 + 1; Sino a <- a - 2; contador2 <- contador2 + 1; FinSi continua <- a > 0; FinMientras Escribir 'Contador1: ', contador1; Escribir 'Contador2: ', contador2; FinProceso </pre>

2. Realice la prueba de escritorio del algoritmo PruebaDeEscritorio1 y determine cuánto valen los contadores al final del proceso.
3. Calcular la nota promedio de un salón de clase y la nota mayor y el código del estudiante que la obtuvo, validar que las notas ingresadas se encuentren en el intervalo [0,10]. El ingreso de notas finaliza a pedido del operador.
4. Ingresar repetidamente valores de N (entero), calcular y mostrar $S = \sum_{i=1}^N (-1)^i i!$ hasta que se ingresa N=0. Mostrar el promedio de los valores de S. Hacer la prueba de escritorio para N=6 (Respuesta S=619)
5. Se tienen registros con los datos de los estudiantes de una universidad: código del estudiante, valor por materia, número de materias a matricular, valor del recargo por matrícula Se requiere leer los registros (suponga un código de 0 como condición de terminación) e imprimir un listado con la siguiente información por estudiante: código, valor por materia, número de materias, valor del recargo, total matrícula; al final se debe mostrar el total a recibir por matrículas y el número total de estudiantes matriculados.
6. Un supermercado el día de hoy promociona el nuevo producto XX, al inicio de la jornada se debe ingresar el stock inicial de este producto y el precio correspondiente. Durante toda la jornada se realiza la venta a los clientes de este producto y pueden pagar al contado o con tarjeta de crédito. Cuando pagan al contado se les realiza un 10% de descuento y si el pago es con tarjeta se efectúa un recargo del 5%. Antes de realizar una venta, se debe mostrar cuantos productos quedan en stock. Cuando se finaliza la venta del producto a un cliente se debe mostrar el total que debe pagar. Se tiene que controlar que la cantidad solicitada por cada cliente del producto XX no supere la cantidad existente en el stock, caso contrario se mostrará un mensaje notificando esto. Las ventas finalizan cuando el stock es cero o cuando el vendedor no desee realizar más ventas. Al finalizar las ventas se debe mostrar la cantidad recaudada por todas las ventas realizadas, la cantidad de clientes atendidos y el stock disponible.

7. Un galpón tiene al comienzo de la jornada una cantidad inicial (Stock inicial) de cajones con productos de un solo tipo, luego repetidamente, entran y salen camiones, que traen o llevan cantidades de cajones. Si no alcanza la cantidad a llevar, se debe mostrar un mensaje "NO ALCANZA", se lleva todo lo que hay; se muestra lo que se lleva y el galpón queda vacío. Se muestra al final de la jornada cuantos cajones hay en el galpón (Stock final), y cuantos cajones ingresaron y cuantos cajones salieron en toda la jornada. Ejemplo:

Stock inicial 1000

1-ingresa 2-sale cantidad 1 200
¿Desea continuar? S/N S
1-ingresa 2-sale cantidad 2 1500
NO ALCANZA se lleva 1200
¿Desea continuar? S/N S
1-ingresa 2-sale cantidad 1 500
¿Desea continuar? S/N S
1-ingresa 2-sale cantidad 2 100
¿Desea continuar? S/N N
Stock final 400
Ingresaron 700
Salieron 1300

8. Haga la prueba de escritorio (traza) del siguiente algoritmo ingresando: V, SEL y muestre claramente los valores finales de S y P.

Algoritmo ej08TP07

variable

Entero: V,P,i,k,SEL

Lógico: OK

Real: S

inicio

P ← 1

S ← 0.0

repetir

escribir 'Ingrese V, SEL'

leer V,SEL

OK ← (V>0) Y (SEL<10)

si OK entonces

según SEL hacer

1:

i ← 1

repetir

S ← S + V*i

i ← i + 1

hasta que S > 20

P ← P * i

2:

para k desde 1 hasta V con paso 3 hacer

S ← S + V

finpara

P ← P * k

de Otro Modo

S ← S + V

finSegun

finsi

hasta que NO OK

escribir S, P

fin

V	SEL
4	1
11	0
7	2
3	6
9	13