Ejercico N1-

1.- Dada una secuencia de enteros positivos terminada en cero, elabore un algoritmo que:

a) Calcule el porcentaje de números impares y el porcentaje de números pares.

b) Calcule la cantidad de valores iguales a un valor N dado por el usuario.

c) El número mayor y el menor de la secuencia

d) Contar la cantidad de números primos.

Análisis del problema

|  |  |  |
| --- | --- | --- |
| Entrada | Proceso | Salida |
| * La secuencia de números que ingresa el usuario (num) * Un valor N ingresado por el usuario (N) | * Pedir el dato de entrada * VerSiicar y contar los números pares e impares (P\_Pares, P\_Impares) * VerSiicar la cantidad de números iguales al valor N (N\_Iguales) * VerSiicar si el numero es el mayor o menor de la secuencia y asignarlos a una variable de ser el caso (Mayor, Menor) * VerSiicar y contar los números primos (Primos) * Mostrar los datos de salida | * El porcentaje de números pares e impares ingresados (P\_Pares, P\_Impares) * La cantidad de valores iguales al numero N (N\_Iguales) * El numero mayor de toda la secuencia y el numero menor de toda la secuencia (Mayor, Menor) * La cantidad de números primos de la secuencia (Primos) |

Algoritmo en pseudocódigo

Programa SecuenciaDeEnterosPositivos;

Variables

j, k, i, Num, C\_Primos, Primos, N, N\_Iguales, mayor, menor: Entero;

p\_pares, p\_impares: Real;

Inicio

p\_pares= 0;

p\_impares= 0;

i= 0;

Primos= 0;

Mayor= 0;

Menor= 2;

N\_Iguales= 0;

j= 0;

Escribir(“Ingrese un valor cualquiera que sea entero y positivo”);

Leer(N);

Escribir(“Se va a comenzar a recibir la secuencia de números enteros positivos, y no se va a terminar hasta que se ingrese un número "0"“);

Escribir(“Ingrese los valores deseados”);

Repetir

Leer(Num);

Si (Num<>0) Entonces

k=0;

i=i+1;

Si (Num MOD 2 = 0) Entonces

P\_pares=P\_pares+1

Sino

P\_Impares=P\_impares+1;

Fin-Si;

Si (Num=N) Entonces

N\_Iguales=N\_Iguales+1;

Fin-Si;

Si (Num>Mayor) Entonces

Mayor=Num;

Si (Num<menor) o (i=1) Entonces

Menor=Num;

Para j=num, 1, -1 haga

Inicio

Si (Num mod j =0) Entonces

k=k+1;

Fin-Si;

Fin;

Si (k=2) Entonces

Primos=Primos+1;

Fin-Si;

Fin;

Hasta que (Num=0);

P\_Pares=P\_Pares/i\*100;

P\_Impares=P\_Impares/i\*100;

Escribir(““);

Escribir(“El porcentaje de numeros impares es de: “,P\_Impares,”%”);

Escribir(“El porcentaje de numeros pares es de: “,P\_pares,”%”);

Escribir(“La cantidad de valores iguales al valor ingresado “,N,” es de: “,N\_Iguales);

Escribir(“El número mayor de la secuencia es: “,Mayor);

Escribir(“El numero menor de la secuencia es: “,menor);

Escribir(“La cantidad de numeros primos ingresados fue de: “,Primos);

Fin.

Ejercicio N2-

2.-Construya un algoritmo que calcule el factorial de un número entero positivo N. La función factorial, representada por N!, es ampliamente utilizada, y se especifica que N!= N\*(N-1)!, y además 0!=1. Implemente el mismo programa utilizando un menú que escoja la forma de calcular el factorial entre un repita mientras, un repita hasta o un repita para

|  |  |  |
| --- | --- | --- |
| Entrada | Proceso | Salida |
| * El numero al cual se le quiere realizar el factorial (N) | * Pedir el dato de entrada (N) * Preguntar con qué estructura repetitiva se quiere realizar el factorial a través de un menú * Verificar si el numero ingresado es 0, y si lo es, se conoce que el factorial es 1 * Calcular el factorial del número ingresado (Fact) * Mostrar el resultado | * El factorial del numero ingresado (Fact) |

Algoritmo en Pseudocódigo

Programa: Factorial;

Variables

N, Fact, i, menu: Entero;

Inicio

i=1;

Escribir(“Este programa calcula el factorial de un numero entero positivo”);

Escribir(“Ingrese el numero.”);

Leer(N);

Fact=N;

Escribir(“Elija que estructura repetitiva quiere que se use para calcule el factorial”);

Escribir(“1: Repita mientras; 2: Repita Hasta; 3: Repita para”);

Leer(menu);

Si (N=0) Entonces

Inicio

Fact=1;

menu=0;

Fin-si;

Caso menu de

1: Inicio //Caso R.mientras

Repita mientras (i<N)

fact=Fact\*i;

i=i+1;

Fin-Mientras;

Fin;

2: Inicio //Caso R.Hasta

Repita

fact=Fact\*i;

i=i+1;

Hasta que (i=N);

Fin;

3: Inicio //Caso R.Para

Para i=1, (N-1), 1 haga

Inicio

fact=Fact\*i;

Fin-Para;

Fin;

Fin;

Escribir(“El factorial del numero “,N,” es igual a: “,Fact);

Escribir(“Es decir; “,N,”! = “,Fact);

Fin.

Ejercicio N3-

3.- La expresión para un número x cualquiera se puede estimar hacSiino uso de la siguiente serie:

= 1 + x +  *;* La cantidad de términos a sumar depFine de la precisión que se desee, que se especificacuando el valor del último término generado sea menor que un valor cualquiera.

Escriba un programa que dado un valor X y el valor de N calcule el valor de ex

Nota : Recuerde que el signo “!” representa el factorial del número.

Análisis del problema

|  |  |  |
| --- | --- | --- |
| Entrada | Proceso | Salida |
| * El exponente del número de Euler (X) * La cantidad de términos que se van a sumar (n) | * Se piden los datos de entrada * Se calcula el exponente de x a cada numeración del término para dividirse entre el factorial de dicha numeración (Aprox) * Se muestra el resultado después de todas las iteraciones pedidas. | * La aproximación al numero de Euler por el exponente dado (Aprox) |

Algoritmo en pseudocódigo

Programa: ExponenteDe\_euler;

Variables

i, k, q, n, x: Entero;

fact, aprox: Real;

Inicio

Escribir(“introduzca el exponente al que quiere elevar la constante de Euler "e"“);

Leer(x);

Escribir(“introduzca el número de veces que quiere repetir la secuencia para calcular la potencia. A mayor cantidad, mas precisión en el cálculo”);

Leer(n);

aprox=1;

Para i=1 , n, 1 haga

Inicio

fact=1;

Para k=1, I, 1 Haga

Inicio

fact=fact\*k;

Fin-Para;

aprox= aprox+((exponente de (i \* logaritmo de (x)))/fact);

Fin;

Escribir(aprox);

Fin.