

Práctica 3 – Estructura Iterativa

Temas

- Estructura iterativa o ciclo.
- Ciclos incondicionales (for) y condicionales (while - repeat).
- Ciclos anidados.

En los ejercicios donde se describe una problemática, implementar la solución en un programa Pascal eficaz, claro y sin pasos innecesarios.

Ej 1) Dados N números enteros, calcular e informar:

- a) El máximo, qué lugar ocupa la primera aparición y cuántas ocurrencias tiene.
- b) El mínimo de los impares (informar si no existe)

Ejemplo: N = 10, 78, 45, 80, 56, 80, 32, 90, 89, 90, 45

- a) 90 - 7º lugar - 2 ocurrencias
- b) 45

Ej 2) Se debe liquidar el sueldo de N empleados, de ellos se sabe el nombre y la antigüedad en años. Todos parten de un básico de \$15000 y la antigüedad se calcula según la siguiente tabla:

- Hasta 5 años 5%
- De 6 a 10 años..... 8%
- De 11 a 15 años..... 12%
- Más de 15 años 20%

Además se descuenta un 11% por cargas sociales y \$500 por seguro obligatorio.

Calcular e informar:

- a) Nombre y sueldo de cada empleado.
- b) Nombre del empleado que más cobra (suponer que es único).
- c) Monto del sueldo promedio.

Ej 3) Ingresar N números enteros, informar cuántas veces un número supera al anterior.

Ej 4) Ingresar N números enteros (ordenados en forma descendente), informar cual es la máxima diferencia entre dos números consecutivos y qué posición ocupa ese par de números dentro de la secuencia.

Ejemplo: 25 19 11 9 3 -3, la máxima diferencia es 8 y corresponde al par 2

Nota: La diferencia se calcula sobre un par de números, por lo tanto deben estar en memoria dos valores consecutivos (Pri y Seg), que se actualizan en cada repetición.

Ej 5) Generar y mostrar los múltiplos de K menores que un valor Q. (K y Q se leen de teclado).

Ej 6) Se conoce el saldo inicial de una cuenta bancaria y una serie de movimientos realizados, por cada uno:

- código de movimiento (D=depósito, R=retiro ó F= fin de datos).
- monto

Escribir un programa que ingrese dichos datos y determine el saldo exacto de la cuenta después de procesar las transacciones. Al final del proceso indique cuantas veces no pudo retirar dinero por insuficiencia de fondos.

Ej 7) Dado un conjunto de alumnos, no se sabe cuántos son, de cada uno de ellos se conoce:

- Nombre (**** indica fin de datos)
- Nota1, Nota2 y Nota3

Ingresar la información e informar:

- a) El promedio de cada alumno y su condición:

- 'Aprobado' si su promedio es mayor o igual a 4
- 'Desaprobado' si su promedio es menor a 4.

b) Porcentaje de alumnos Aprobados

Ej 8) Dado el siguiente programa Pascal indicar qué realiza y cuál es la salida, para ello grabar en un archivo de texto, un número por línea y luego ejecutar el programa.

```

Program Ejercicio8;                                Lote de prueba: 3, -5, 8, 12, -9, 1, -6, 7, -13
Var
    Arch: text;
    A, Sum, Cont : integer;
Begin
    Assign( Arch, 'datos.TXT' );
    Reset ( Arch );
    Sum:=0;
    Cont := 0;
    While NOT eof ( Arch ) do
        Begin
            Readln( Arch, A );
            If A > 0 then
                begin
                    Sum := Sum + A ; Cont := Cont + 1;
                End;
            End;
        Close ( Arch );
        If Cont <> 0 then
            Writeln( Sum / Cont )
        Else
            Writeln ( 'no hay numeros positivos' );
        End.

```

Ej 9) Dado un conjunto de números leídos desde un archivo, informar si están ordenados en forma ascendente.

Ej 10) Leer desde un archivo números enteros ordenados de menor a mayor, puede haber elementos repetidos.

Calcular y mostrar cada número con su respectiva frecuencia.

Ejemplo: 2, 2, 3, 3, 3, 3, 7, 9, 9, 9

Numero	Frecuencia
2	2
3	4
7	1
9	3

Ej 11) Reescriba utilizando archivo de texto para ingreso de datos los siguientes ejercicios:

a) Ej 6, considerar en la primera línea del archivo el saldo inicial y en cada una de las líneas siguientes un movimiento: Tipo, Monto (notar que no es necesario el tipo F).

b) Ej 7, en cada línea del archivo: Nombre (15 caracteres) y las tres notas reales

Ej 12) Considere el siguiente archivo de texto, donde **b** indica el caracter blanco:

```

abcd45.9111<eoln>
80xyz<eoln>
111<eoln>< eof >

```

y se ha declarado:

```
T:text;  
Cad1: string[8];  
Cad2, Cad3 : string[3];  
a,b : word;  
x: real;
```

c: char;

Indique para cada caso la/s sentencia/s de lectura que provocaría/n los siguientes resultados:

Sentencia/s de lectura	Cad1	Cad2	Cad3	a	b	x	c
	'b b b b b 111'	'abc.'	-	80	-	45.9	'd'
	-	'd b b'	'abc'	111	-	80.0	-
	'a b c d b b 45'	'b 80'	'9 b'	-	111	-	'b'
	'b b 80 b x z y'	-	-	-	-	111.0	-

Nota: el '-' indica que no se ha leído sobre dicha variable.

Ej 13) Dado el siguiente fragmento de código, analice si se puede reescribir usando la estructura **Repeat-until**, comparar ambos códigos. ¿A qué conclusión llega?

```
. . .
Readln( Tipo, Precio);  Tipo := Uppcase(Tipo);
{P=panadería, F=fiambrería, A=almacén}
While (Tipo<>'P') and (Tipo<>'F') and (Tipo<>'A') do
Begin
  Readln (Tipo);  Tipo := Uppcase(Tipo);
End;
. . .
```

Ej 14) a) Analice y explique brevemente qué realiza el siguiente fragmento de código. Indicar el resultado en cada uno de los tres lotes de prueba.

```
. . . . .
Bool:=false;
Sum:=0;
Readln( x );
If x <>0 then
  Repeat
    Sum := Sum + x;
    If Sum > 100 then
      Bool :=true;
    Readln( x );
  Until ( x=0) or Bool;
Writeln('Resultado', Sum );
. . . . .
```

Lote1 : 0	Resultado
Lote2 : 70, 40, 20	Resultado
Lote3 : 70 , 0	Resultado

b) Reescríbalo reemplazando la estructura **Repeat**, los dos **If** y la variable booleana **Bool** por una estructura **While**.

Ej 15) A partir de un archivo de texto que contiene una secuencia de caracteres terminada en punto, desarrollar un programa para cada ítem:

- Cuántas palabras comienzan con una letra que ingresa por teclado.
- Cuántas palabras contienen una letra que ingresa por teclado.

Ej 16) Una empresa de autos de alquiler posee N vehículos y ha registrado la siguiente información de los viajes realizados, por cada uno de ellos:

- ✓ Patente
- ✓ Total de litros de combustible consumidos

Y por cada viaje realizado

- Kilómetros recorridos. (0 indica fin de datos).

Se pide ingresar los datos e informar:

- Por cada vehículo el consumo que tuvo (cantidad de litros por km. recorrido)
- Patente del vehículo que más viajes hizo.
- Total de litros consumidos por el auto que hizo la menor cantidad de kms.

Ej 17) La cajera de un supermercado ha registrado varias ventas, de cada una sabe:

- ✓ Nombre del cliente (“ ***** “ indica fin de datos)
- ✓ Cantidad de artículos que compró el cliente y por cada uno se conoce:
 - Precio base del artículo
 - Si está o no en promoción (‘S’ ó ‘N’)

En el caso de estar en promoción, se realiza un descuento del 10% sobre el precio base.

Se pide ingresar los datos e informar:

- a) Monto total de cada venta, el total del descuento y el precio final a pagar.
- b) Nombre del cliente que compró la mayor cantidad de artículos en oferta.
- c) Cuántos clientes compraron solo artículos en oferta.

Ej 18) Reescribir el ejercicio 17, considerando que los datos están grabados en un archivo (con el formato que considere más adecuado).

Ej 19) Leer desde archivo un conjunto de números, con elementos negativos no consecutivos que delimitan subconjuntos. Obtener e informar el valor máximo de cada subconjunto (el máximo entre dos negativos).

Ejemplo:

2 3 4 -7 4 5 -5 7 5 3 9 8 7 -1 3 1 -2 → 5 9 3

2 3 4 -7 4 5 -5 7 5 3 9 8 7 -1 3 1 → 5 9

Ej 20) Leer desde un archivo números enteros, calcular y escribir el promedio de los elementos positivos y distintos de cero que se encuentran entre dos negativos no consecutivos.

Ejemplos:

-7 6 -1 0 2 4 -8 -7 3 1 -24 6 7 0 9 → debe escribir : 6 3 2

7 6 -1 0 2 4 -8 -7 3 1 -24 6 7 0 9 -8 → debe escribir : 3 2 7.33

Ej 21) Leer un conjunto de números enteros, la presencia de ceros intermedios indica fin de un subconjunto y comienzo del siguiente, dos ceros consecutivos es fin de datos. Se pide calcular e informar el máximo en cada subconjunto y el orden del subconjunto más numeroso.

Ejemplo: 3,5,2,0,6,7,4,7,0,2,1,0,0

Respuesta: subconjunto Máximo

1	5
2	7
3	2

El subconjunto con más elementos es: 2

Ej 22) A partir de un archivo de texto que contiene palabras separadas entre sí por uno o más blancos desarrollar un programa para:

- a) Mostrar la palabra con más cantidad de vocales del texto.
- b) Grabar en un texto de salida las palabras con más de cuatro letras pero escritas al revés.
- c) Contar cuántas son palíndromas (capicúas).

Ej 23) Un archivo contiene palabras (letras mayúsculas, minúsculas, dígitos, caracteres especiales) separadas entre sí por uno o más blancos. Se debe verificar para cada una si se trata de una contraseña válida, y en ese caso grabarla en un archivo de salida. Indicar al final del proceso el porcentaje de palabras que no son contraseñas válidas y mostrar la contraseña inválida más larga (puede no existir).

Una contraseña válida debe:

- ✓ estar conformada como mínimo por 8 caracteres,
- ✓ incluir obligatoriamente, *al menos una* letra mayúscula y *al menos una* minúscula
- ✓ incluir *exactamente* cuatro dígitos.
- ✓ no contener caracteres diferentes de letras y dígitos.

Ejemplo:

eR68G12a 91jY643ebjp eRty74kLh 24fG92 aj85gT32 eL8j\$8215 dGb9357jKoup

La salida del programa sería:

El % de contraseñas inválidas es 57.14

La longitud de la contraseña inválida más larga es de 11 caracteres

Ej 24) En un archivo se han grabado palabras, separadas por uno o más blancos en una misma línea, finalizando con un punto. Cada palabra está conformada por letras y dígitos (3ab4c3Hd). Se pide, a partir del archivo descripto generar otro cambiando las parejas “LetraminusculaDigito” por las repeticiones de la Letra según indique el dígito.

Ejemplos: ab4C3hd 1Ab2c3hD2. → abbbbC3hd 1AbbccchD2.
Ab0C3hd 1Ab2c0hD2. → AC3hd 1AbbhD2.