

ALGORITMOS Y ESTRUCTURAS
DE DATOS

Ejercicios Resueltos:
CUADERNILLO DE PRACTICA
PRIMERA PARTE

I - Entrada, salida, asignación de datos

Ejercicio Resuelto: Dados dos números enteros obtener su suma, resta, multiplicación y división.

Chapín.

Ejer2
Leer (num1, num2)
$\text{Sum} \leftarrow (\text{num1} + \text{num2})$
$\text{Res} \leftarrow (\text{num1} - \text{num2})$
$\text{Mul} \leftarrow (\text{num1} * \text{num2})$
$\text{Divi} \leftarrow (\text{num1} / \text{num2})$
Mostrar ('el resultado de la suma es ', sum)
Mostrar ('el resultado de la resta es ', res)
Mostrar ('el resultado de la multiplicación es ', mul)
Mostrar ('el resultado de la división es ', divi)

Programa Pascal.

```
PROGRAM EJER2(input, output);
{suma, resta, multiplica y divide 2 numeros enteros}
VAR num1,num2:INTEGER;
VAR sum,res,mul:INTEGER;
VAR divi:REAL;

BEGIN
WRITELN ('Escriba dos numeros enteros');
READLN(num1);
READLN(num2);
sum:=num1 + num2;
res:=num1 - num2;
mul:=num1 * num2;
divi:=num1 / num2;
WRITE ('EL RESULTADO DE LA SUMA: '); WRITELN (sum);
WRITE ('EL RESULTADO DE LA RESTA: '); WRITELN (res);
WRITE ('EL RESULTADO DE LA MULTIPLICACION: '); WRITELN (mul);
WRITE ('EL RESULTADO DE LA DIVISION: '); WRITE (divi);
END.
```

Ejercicio Resuelto: Determinar el número de horas, minutos y segundos que hay en 6250 segundos.

Chapín.

Ejer3

Seg \leftarrow 6250
Hor \leftarrow seg div 3600
min \leftarrow (seg mod 3600) div 60
seg_res \leftarrow (seg mod 3600) - (min * 60)
Mostrar ('En', seg, 'segundos hay: ')
Mostrar ('Horas: ' hor, ' Minutos: ', min, 'y segundos: ', seg_res)

Programa Pascal.

```

PROGRAM EJER3(input, output);
VAR seg,hor, min, seg_res:INTEGER;

BEGIN
seg= 6250;
hor:= seg div 3600;
min:= (seg mod 3600) div 60;
seg_res:= (seg mod 3600) - (min * 60);
WRITELN ('EN ',seg,' SEGUNDOS HAY: ');
WRITE (hor,' hora',min,' minutos ', seg_res,' segundos');
END.

```

II - Estructuras de decisión simple

Ejercicio Resuelto: Dado un número determinar si es positivo o negativo.

Chapín.

Ejer4

Leer (num1)		
<div style="display: flex; justify-content: space-between;"> V Num1 > 0 F </div>		
Mostrar ('el número es positivo')	Num1 < 0	
	V	F
F	Mostrar ('El Número es Negativo')	

Programa Pascal.

```

PROGRAM EJER4 (input, output);
VAR num1:INTEGER;

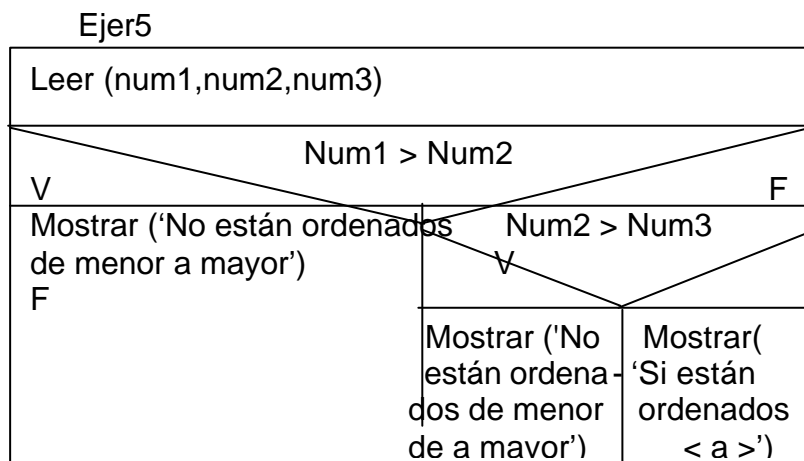
BEGIN
WRITE ('Introduzca un numero entero: ');
READLN (num1);

IF num1 > 0 THEN
    WRITE ('El numero es positivo')
ELSE
    IF num < 0 THEN
        WRITE ('El numero es negativo')
    ELSE
        WRITE ('El numero no es positivo ni
negativo, es 0');
END.

```

Ejercicio Resuelto: Ingresar tres nros. Enteros, distintos. Determinar y mostrar si ingresaron en orden creciente.

Chapín.



Programa Pascal.

```

PROGRAM EJER5(input, output)
VAR num1,num2,num3:INTEGER;

BEGIN
WRITE ('Introduzca el primer numero : ');
READLN (num1);
WRITE ('Introduzca el segundo numero (2) : ');
READLN (num2);
WRITE ('Introduzca el tercer numero (3) : ');
READLN (num3);

```

Algoritmos y Estructuras de Datos
Práctica 1

```

IF num1 > num2 THEN
    WRITE ('Los numeros no están ordenados de menor a mayor')
ELSE IF num2 > num3 THEN
    WRITE ('Los numeros no están ordenados de
menor a mayor')
ELSE
    WRITE ('Los numeros están ordeados de menor a
mayor');
END.

```

Ejercicio Resuelto: Dado un número del 1 a 7 determinar el nombre del día de la semana que corresponde.

Chapín utilizando Sentencia condicional simple.

Ejer6

Leer (num)	
Num = 1	
V	F
Mostrar ('Lunes')	
Num = 2	
V	F
Mostrar ('Martes')	
Num = 3	
V	F
Mostrar ('Miércoles')	
Num = 4	
V	F
Mostrar ('Jueves')	
Num = 5	
V	F
Mostrar ('Viernes')	
Num = 6	
V	F
Mostrar ('Sábado')	
Num = 7	
V	F
Mostrar ('Domingo')	

Programa Pascal.

```
PROGRAM EJER6(input, output);
VAR num:INTEGER;

BEGIN
WRITE ('Escriba un numero para ver que día corresponde: ');
READLN (num);
IF num=1 THEN
    WRITE ('Lunes');
IF num=2 THEN
    WRITE ('Martes');
IF num=3 THEN
    WRITE ('Miercoles');
IF num=4 THEN
    WRITE ('Jueves');
IF num=5 THEN
    WRITE ('Viernes');
IF num=6 THEN
    WRITE ('Sabado');
IF num=7 THEN
    WRITE ('Domingo');
END.
```

Chapín utilizando Sentencia condicional múltiple.

Ejer6

Leer (num)						
						num
1	2	3	4	5	6	7
Mostrar('Lunes')	Mostrar('Martes')	Mostrar('Miercoles')	Mostrar('Jueves')	Mostrar('Viernes')	Mostrar('Sabado')	Mostrar('Domingo')

Programa Pascal.

```
PROGRAM EJER6(input,output);
VAR num:INTEGER;

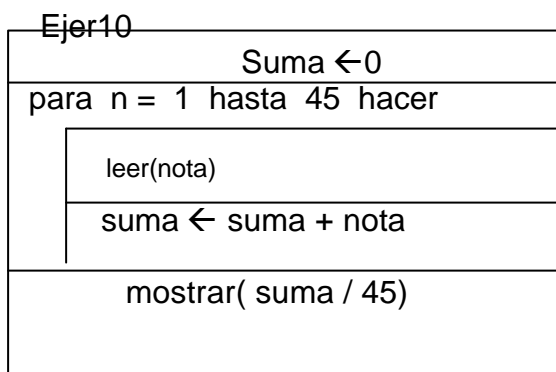
BEGIN
WRITE('Ingrese un número entre 1 y 7 ');
READLN(num);
CASE num OF
1: Writeln('Lunes');
2: Writeln('Martes');
3: Writeln('Miercoles');
4: Writeln('Jueves');
```

```
5: WRITELN('Viernes');  
6: WRITELN('Sabado');  
7: WRITELN('Domingo');  
END;  
END.
```

III - Estructuras de Iteración

Ejercicio Resuelto: Dadas las notas de un parcial de los 45 alumnos de un curso, se desea obtener la nota promedio del curso.-

Chapín



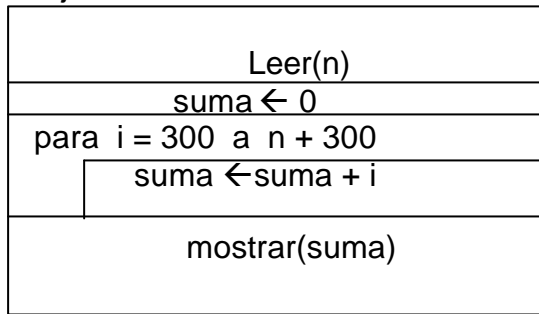
Programa Pascal.

```
PROGRAM EJER10 (INPUT, OUTPUT);  
VAR  
    N : INTEGER;  
    SUMA , NOTA : REAL;  
BEGIN  
    SUMA := 0;  
    FOR N := 1 TO 45 DO  
        BEGIN  
            WRITE ( ' INGRESE NOTA ');  
            READLN ( NOTA);  
            SUMA := SUMA + NOTA  
        END;  
    WRITE ( ' EL PROMEDIO DEL CURSO ES = ', SUMA/45 :4:2)  
END.
```

Ejercicio Resuelto: Se desea obtener la suma de los N números naturales posteriores al número 300 inclusive.-

Chapín

Ejer11



Programa Pascal.

```
PROGRAM EJER11 ( INPUT, OUTPUT);
VAR
    N , I : INTEGER;
    SUMA : REAL;
BEGIN
    WRITE ('INGRESE CANTIDAD DE NROS. ');
    READLN (N);
    SUMA := 0;
    FOR I := 300 TO N + 300 DO
        SUMA := SUMA + I;
    WRITE ('LA SUMA DE LOS',N:4,'NROS NATURALES >= 300 ES',
        SUMA:10:0)
END.
```

III.2) Con cantidad desconocida de veces

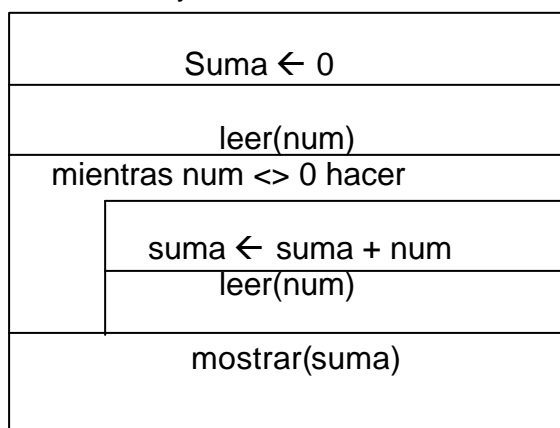
FIN DE DATOS: es una marca o señal que se agrega al final del conjunto de datos a procesar para indicar el fin de dicho conjunto. Debido a que se adiciona a los datos, su tipo debe ser compatible con aquellos. Se lo denomina también "dato centinela".

III.2.1) Estructura Mientras

Ejercicio Resuelto: Se van ingresando números distintos de cero, salvo el último valor. Determinar su suma.-

Chapín

Ejer12



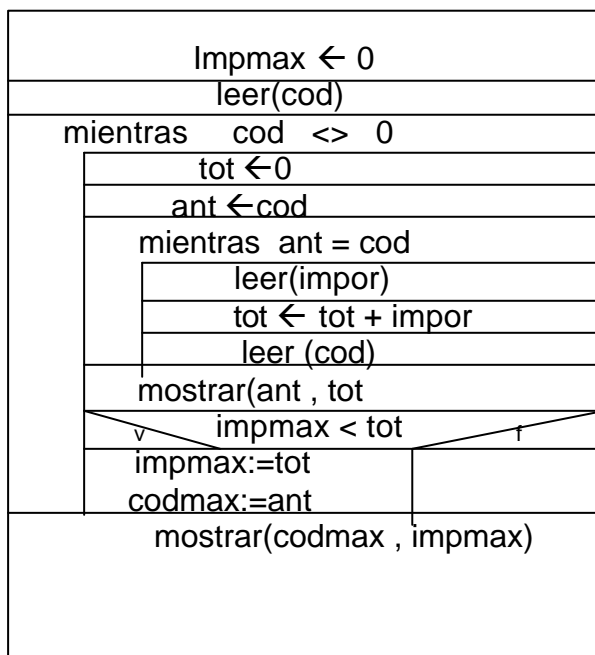
Programa Pascal.

```
PROGRAM EJER12(INPUT, OUTPUT);
VAR
    NUM , SUMA : REAL;
BEGIN
    SUMA := 0;
    WRITE ('INGRESE NUMERO');
    READLN (NUM);
    WHILE NUM <> 0 DO
        BEGIN
            SUMA := SUMA + NUM;
            WRITE ('INGRESE NUMERO');
            READLN (NUM)
        END;
    WRITE ('LA SUMA ES = ', SUMA:9:2)
END.
```

Ejercicio Resuelto: Se desea saber el total de ventas de cada uno de los vendedores de una empresa. A tal fin se tienen como datos: el código de vendedor y el importe de cada una de las ventas; un vendedor puede haber realizado más de una venta. No se sabe la cantidad de vendedores que tiene la empresa ni la cantidad de ventas hechas por cada vendedor (un código de vendedor igual a cero es fin de datos).-ESTOS DATOS ESTAN ORDENADOS POR CODIGO DE VENDEDOR. Exhibir cada código de vendedor y su total correspondiente y al final, el código de vendedor con mayor importe vendido y dicho importe.-

Chapín

Ejer13



Programa Pascal.

```
PROGRAM EJER13(INPUT, OUTPUT);
{EJEMPLO DE CORTE DE CONTROL}
```

```

VAR
    COD , ANT , CODMAX : INTEGER;
    IMPOR , TOT , IMPMAX : REAL;
BEGIN
    IMPMAX := 0;
    WRITE ('INGRESE CODIGO');
    READLN (COD);
    WHILE COD <> 0 DO
    BEGIN
        TOT := 0;
        ANT := COD;
        WHILE ANT = COD DO
        BEGIN
            WRITE ('INGRESE IMPORTE');
            READLN (IMPOR);
            TOT := TOT + IMPOR;
            WRITE ('INGRESE CODIGO');
            READLN (COD)
        END;
        WRITE ('EL VENDEDOR',ANT:4,' VENDIO $ ',TOT:15:2);
        IF TOT > IMPMAX
        THEN
            BEGIN
                IMPMAX := TOT;
                CODMAX := ANT
            END;
        END;
        WRITE ('EL VENDEDOR : ',CODMAX:4,' TUVO MAYOR IMPORTE: $',
            IMPMAX:15:2)
    END.

```

III.2.2) Estructura Repetir – Hasta

Ejercicio Resuelto: Evaluar y tabular la función $f(X) = 3X + 2$ para diferentes valores de X .-

Chapín

Ejer14

	leer(x)
	$fx \leftarrow 3 * x + 2$
	mostrar(x , fx)
	mostrar('continúa o finaliza?: c/f')
	leer(rta)
	hasta rta = 'f'

Programa Pascal.

```

PROGRAM EJER14(INPUT, OUTPUT);

VAR
    X , FX : INTEGER;
    RTA : CHAR;
BEGIN
    REPEAT
        WRITE ('INGRESE VALOR ');
        READLN (X);
        FX := 3 * X + 2;
        WRITE ('CONTINUA O FINALIZA INGRESANDO? C/F ');
        READLN (RTA)
    UNTIL RTA = 'F'
END.

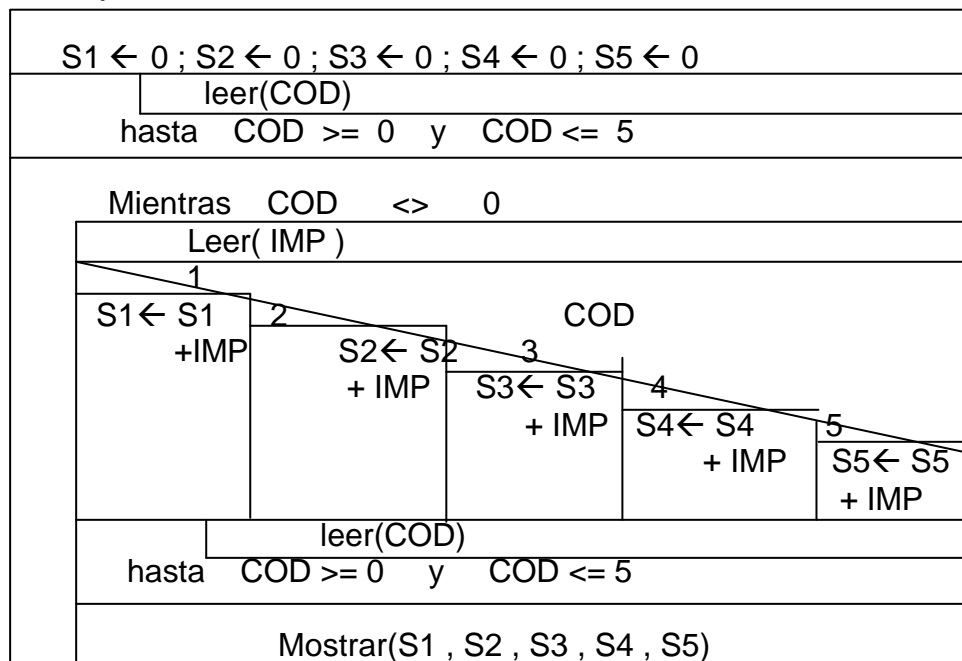
```

IV - Estructuras de Selección Múltiple

Ejercicio Resuelto: Se tienen como datos los importes de las ventas de cada una de las sucursales de una empresa, junto con el código de sucursal (1, 2, 3, 4 ó 5).- Cada sucursal puede tener varias ventas. Los datos no están ordenados por código de sucursal. Un código igual a cero indica fin de datos.- Obtener el total de ventas para cada sucursal.-

Chapín

Ejer15



Programa Pascal.

```

PROGRAM EJER15 (INPUT, OUTPUT);
VAR
    COD : INTEGER;
    S1 , S2 , S3 , S4 , S5 , IMP : REAL;

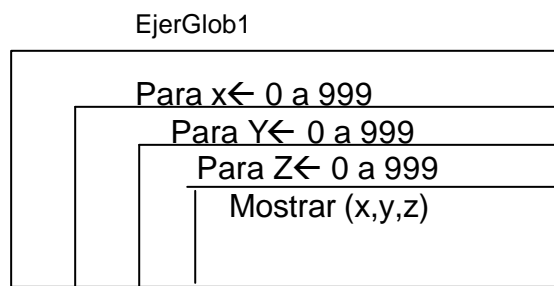
```

```
BEGIN
  S1:= 0; S2:= 0; S3:= 0; S4:= 0; S5:= 0;
  REPEAT
    WRITE ('INGRESE CODIGO');
    READLN (COD)
  UNTIL ( COD >= 0 ) AND ( COD <= 5 );
  WHILE COD <> 0 DO
    BEGIN
      WRITE ('INGRESE IMPORTE');
      READLN (IMP);
      CASE COD OF
        1 : S1 := S1 + IMP;
        2 : S2 := S2 + IMP;
        3 : S3 := S3 + IMP;
        4 : S4 := S4 + IMP;
        5 : S5 := S5 + IMP
      END;
      REPEAT
        WRITE ('INGRESE CODIGO');
        READLN (COD)
      UNTIL ( COD >= 0 ) AND ( COD <= 5 )
      END;
      WRITELN ('TOTAL SUCURSAL 1 ::30, S1:12:2);
      WRITELN ('TOTAL SUCURSAL 2 ::30, S2:12:2);
      WRITELN ('TOTAL SUCURSAL 3 ::30, S3:12:2);
      WRITELN ('TOTAL SUCURSAL 4 ::30, S4:12:2);
      WRITE ('TOTAL SUCURSAL 5 ::30, S5:12:2)
    END.
  END.
```

VI - Ejercicios Globalizadores

Ejercicio Resuelto: Simular un contador de vueltas de tres dígitos de un grabador. Debe comenzar con tres dígitos iguales a 0 0 0, variar de a un dígito por vez, cuando cada uno llega a 9 se debe poner en 0, y reiniciar la operación hasta que los 3 dígitos lleguen a 9 (9 9 9).

Chapín



Programa Pascal.

```
PROGRAM EjerGlob1 (INPUT, OUTPUT);
VAR
  X,y,z : INTEGER;
```

```

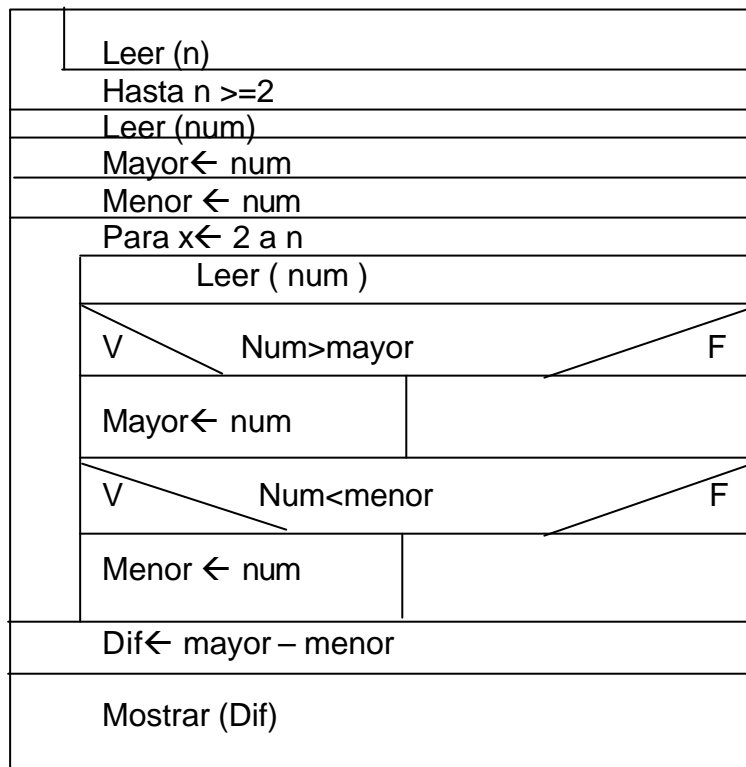
BEGIN
  FOR X:= 0 TO 999 DO
    FOR Y:=0 TO 999 DO
      FOR Z:=0 TO 999 DO
        WRITE (X,Y,Z)
      END.
    END.
  END.

```

Ejercicio Resuelto: Ingresar un conjunto de N (validar que N sea mayor ó igual que 2) valores numéricos desordenado. Hallar y exhibir su rango, esto es: la diferencia entre su valor máximo y su valor mínimo.

Chapín

EjerGlob2



Programa Pascal.

```

PROGRAM EjerGlob2 (INPUT, OUTPUT);
VAR
  N, num, mayor, menor : INTEGER;
BEGIN
  Repeat
    read (n);
  until n >=2;
  read (num);
  Mayor := num;
  Menor := num;
  For x:= 2 to n do
    Begin
      read ( num );
      IF num > mayor then mayor:= num;
      IF num < menor then menor := num;
    End
  End

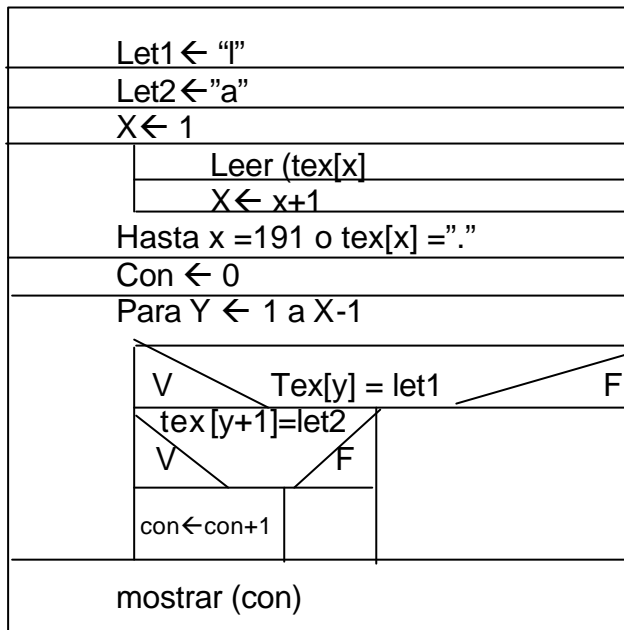
```

End;
Dif:= mayor – menor;
Write ('la diferencia entre el mayor y el menor es: 'dif)
END.

Ejercicio Resuelto: Se cuenta con un texto de hasta 190 caracteres. Determinar cuantas veces aparece la sílaba "la". Se sabe que el texto no cuenta con caracteres de puntuación.

Chapín

EjerGlob3



Programa Pascal.

```

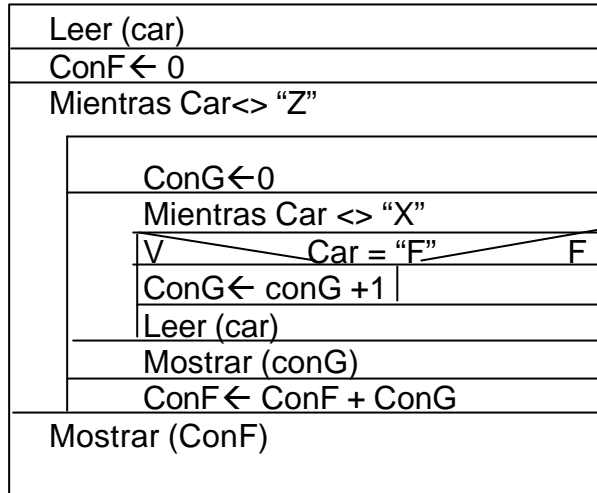
PROGRAM EjerGlob3 (INPUT, OUTPUT);
VAR
  x, y : INTEGER;
  let1, let2: char;
  tex: array [1..190] of char;

BEGIN
  Let1:= "l";
  Let2:= "a";
  X:=1;
  Repeat
    read (tex[x];
    X:=x+1;
  Until x =191 or tex[x] = ".";
  Con :=0;
  FOR Y := 1 to X-1 DO
    If Tex[y] = let1 then IF tex[y+1]=let2 then con←con+1;
  Write ( 'la cantidad de veces que aparece la silaba LA es: ', con)
END.
```

Ejercicio Resuelto: Se tiene una sucesión de grupos que contienen caracteres (los caracteres vienen sin ningún orden específico). Luego de cada grupo viene una 'X' y al final de todos los grupos una 'Z'. Se desea determinar la frecuencia relativa y absoluta de la letra 'F'.

Chapín

EjerGlob4



Programa Pascal.

```

PROGRAM EjerGlob4 (INPUT, OUTPUT);
VAR
    cong, conf : INTEGER;
    car:char;
BEGIN
    Read(car);
    Conf:= 0;
    While car <> "Z" do
    BEGIN
        Cong:=0;
        While car <> "X" do
            BEGIN
                If car = "F" then Cong:=cong+1;
                Read(Car)
            END;
        Write( "Frecuencia relativa: ",cong);
        Conf:=conf+cong
    END;
    Write ("Frecuencia Absoluta: ", conf)
END.

```