

República Bolivariana de Venezuela
Ministerio de Educación Superior
Instituto Universitario de Tecnología Valencia
Departamento de Informática



GUIA TEORICA N° 1
LENGUAJE DE PROGRAMACION I
APUNTES: INTRODUCCIÓN A PASCAL.

Profesor : DELY M. GIL A.

VALENCIA, 2004

**PASCAL ES UN LENGUAJE DE PROGRAMACIÓN
DE ALTO NIVEL**

- Lenguaje excelente para el aprendizaje de Programación
- Lenguaje estructurado
- Lenguaje fuertemente tipificado
- Lenguaje recursivo
- Facilidad para realizar programación modular

Ing Del y Gil

ESTRUCTURA GENERAL DE UN PROGRAMA

Program Identificador; {Cabecera Opcional}
Uses
Unit_1, Unit_2,..., Unit_n;
Label
lista_de_etiquetas; {Sección de etiquetas}
Const
Definición de constantes;
Type
Declaración de tipos de datos definidos por el Usuario
Var
Declaración de Variables Globales
Definición de Procedimientos y/o Funciones
Begin { Cuerpo principal del Programa }
Sentencias;
End.

Ing Del y Gil

USES

lista de identificadores;

Lista de identificadores: unidades estándar de Turbo Pascal y unidades definidas por el Usuario

Unidades estándar de Turbo Pascal:

System	Dos	Crt
Printer	Turbo3	Graph
Graph3.		

Ejemplo:

```
Uses Dos, Crt, Demo;  
{Dos,Crt: unidades estándar; Demo:  
unidad creada por el Usuario
```

Ing Del y Gil

SYSTEM

Contiene todos los procedimientos y funciones de Turbo Pascal relativas a E/S, cálculos de coma flotante, gestión de memoria, etc. Se enlaza automáticamente en la compilación de cada programa y no es necesaria referenciarla, está explícita.

Procedimientos y Funciones:

Abs	Assign	Delete	Exit	Fillchar
Halt				
Inc	IOResult	Random	Read	Seek
Sizeof				
Sqr	Sqrt	Uppcase	Write	

Ing Del y Gil

CRT

Contiene los procedimientos y funciones específicos de control de la pantalla de un PC.

Procedimientos y Funciones:

ClrEol	InsLine	Sound	ClrScr
Delay(Ms)	DelLine	Gotoxy(x,y)	
NoSound	ReadKey	Keypressed	
TextColor(color)		TextBackground(color)	
Window	WhereX	WhereY	

Ing Del y Gil

DOS

Esta unidad define numerosos procedimientos y funciones en Pascal que son equivalentes a los que comúnmente son usados en el sistema DOS.

Procedimientos y Funciones:

DiskFree	DiskSize	Exec	GetDate
GetTime	SetDate	SetTime	

```
Uses DOS;  
begin  
  write(DiskFree(0) div 1024);  
  write(' K bytes disponibles en la unidad por  
defecto');  
end.
```

Ing Del y Gil

PRINTER

Facilita el manejo de la impresora usando en las instrucciones de salida WRITE y WRITELN, el comando LST.

Ejemplo:

```
Program PruebaImpresora;  
Uses Printer;  
begin  
  writeln(Lst, 'Prueba Impresora':40);  
end.
```

Ing Del y Gil

STRINGS

Contiene funciones para el manejo de cadenas terminadas en nulo

Funciones:

```
StrCat      StrComp  StrCopy   StrLen  
StrLower   StrUpper  StrPos
```

Ejemplo:

```
ReadLn(Cad);  
WriteLn(StrUpper(Cad));  
WriteLn(StrLower(Cad));
```

Ing Del y Gil

TURBO3 y GRAPH 3

Manejan compatibilidad con Turbo Pascal 3.0

GRAPH

Esta unidad permite utilizar las capacidades gráficas del PC, conteniendo una serie de constantes, tipos, variables, procedimientos y funciones relacionadas con gráficos.

En el siguiente ejemplo haremos un programa que permite realizar una serie de círculos, unos dentro de los otros.....

Ing Del y Gil

```
Program Graficos;  
Uses  
  Graph; {Se carga la unidad estándar}  
Var  
  gd,gm:Integer;  radio:Byte;  
Begin  
  gd:= DETECT;  
  INITGRAPHIC(GD,GM,"");  
  CASE GRAPHResult <> GROCK OF  
    TRUE: HALT(1);  
  End;  
  For radio:= 1 to 23 do  
    CIRCLE(300,240,radio*10);ReadLn;  
End.
```

Ing Del y Gil

LABEL

Es una constante entera que identifica a una instrucción dentro del programa. Se usa con la sentencia GOTO.

```

Program Ejm_Label;
Var
a,b,c :integer;
Begin
  C:= 10;
  10: Read(a);
  write(a);
End.
    
```

```

b:= a*100;
writeln(b);
c:= c+1; {Inc(i)}
If c<17 then
  goto 10
    
```

Ing Del y Gi I

CONST

Como su nombre lo indica, son objetos que mantienen su valor durante toda la ejecución del programa. Pueden ser de diferentes tipos.

TYPE

Son identificadores de datos tipos estructurados, como registros, archivos...

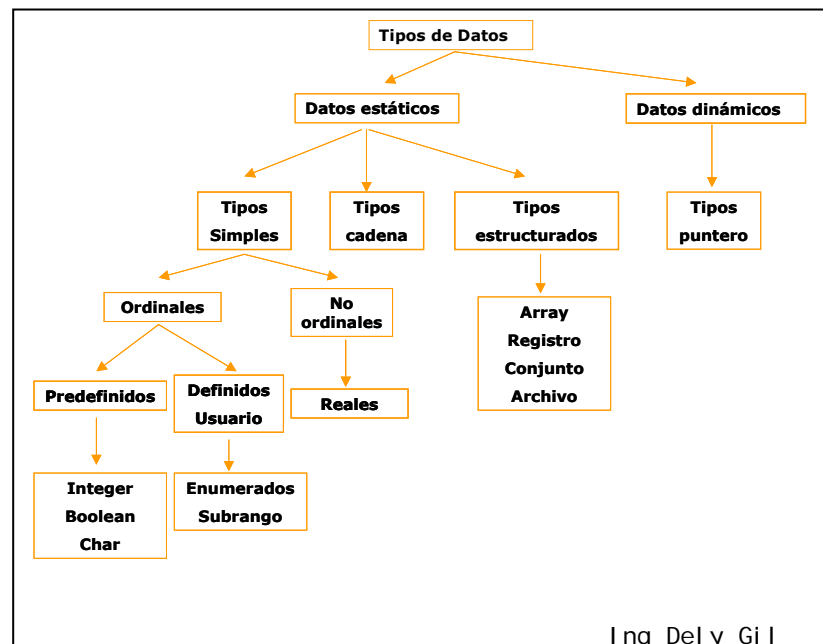
VAR

Declaración de las variables y su tipo, a usar en el programa principal.

```

Var
  Var1, Var2,...: tipo1; {tipo de dato válido}
    
```

Ing Del y Gi I



Ing Del y Gi I

TIPOS ENTEROS

Pascal dispone de cinco (5) tipos predefinidos que permiten representar los valores enteros

TIPO	RANGO
Byte	0..255
Integer	-32768..32767
Longint	-217483648..2147483647
Shortint	-128..127
word	0..65535

Ing Del y Gi I

TIPOS REALES

Están comprendidos de una parte entera y una parte decimal.

TIPO	RANGO
Real	21.9 10E-39 ... 1.7E-38
Single	1.6 E-45 ... 3.4E38
Double	5.0E-324 ... 1.7E308
Extended	3.4E-4932 ... 1.1E4932
Comp	-2E63 +1 ... 2E63 -1

Ing. Dely Gil

BOOLEAN (Lógicos)

Pueden tomar sólo dos valores:
True y False.

CHAR (Carácter)

Es un tipo de dato que puede tener sólo un carácter ASCII (American Estándar Code for Information Interchange) (Código Estándar Americano para Intercambio de Información)

Ejemplo:

```
Chr(65)→ A      #65      #13?  
Chr(66)→ B      #66      #27?  
                ^G?  
                ^M^J?
```

Ing. Dely Gil

STRING (Cadena de Caracteres)

Es una secuencia de caracteres de cero o más caracteres correspondientes al código ASCII.

Type

Identificador = String[N];

Var

Identificador : String[N];

N → Especifica el número de caracteres máximos a almacenar (1..255)

Ejm:

```
Type      Cad30 = string[30];  
Var       Nombre : string[30];  
          {Nombre : cad30;}
```

Ing. Dely Gil

TIPOS DEFINIDOS POR EL USUARIO

A partir de los datos simples se crean las estructuras de datos, estos aumentan la legibilidad de los programas y simplifican su mantenimiento.

Se clasifican en:

- Escalares (Enumerados y Subrangos)
- Registros (Record)
- Arreglos (Arrays)
- Conjuntos (Set)
- Archivos (File)
- Punteros (Pointer)
- Objetos (Object)

Ing. Dely Gil

SUBRANGOS

Sintaxis: Se define en la Sección TYPE.

Type

Nombre = Límiter inferior..límite superior;

Ejemplos:

IntervalosEnteros = -100..100;

LetrasMin = 'a'.. 'z';

Digitos = '0'.. '9';

Ing Del y Gil

ENUMERADOS

Sintaxis: Se define en la Sección TYPE.

Type

Nombre = (constante1,constante2,..,constanten)

Ejemplos:

Type

Vehiculos = (moto,automovil,autobus,tranvia,tren);

Dias = (lunes,martes,miercoles,jueves,viernes,
sabado, domingo);

Arcoiris = (verde,amarillo,rojo,azul);

Ing Del y Gil

ARREGLOS

Type

Nombre = array[1..Max] of tipo;

Letras = array[1..5] of char;

REGISTROS

reg_emp = record

Cedula : longint;

Nombre : string[20];

end;

Var

registro : reg_emp;

archivo : file of reg_emp;

Ing Del y Gil

CONJUNTOS

Type

Automoviles = (Seat, Ford, BMW,Peugeot,Opel,
Mercedes);

Marca = Set of Automoviles;

Var

Avis, Hertz, rentCar : Marca;

Type

Letras = ['A', 'B', 'C', 'D', 'E'];

NoLetras = [];

Var

Nombre : Set of Letras;

Ing Del y Gil

PUNTEROS

Son tipos especiales de variable que se utiliza para almacenar la dirección de memoria de otra variable, o lo que es igual, su valor es una dirección de una posición de memoria donde está almacenada otra variable.

```
Type
  nombre_tipo = identificador_tipo;
  PunteroReal = real;
Var
  nombre_var_ptr : nombre_tipo;
  A,B: PunteroReal;
```

Ing Del y Gil

OBJETOS

Los objetos son tipos definidos por el Usuario. Estos se llaman *Clases*, sus campos se llaman variables de *instancias* y sus procedimientos o funciones se llaman *métodos*.

```
Type
  estudiante = Object
                nombre : string[20];
                nota1,nota2 : integer;
                procedure Iniciar(num:string;
                integer);
                function media: real;
                procedure informar;
end;
```

Ing Del y Gil

IDENTIFICADORES

Son los nombres que se le dan a las constantes, variables, funciones o procedimientos.

Reglas:

- √ Debe resultar significativo, sugiriendo lo que representa
- √ No debe coincidir con palabras reservadas de Pascal
- √ Máximo 63 caracteres.
- √ Comenzar por un carácter alfabético o el subrayado y los siguientes pueden ser letras, dígitos o el símbolo de subrayado. (ASCII < 127)
- √ Usar mayúscula o minúscula.

Ing Del y Gil

VARIABLES

Las variables son objetos de un programa cuyo valor puede cambiar durante la ejecución del programa. El cambio se produce mediante sentencias ejecutables.

Declaración:

```
Var
  Lista de variables : tipo;
                Lista de variables separadas ( ,)
Ejm:
  Cont_Fem, Cont_Masc : byte;
  Salario_Base, Salario : real;
```

Ing Del y Gil

SENTENCIAS

Las sentencias describen las acciones algorítmicas que pueden ser ejecutadas.

Sentencia de Asignación:

Variable := expresión;

Ejemplo

```
A           := 16;  
Interruptor := true;  
Car         := #70  
           { ASCII 70 "F"}
```

Ing Del y Gil

CONTADORES

Es una variable cuyo valor se incrementa o decrementa en una cantidad constante cada vez que se produce un determinado suceso o acción. (Estructuras repetitivas)

✓ INICIALIZACIÓN (antes del bucle)

Cont_Fem := 0;

✓ INCREMENTO O DECREMENTO (Interior del bucle)

Cont_Fem := Cont_Fem + 1 ;

EL TIPO DE DATO DE LOS CONTADORES DEBE SER POSITIVO (BYTE, WORD)

Ing Del y Gil

ACUMULADORES

Son variables cuyo valor se incrementa o decrementa en una cantidad variables.

✓ INICIALIZACIÓN (antes del bucle)

Suma := 0; Factorial := 1;

✓ ACUMULACIÓN (Interior del bucle)

```
Suma      := Suma + nota;  
Factorial := Factorial*valor
```

Ing Del y Gil

INTERRUPTORES

Un interruptor o bandera (*switch*) puede tomar los valores true o false a lo largo de la ejecución del programa. Se utilizan para controlar los bucles.

✓ INICIALIZACIÓN (antes del bucle)

Encontro := false;

✓ ASIGNACIÓN (Interior del bucle)

```
Si la condición ocurre entonces  
Encontro := true;
```

Ing Del y Gil

OPERADORES ARITMÉTICOS

+, -, * , /(División Real), DIV, MOD

7	2	7 DIV 2 = 3
1	3	7 MOD 2 = 1
↳	MOD	

OPERADORES LOGICOS

NOT	Not Operando
AND	oper1 and oper2
OR	oper1 OR oper2
XOR	oper1 XOR oper2

Ing Del y Gi l

OPERADORES RELACIONALES

>	,	>=	Mayor y Mayor que
<	,	<=	Menor y Menor que
=	,	<>	Igual y Diferente
IN			Pertenece a un conjunto

Ejm:

If (Resp in ['S', 's', 'N', 'n']) then

.....

If **not** (Resp in ['S', 's', 'N', 'n']) then

.....

If (Nota in [0..20]) then

...

IF 'Ala' _____ ' Alex' Then {Operador:< o >}

Ing Del y Gi l

OPERACIONES DE ENTRADA

Write('Expresion', Variable);
Writeln('expresion', variable);

Nombre:= 'María';
Write('El nombre es: ' :17, Nombre:9);

123456789012345678901234567890

El nombre es:	Maria
---------------	-------

Ing Del y Gi l

OPERACIONES DE SALIDA

Read(cad);
ReadLn(Cadena);

Write('Ingreso Km recorrido: ');
Read(Km) {variable real}
Write('El km recorrido es : ',Km:0:2);

Formato de salida de variables reales:

Variable:entera:decimal

Ing Del y Gi l

ESTILOS DE PROGRAMACIÓN

“El buen estilo de programación es, sin lugar a duda, una de las características más notables que DEBE tener un programador. Un programa con buen estilo es más fácil de leer, de corregir – si tiene errores – y de mantener. Aunque la experiencia proporciona el estilo hay una serie de reglas que se recomienda seguir desde el aprendizaje en programación.”

Ing. Dely Gil

ESTILOS DE PROGRAMACIÓN

Sangrado (Indentación)

Comentario

1. {comentario}
2. (*comentario*)

Líneas en blanco

Elección de nombres de identificadores significativos

Evitar punto y coma superfluos

Antes de la palabra END y UNTIL

Alinear Operaciones (o separadores) en líneas consecutivas.

Ing. Dely Gil

```
Program DemoPrimero;      {Cabecera Opcional}
Uses
  Crt,Dos;                {declaraciones}
Const
  PI    = 3.141592;
  IVA   = 12;
Type
  Cad20 = String[20];
  Notas = 1..10;
Var
  Salario : real;
  Numero  : Integer;
  Apellido : cad20;
  Pesos   : Notas;

Begin                    {Programa Principal}
  ClrScr;                {Borra la pantalla}
  Apellido = "";        {Sentencia de Asignación := }
  Write('Cuál es su primer apellido?');
  Readln(Apelido);      {Operaciones de Entrada}
  Writeln(' Escriba un número Sr: ', Apellido);
                        {Operaciones de Salida}

  Read(Numero);
  absoluto := abs(numero);
                        {No está declarada la variable
absoluto}
  raiz := sqr(numero);
  Writeln ('El cuadrado del número es ', Numero*
Numero);
  {cuadrado:= numero*numero
  writeln(' El cuadrado es : ' cuadrado);}
End;
```

Ing. Dely Gil

```
Program Sumar;
{Programa que suma dos números}
Uses crt;
Var
  Num1,Num2,suma : integer;
Begin
  {Programa Principal}
  ClrScr;           {Borra la pantalla}
  write('Ingrese primer número. ');
  read(Num1);
  write('Ingrese segundo número: ');
  read(Num2);
  suma := Num1 + Num2;
  write('La suma de ':15,Num1:4,' y ':3);
  writeln(Num2:4,'es ':4,suma:4);
  readkey;
End.
```

```
Program Permanencia;
{Programa que calcula el Índice de Permanencia de los
estudiantes de una Universidad de acuerdo a la fórmula:
 $IP = \sqrt{\text{Prom\_Notas}^2 + \text{Ac\_Com}^2 + \text{Edu\_Dep}^2}$  }
```

```
Uses
  Crt;
Var
  Prom_notas,Ac_com,Edu_Dep,Ind_Per: real;
Begin
  ClrScr;           {Borra la pantalla}
  Write('Ingrese el promedio de notas: ');
  readln(Prom_notas);
  write('Ingrese nota de acción comunitaria: ');
  readln(Ac_Com);
  write('Ingrese nota de educación deportiva: ');
  readln(Edu_Dep);
  Ind_Per := SQRT( SQR(Prom_Notas) + SQR(Ac_Com)
+SQR(Edu_Dep));
  write('El índice de permanencia es :', Ind_Per:0:2);
  readkey
End.
```

```
Program PlanViaje;
{ Calculo del tiempo y el costo estimado de un viaje en
automóvil utilizando las siguientes fórmulas:
  tiempo = distancia/velocidad
  litros consumidos = distancia/kilometros por litro
  costo del viaje = litros consumidos * costo por
litro}
Uses
  Crt;
Var
  distancia,velocidad,tiempo,costolit,kilomporlit,: real;
  litusados,costotal :real;

Begin
  ClrScr;
  Writeln ('Calculo del tiempo y costo del viaje: ');
  Write('Ingrese distancia(kms) y velocidad(kms/h)');
  Read (distancia,velocidad);
  Writeln ('Distancia = ',distancia :8:2, 'kiloms');
  Writeln ('Velocidad promedio = ', velocidad :0:2, ' kms
por hora');
  tiempo := distancia / velocidad;
  Writeln ('tiempo del viaje = ', tiempo :0:2, ' horas');
  Write('Ingrese Kilómetro por litro y el Costo por Litro');
  Read (kilomporlit,costolit);
  writeln ('Cantidad de kilómetro = ', kilomporlit :0:2,
'kilom por litros');
  writeln ('Costo por litros = ', costolit :0:2, 'peso');
  litusados := distancia / kilomporlit;
  costotal := litusados * costolit;
  writeln ('Costo estimado del viaje = ', costotal :0:2, '
bolivares');
  writeln('Presione tecla para terminar');
  readkey
End.
```

EJERCICIOS PROPUESTOS

- Φ Realizar un programa que ingrese 2 números, calcule la suma y el promedio.
- Φ Construir un programa que lea números enteros representativos de día, mes y año e imprima esta fecha bajo la forma: dd / mm / aa
- Φ Realizar un programa que lea dos números enteros e imprima su suma.
- Φ Elaborar un programa que calcule el área de un rectángulo conociendo su base y su altura, y sabiendo que: Area = Base * Altura.
- Φ Elaborar un programa que calcule el área de un triángulo rectángulo dada la altura y la base, sabiendo que: Area = (Base * Altura)/2
- Φ Elaborar un programa que calcule el volumen y el área de un paralelepípedo de dimensiones L(argo), A(ncho) y H(altura), sabiendo que: Area Sup = 2 * (L*A +L*H+A*H) y V = L *A**H
- Φ Realizar un programa que calcule el total de la compra de tres pares de zapatos, los cuales tiene un 12,5% de descuento pero tiene un recargo de 14,5% de impuesto.

<p>Cuál de los siguientes identificadores son válidos:</p> <ul style="list-style-type: none"> a) Renta b) Alquiler c) Constante d) Tom's e) Dos Pulgadas f) C3P0 g) Bienvenidos#5 h) Elemento i) 4a2D2 j) 13Nomre k) Nombre_Apellidos l) NombreApellidos 	<p>Si A=6, B=2 y C=3, encontrar el valor de las siguientes expresiones:</p> <ul style="list-style-type: none"> a) A—B+C b) A*B DIV C c) A DIV B +C d) A*B MOD C e) A+B MOD C f) A DIV B DIV C 	<p>Codifique las siguientes expresiones:</p> <ul style="list-style-type: none"> a) $\frac{M}{N} + 4$ b) $M + \frac{N}{P - Q}$ c) $\frac{\text{Sen}x + \text{cos}y}{\text{Tan}x}$ d) $\frac{M + N}{P - Q}$ e) $\frac{M + \frac{N}{P}}{Q - \frac{R}{5}}$ f) $\frac{-b + \sqrt{b^2 - 4ac}}{2a}$ <p style="text-align: right;">Ing Del y Gil</p>
--	---	--

Palabras reservadas y utilizadas en el Lenguaje PASCAL

FUNCIONES PREDEFINIDAS

Add	File	Previous	Text	pred(x)
Again	Fill	Printer	True	random(x)
And	Find	Procedure	Trunc	round(x)
Append	For-Downto-Do	Program	Type	sin(x)
Array	For-To-Do	Random	Unit	sqr(x)
Assign	Function	Read	Uppcase	sqrt(x)
Background	Get	Readkey	Value	succ(x)
Begin	Graph	Record	Var	trunc(x)
Boolean	Halt	Rename	Watch	upcase(x)
Break	If... Then ..Else...	Repeat... Until	While	
Byte	In	Replace	Window	
Case	Integer	Reset	With Do	
Change	Interface	Rewrite	Word	
Char	Into	Round	Write	
Close	Keypressed	Run	Zoom	
Concat	Label	Save As		
Const	Length	Screen	abs(x)	
Continue	Longint	Search	arctan(x)	
Copy	Make	Seek	chr(x)	
Cut	Memory	Set	cos(x)	
Debug	Move	Shortint	dec(x)	
Delay	New	Size	exp(x)	
Delete	Next	Sizeof	frac(x)	
Dos	Object	Sound	inc(x)	
Edit	Open	Source	int(x)	
End	Over	Step	length(x)	
Erase	Paste	String	odd(x)	
Exit	Point	System	ord(x)	
False				

- TURBO PASCAL - Opciones de la Barra de Acceso rápido en la pantalla inicial.

OPCION	DESCRIPCIÓN
F1	Obtener ayuda sobre el contexto actual.
F2	Guardar un Archivo.
F3	Abrir un Archivo.
F4	Ejecutar el programa y a continuación detener en la posición actual del cursor.
F5	Conmuta ventana de edición entre pantalla completa y parcial.
F6	Mover a la siguiente ventana en la lista de ventanas abiertas.
F7	Traza de un programa paso a paso.
F8	Traza de un programa paso a paso a través de procedimientos y funciones.
F9	Construir el programa compilando todos los archivos requeridos.
F10	Cambiar a líneas de menús.
ALT+F1	Visualizar pantalla de ayuda más reciente.
ALT+F3	Cerrar ventana activa.
ALT+F5	Cambiar entre pantalla de salida y ventana de edición.
ALT+F9	Compilar archivo.
CTRL+F1	Obtener ayuda sobre tópico del lenguaje seleccionado.
CTRL+F2	Reiniciar el programa para ejecutar desde el principio.
CTRL+F3	Visualizar pila de llamada durante ejecución del programa.
CTRL+F4	Evaluar expresión, valor o función durante ejecución del programa.
CTRL+F7	Añadir variable de observación.
CTRL+F8	Bascular punto de interrupción.
CTRL+F9	Ejecutar programa, compilando primero si es necesario para detectar errores.
Alt-X	Salir.