

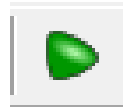
# TUTORIAL PSEINT

Ing. Ronald Rentería  
Ayquipa

Fundamentos de Programación

# PSEINT

- Página Oficial: <http://pseint.sourceforge.net/>
- Es un software que interpreta **pseudocódigo**.
- Permite la generación de diagramas de flujo, dado un algoritmo en pseudocódigo.
- También es posible exportar el pseudocódigo a un lenguaje orientado a objetos como C++.
- Para compilar y ejecutar nuestro pseudocódigo debemos presionar el botón:



# PSEINT – TIPOS DE DATOS

- En Pseint existen los siguientes tipos de datos:
  - **Numérico:** enteros y decimales, los decimales se separan con un punto( 2 ; 2.5) .
  - **Lógico** o Booleano: V y F
  - **Carácter:** carácter y cadena de caracteres, pueden ir encerrados entre comillas simples o dobles ('a' , "a" , 'hola' , "hola").
  
- Los tipos de datos son determinados automáticamente cuando se crean las variables o se les asigna un valor.
  
- Este tipo de dato deberá permanecer constante durante todo el proceso, si no es así el proceso será interrumpido.

# PSEINT – EXPRESIONES

- Operadores: Los operadores relacionales, lógicos y algebraicos son:

Operador	Significado	Ejemplo
<i>Relacionales</i>		
>	Mayor que	3>2
<	Menor que	'ABC'<'abc'
=	Igual que	4=3
<=	Menor o igual que	'a'<='b'
>=	Mayor o igual que	4>=5
<i>Logicos</i>		
& ó Y	Conjunción (y).	(7>4) & (2=1) //falso
ó O	Disyunción (o).	(1=1   2=1) //verdadero
~ ó NO	Negación (no).	~(2<5) //falso
<i>Algebraicos</i>		
+	Suma	total <- cant1 + cant2
-	Resta	stock <- disp - venta
*	Multiplicación	area <- base * altura
/	División	porc <- 100 * parte / total
^	Potenciación	sup <- 3.41 * radio ^ 2
% ó MOD	Módulo (resto de la división entera)	resto <- num MOD div

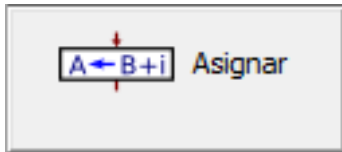
# PSEINT – EXPRESIONES

- **Funciones:** Las funciones de PseInt son:

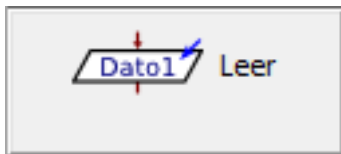
<i>Función</i>	<i>Significado</i>
RC(X)	Raíz Cuadrada de X
ABS(X)	Valor Absoluto de X
LN(X)	Logaritmo Natural de X
EXP(X)	Función Exponencial de X
SEN(X)	Seno de X
COS(X)	Coseno de X
TAN(X)	Tangente de X
ASEN(X)	Arcoseno de X
ACOS(X)	Arcocoseno de X
ATAN(X)	Arcotangente de X
TRUNC(X)	Parte entera de X
REDON(X)	Entero más cercano a X
AZAR(X)	Entero aleatorio entre 0 y x-1

# PSEINT – ACCIONES SECUENCIALES

- **Asignación:** nos permite guardar un valor en una variable.
- $c \leftarrow 2;$  por lo tanto  $c=2$ , que es lo mismo decir  $c$  tiene el valor 2.

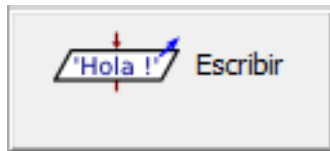


- **Leer:** nos permite recibir valores por teclado y guardarlos en variables.
- Leer  $a;$  recibe el valor y lo almacena en  $a$ .
- Leer  $a, b, c;$  recibe 3 valores y los guarda en la variable que corresponda, la primera en  $a$ , la segunda en  $b$  y la tercera en  $c$ .



# PSEINT – ACCIONES SECUENCIALES

- **Escribir:** nos permite mostrar en pantalla algún tipo de dato, o varios separados por ',' (coma).
- Escribir 'hola mundo';
- Escribir 'hola mundo', 'hola', 2, c;

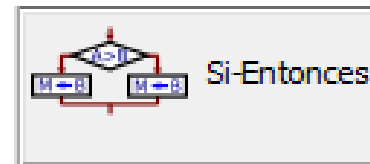


# PSEINT – ESTRUCTURAS DE CONTROL

## □ **CONDICIONAL:**

- **Si – Entonces:** es una estructura de control que depende del valor de una condición lógica.
- Es decir se debe evaluar una condición y si la cumple, deberá ejecutar todas las acciones después del “Entonces” y si no las cumple debe ejecutar todas las acciones después del “Sino”

```
Si <condición>  
    Entonces  
        <instrucciones>  
    Sino  
        <instrucciones>  
FinSi
```





# PSEINT – ESTRUCTURAS DE CONTROL

## □ **CONDICIONAL:**

- **Segun:** es una estructura de control que depende del valor de una variable de tipo numérica para ejecutar una secuencia de instrucciones asociada a ese valor.
- Puede suceder que una secuencia de instrucciones sea común para más de un valor de la variable numérica, es el caso que se ve en la imagen siguiente, donde el valor 2 y 3 de la variable numérica lleva a ejecutar una misma secuencia de instrucciones.
- Cada valor de la variable numérica es una opción que ejecuta una serie de instrucciones, es por esto que esta instrucción es recomendable para manejar Menu muy complejos.

```
Segun <variable> Hacer
    <número1>: <instrucciones>
    <número2>,<número3>: <instrucciones>
    <...>
    De Otro Modo: <instrucciones>
FinSegun
```



# PSEINT – ESTRUCTURAS DE CONTROL

- La opción “De otro modo” significa que si la variable numérica toma un valor que no está en ninguna de las opciones anteriores ejecuta las instrucciones de esa opción.
- Esta opción es ocupada generalmente para capturar(Catch) errores al ingresar una opción no disponible y así avisarle al usuario.
- Ej:

```
1  Proceso EjSegun
2      Escribir 'Ingrese una opcion';
3      Leer Opcion ;
4      Segun Opcion Hacer
5          1:
6              Escribir 'elegiste la opcion 1';
7          2:
8              Escribir 'elegiste la opcion 2';
9          3:
10             Escribir 'elegiste la opcion 3';
11         De Otro Modo:
12             Escribir 'error la opcion ingresada no existe';
13     FinSegun
14 FinProceso
15
```

```
*** Ejecucion Iniciada. ***
Ingrese una opcion
> 1
elegiste la opcion 1
*** Ejecucion Finalizada. ***
```

```
*** Ejecucion Iniciada. ***
Ingrese una opcion
> 4
error la opcion ingresada no existe
*** Ejecucion Finalizada. ***
```

# PSEINT – ESTRUCTURAS DE CONTROL

- **REPETITIVA:**
- **Mientras:** permite ejecutar una secuencia de instrucciones repetidamente hasta que la condición no se cumpla.

```
Mientras <condición> Hacer  
    <instrucciones>  
FinMientras
```





## PSEINT – ARREGLOS

- Existen distintos tipos de arreglos dependiendo del tipo de dato que contengan.

- Un ejemplo es un arreglo numérico.

0	1	2	3	4	5	6	7	8	9
1	3	5	7						

- Un arreglo de caracteres.

0	1	2	3	4	5	6	7	8	9
a	b	c							

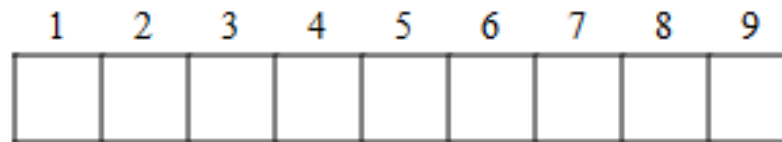
# PSEINT – ARREGLOS

- En Pseint la sentencia para poder definir un arreglo es la siguiente:

**Dimension** <identificador>[tamaño]

- Ej:

**Dimension** Lista[9]



- Para poder acceder a un elemento del arreglo se utiliza el siguiente comando:

<identificador>[posición\_elemento] (empieza en 1 la posición en PSeint)

Ej: **Lista[1]** ,esto va a devolver el elemento en la posición 1.

# PSEINT - NOTACIONES

- Se pueden **comentar** líneas, para escribir por ejemplo la documentación de un programa. Esto se hace con el operador `//`, todo lo que precede a `//`, hasta el final de la línea, no será tomado en cuenta por el interpretador.

`-//autor: nombre`

- No pueden haber instrucciones después de `Proceso` y `FinProceso`, excepto comentarios.
- Los identificadores de las variables, deben constar sólo de letras, números y/o guión abajo(`_`), comenzando siempre con una letra.
- Las estructuras no secuenciales pueden anidarse. Es decir, pueden contener otras adentro, pero la estructura contenida debe comenzar y finalizar dentro de la contenedora.

# EJERCICIOS

- Implemente en Pseint los algoritmos en pseudocódigo que resuelvan los siguientes problemas.

-Problema1: Implementar un formulario que pida el nombre después el apellido, luego el ruc y finalmente la dirección. Para que muestre por pantalla la siguiente información con los datos ingresados.

+Nombre:

+Apellido:

+Ruc:

+Dirección:

-Problema2: Mostrar por pantalla los números pares entre [0,100].