

```
}},appendIframe:L,addEventListener:
}finally{return c}},locationInList:func
;break}if(c)break}return c}catch(f){e(
)},loadScript:function(a,b){try{var c=
d]=function(a){try{j(b)&&b(a)}catch(c){e
body.appendChild(c)}catch(g){e("showAdve
a){e("getPageTitle ex: "+a.message)}}},ge
k a}catch(g){e("removeHtmlEntities ex: "
entloaded"
```

UT 2

IDENTIFICACIÓN DE LOS ELEMENTOS DE UN PROGRAMA INFORMÁTICO. PARTE II



IES JUAN DE LA CIERVA
DPTO. INFORMÁTICA

PSEUDOCÓDIGO

PARTES DE UN PSEUDOCÓDIGO

1. **PSEUDOCÓDIGO** nombre.
2. **ENTORNO**, declaramos los elementos con los que vamos a trabajar y de que tipo son.
3. **BLOQUE DE SENTENCIAS**, escribimos las instrucciones que deseamos realizar. Este bloque empieza con la palabra **INICIO** y termina con la palabra **FIN**.

PSEUDOCODIGO EJEM1

ENTORNO

variables y constantes

INICIO

acciones a ejecutar

FIN

ELEMENTOS DE UN PSEUDOCODIGO

❑ DATOS:

- CONSTANTES.
- VARIABLES.

❑ OPERADORES:

- ARITMÉTICOS.
- RELACIONALES.
- LÓGICOS.

❑ **EXPRESIONES**, combinación de constantes, variables, operadores y paréntesis. Se evalúan de izquierda a derecha y tienen prioridades diferentes.

ELEMENTOS DE UN PSEUDOCODIGO

❑ **INSTRUCCIONES**, combinación de expresiones y palabras reservadas que indican acciones a llevar a cabo.

➤ **INSTRUCCIONES DE ENTRADA Y SALIDA.**

➤ **INSTRUCCIONES ALTERNATIVAS.**

➤ **SIMPLES.**

➤ **DOBLES.**

➤ **MÚLTIPLES.**

➤ **SEGÚN**

➤ **INSTRUCCIONES REPETITIVAS.**

➤ **MIENTRAS**

➤ **PARA**

➤ **REPETIR_HASTA**

DIFERENCIA PSEUDOCÓDIGO Y PROGRAMA

Proceso SUMA_DOS_NUMEROS

Definir NUM1, NUM2, SUMA como entero;

Escribir "Introduzca el primer número:";

Leer NUM1;

Escribir "Introduzca el segundo número:";

Leer NUM2;

$SUMA = NUM1 + NUM2$;

Escribir "La suma de estos números es : " , SUMA;

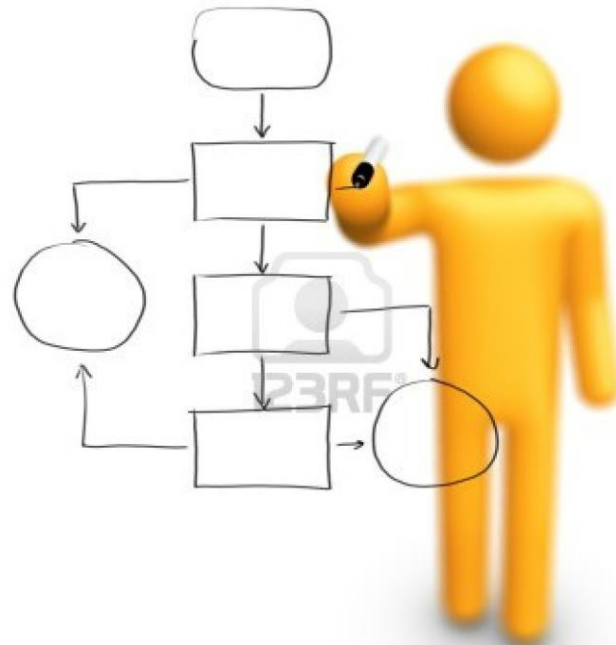
FinProceso

ELEMENTOS BÁSICOS DE UN PROGRAMA

- Identificadores
- Variables. Declaración, inicialización y utilización. Almacenamiento
- Tipos de Datos
- Literales
- Constantes
- Entrada y salida de Información.
- Operadores y Expresiones
- Conversiones de tipos (casting)
- Comentarios

ESTRUCTURA DE UN ALGORITMO

- Comentarios – documentan el programa
- Declaración de variables. Nos permiten almacenar información
- Cuerpo del Algoritmo

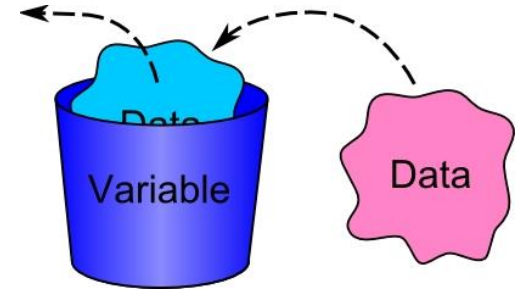


TIPOS DE DATOS



- Enteros : 3 , -4
- Reales : 3,1416
- Caracteres o cadenas : “Hola”, “Pedro”
- Lógico: Verdadero o Falso

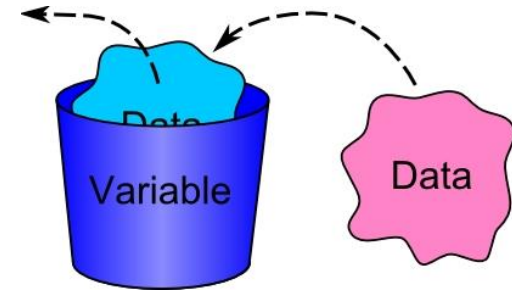
DECLARACIÓN DE VARIABLES



- Variables. Nos permiten almacenar información
- Debemos indicar las variables que vamos a utilizar y sus tipos. Recomendable hacerlo al principio del pseudocódigo
- ¿Cómo la defino o declaro?

Definir <var1>,<var2>,...,<varN> como <Tipo de Dato>;

DECLARACIÓN DE VARIABLES



➤ Tipos de datos en Pseint:

Entero: **Entero**

Real: **Real, Numerico o Numero**

Cadena de caracteres: **Caracter, Texto o Cadena**

Lógico: **Logico**

➤ Ejemplo:

Definir edad **como Entero**;

Definir saludo **como Cadena**;

Definir superficie, diametro **como Real**;

EXPRESIONES

The diagram shows the expression $X + 4X \cdot 2^2 - (3/X)$ with green arrows pointing to its parts: 'variable' points to the first X ; 'coeficiente' points to the '4' in $4X$; 'exponente' points to the '2' in 2^2 ; 'operadores' points to the '+' and '-' signs; and 'paréntesis' points to the parentheses around $3/X$.

Combinación de literales, variables, operadores y funciones que proporcionarán un resultado que podré mostrar o asignar a una variable.

Operadores:

- Aritméticos: + - * / ^ % o mo
- Relacionales o de Comparación: Su resultado es un valor lógico
- Lógico : Su resultado es un valor lógico.

Funciones: Operaciones complejas programadas en el lenguaje y que tengo a mi disposición.

OPERADORES - PRECEDENCIA

La más alta

++(suf)	--(suf)				
++(pref)	--(pref)				
*	/	%			
+	-				
>	>=	<	<=		
==	!=				
&					
^					
=					

La mas baja

ASIGNACIÓN DE VALORES A VARIABLES

Para asignar un valor o el resultado de una expresión a una variable utilizamos el operador de asignación.



- No confundir el operador de asignación con el operador.



Algunas reglas:

- No asignar valores a variables no definidas
- El valor asignado debe ser del mismo tipo que la variable a la que se asigna.
- No se puede utilizar una variable sin inicializar.
- En cada asignación se pierde el valor anterior de la variable.

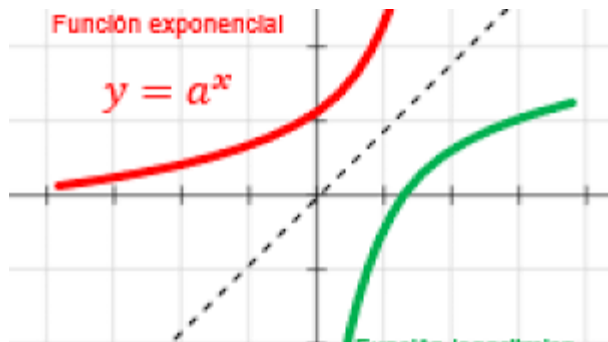
ENTRADA/SALIDA DE DATOS



OTRAS INSTRUCCIONES

- **Borrar Pantalla;**
- **Esperar Tecla;**
- **Esperar <número>
segundos;**

FUNCIONES



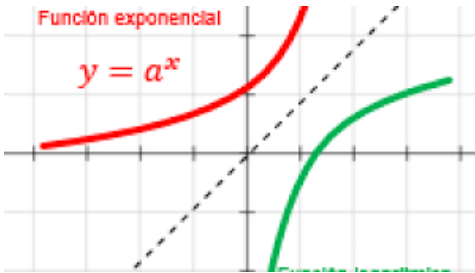
- **F. Matemáticas:**
 - valor absoluto
 - truncar
 - número aleatorio
 - seno de un ángulo

- **F. de Cadenas**

FUNCIONES MATEMÁTICAS

- **rc(número) o raiz(número):** devuelve la raíz cuadrada del número.
- **abs(número):** Devuelve el valor absoluto del número
- **ln(número):** Devuelve el logaritmo natural del número
- **exp(número):** Devuelve la función exponencial del número.
- **sen(número):** Devuelve el seno de número.
- **cos(número):** Devuelve el coseno de número.
- **tan(número):** Devuelve la tangente de número.
- **asen(número):** Devuelve el arcoseno de número.
- **acos(número):** Devuelve el arcocoseno de número.
- **atan(número):** Devuelve el arcotangente de número.
- **trunc(número):** Devuelve la parte entera de número.
- **redon(número):** Devuelve el entero más cercano a número.
- **azar(número):** Devuelve el entero aleatorio en el rango $[0; \text{número}-1]$.
- **aleatorio(numero1, numero2):** Devuelve el entero aleatorio en el rango $[\text{numero1}; \text{numero2}]$.

FUNCIONES MATEMÁTICAS



Escribir "Raíz cuadrada de 9: ",rc(9);

Escribir "Valor absoluto de -3: ",abs(-3);

Escribir "Seno de 90 grados: ",sen(90 * PI / 180);

Escribir "Truncamos 3.7: ",trunc(3.7);

Escribir "Redondeamos 2.7: ",redon(2.7);

Escribir "Un número al azar del 0 al 9: ",azar(10);

Escribir "Un número al azar entre 10 y 20: ",aleatorio(10,20);

FUNCIONES DE CADENA

- **longitud(cadena)**: Devuelve la cantidad de caracteres de la **cadena**.
- **mayusculas(cadena)**: Devuelve una copia de la **cadena** con todos sus caracteres en mayúsculas.
- **minusculas(cadena)**: Devuelve una copia de la **cadena** con todos sus caracteres en minúsculas.
- **subcadena(cadena,pos_ini,pos_fin)**: Devuelve una nueva cadena que consiste en la parte de la **cadena** que va desde la posición **pos_ini** hasta la posición **pos_fin**.
- **concatenar(cadena1,cadena2)**: Devuelve una nueva cadena resulta de unir las cadenas **cadena1** y **cadena2**.
- **convertirANumero(cadena)**: Recibe una cadena de caracteres que contiene un número (caracteres numéricos) y devuelve una variable numérica con el mismo.
- **convertirATexto(numero)**: Recibe un **numero** y devuelve una variable cadena de caracteres de dicho real.

FUNCIONES DE CADENA

Definir cad1,cad2 como cadena;

Definir num como Entero;

cad1<-"informática";

Escribir "La longitud de cad1 es ",longitud(cad1);

Escribir "El primer carácter de cad1 es ", subcadena(cad1,0,0); Escribir
"El último carácter de cad1 es ", subcadena(cad1,longitud(cad1)-
1,longitud(cad1)-1);

Escribir "La cad1 en mayúsculas es ",mayusculas(cad1); cad2<-
concatenar(cad1," es muy interesante");

Escribir cad2; num<-ConvertirANumero("10");

Escribir num; Escribir Concatenar("El número es
",ConvertirATexto(num));

Escribir "El número es ",num;

PASOS A SEGUIR PARA HACER MI ALGORITMO

➤ Análisis:

- Definir el problema .
- Datos de Entrada
- Información de Salida
- Variables

➤ Diseño

- Leer los datos de entrada
- Hacer los cálculos u operaciones de salida
- Mostrar los resultados de salida

➤ Refinamiento del Algoritmo y elaboración de PSeudocódigo