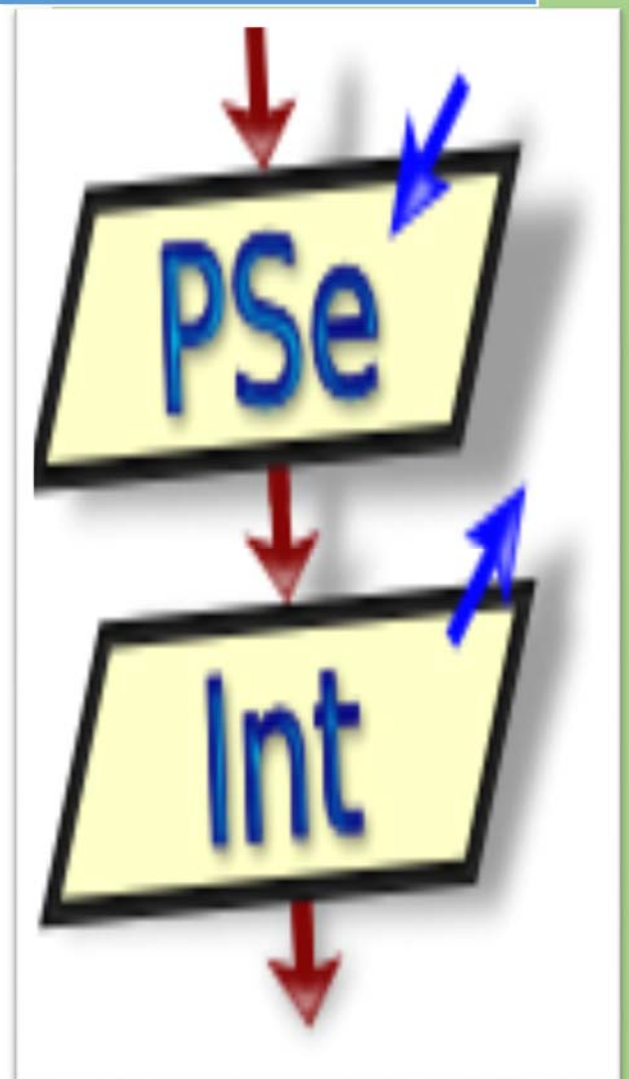


2013

UNIDAD II: Estructuras Secuenciales



ESTRUCTURA SECUENCIAL

1.1. Introducción

La estructura secuencial permite representar aquellas instrucciones que se ejecutan una tras otra, en secuencia; o sea, instrucciones en que la salida de una es la entrada de la próxima instrucción. También se puede decir que son una o más instrucciones seguidas, donde ninguna de ellas hace que se bifurque el control de ejecución del algoritmo, o que se repita la ejecución de una o más instrucciones. Mediante esta estructura se pueden representar instrucciones de asignación, entrada y salida de datos e invocación a subprogramas. Para diferenciar una de otra, se añaden características intrínsecas de cada instrucción.

Ésta, como todas las estructuras de la programación estructurada, tiene una sola entrada y una sola salida.

Representación

```
Proceso NombreProceso
    <instrucción 1>
    <instrucción 2>
    <instrucción 3>
    .
    .
    .
    <instrucción n>
FinProceso
```

Ejemplo

```
Proceso suma
    Escribir 'Ingrese dos enteros: ';
    Leer a,b;
    c<- a-b;
    Escribir 'La suma es: ',c;
FinProceso
```

1.2. Definición de variables

La instrucción definir permite explicitar el tipo de una o más variables. Esta definición puede ser opcional u obligatoria dependiendo de la configuración del lenguaje. La sintaxis

Definir <var1>, <var2>, ..., <varN> Como [REAL/ENTERO/LOGICO/CARACTER];

Una variable debe definirse antes de ser utilizada por primera vez. Los arreglos, se definen utilizando su identificador (sin subíndices ni dimensiones) antes o después de dimensionarlos, y el tipo aplica para todos los elementos del mismo (ya que se trata de una estructura de datos homogénea).

Los tipos posibles son NUMERO, NUMERICO, REAL, ENTERO, LOGICO, CARACTER, TEXTO, CADENA.

NUMERO, NUMERICO y REAL son sinónimos para el tipo de datos numérico básico, que puede almacenar tanto números reales como enteros. El tipo ENTERO es una especialización que sólo permite almacenar valores enteros; cualquier valor no entero que se lea o asigne en una variable de este tipo será truncado.

Una variable de tipo LOGICO sólo puede tomar los valores VERDADERO y FALSO, pero cuando se lee una variable ya definida como lógica, el usuario puede ingresar también las abreviaciones V y F.

CARACTER, TEXTO y CADENA son sinónimos para definir variables de tipo carácter. Estas pueden contener cero, uno o más caracteres arbitrarios y no tienen una longitud máxima. Si se declara una variable de este tipo y en una lectura el usuario ingresa un número o un valor lógico, se asignará una cadena que contiene el texto ingresado (ejemplo: "1", "VERDADERO", etc).

Si se intenta asignar a una variable ya definida un dato de un tipo incorrecto se producirá un error en tiempo de ejecución.

1.3. INSTRUCCIÓN DE ASIGNACION

Por medio de esta instrucción se asigna a una variable el resultado de evaluar una expresión. Al lado izquierdo de la instrucción está la variable, al derecho la expresión y uniéndolas el signo "<-", denominado **operador de asignación**.

Al ejecutarse la instrucción se evalúa la expresión, la cual da como resultado un único valor; éste es llevado a la dirección de memoria asignada a la variable que recibe la información. Al hacerse la asignación es destruida cualquier otro valor que tenga la variable. El dato o valor que arroje la expresión debe ser del mismo tipo que la variable: si el resultado es numérico la variable debe ser numérica, si es lógico o carácter la variable también lo debe ser.

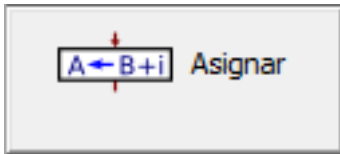
Existen dos operadores de asignación alternativos que pueden utilizarse indistintamente en cualquier caso, pero la habilitación del segundo (=) depende del perfil de lenguaje seleccionado.

<variable> := <expresión>;

<variable> = <expresión>;

FORMATO

<variable> <- <expresión>

En el área de Comandos del PSeint**Ejemplos**

SALDO <- VFINAL – VINICIAL

RESUL <- (X > Y) ^ (B > C)

SUMA <- 0

VARIABLE1 <- VARIABLE2

NOMBRE <- 'CARMEN'

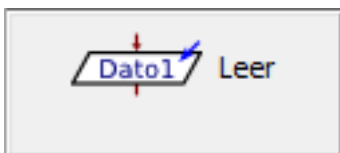
1.4. INSTRUCCIÓN DE ENTRADA DE DATOS

Los datos de entrada son los valores necesarios para que la computadora haga los cálculos pertinentes.

Una instrucción de entrada de datos es una orden, para que desde un medio externo se introduzcan valores y dichos datos sean llevados a las direcciones de memoria, asignadas a las variables utilizadas para almacenar los datos de entrada en el proceso.

FORMATO

Leer <lista de variables separadas por coma>;

En el área de Comandos del PSeint**Ejemplo**

Leer CODIGO, NOMBRE;

Los valores que se introducen deben estar en el mismo orden de las variables en la lista.

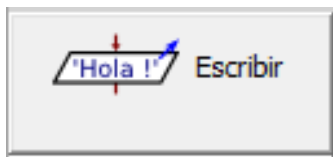
INSTRUCCIÓN DE SALIDA DE DATOS

Mediante la instrucción de salida se extraen valores de la memoria hacia un medio externo de salida permitido. Se permite acompañar los resultados con comentarios explicativos de los valores que se muestran en el medio de salida.

FORMATO

Escribir <lista de variables separadas por comas o comentarios>;

En el área de Comandos del PSeint



La instrucción Escribir permite mostrar valores al ambiente.

Esta instrucción informa al ambiente (en este caso escribiendo en pantalla) los valores obtenidos de evaluar N expresiones. Dado que puede incluir una o más expresiones, mostrará uno o más valores. Si hay más de una expresión, se escriben una a continuación de la otra sin separación, por lo que el algoritmo debe explicitar los espacios necesarios para diferenciar dos resultados si así lo requiere.

Si en algún punto de la línea se encuentran las palabras clave "SIN SALTAR" o "SIN BAJAR" los valores se muestran en la pantalla, pero no se avanza a la línea siguiente, de modo que la próxima acción de lectura o escritura continuará en la misma línea. En caso contrario, se añade un salto de línea luego de las expresiones mostradas.

Escribir Sin Saltar <expr1> ,..., <exprN>;

Escribir <expr1>, ..., <exprN> **Sin Saltar**;

Ejemplos

Proceso prueba

Escribir 'Ingrese tres valores:';

Leer A,B,C;

Escribir A;

Escribir B;

Escribir C;

Escribir 'Los valores son: ',A,' ',B,' ',C;

Escribir 'UNIVERSIDAD AUTONOMA DE NICARAGUA';

Escribir 'El valor de A es:',A,' El de B es:',B,' Y el de C es:',C

FinProceso

Al ejecutarse cada una de las instrucciones anteriores, y si las variables A, B y C contienen los valores: A = 20; B = 30.75; C = -255, las salidas serán:

```

*** Ejecución Iniciada. ***
Ingrese tres valores:
> 20
> 30.75
> -255
20
30.75
-255
Los valores son: 20 30.75 -255
UNIVERSIDAD AUTONOMA DE NICARAGUA
El valor de A es:20 El de B es:30.75 Y el de C es:-255
*** Ejecución Finalizada. ***

```

Cuando en una instrucción de salida se tiene un grupo de caracteres entre comillas éste será impreso sin ninguna modificación en el medio de salida, y lo que aparezca sin comillas se asume que es una variable; por lo tanto, se extraerá de la memoria el contenido de ésta.

Ejercicio resuelto

Hacer un algoritmo que calcular el número de pulsaciones que una persona debe tener por cada 10 segundos de ejercicio, si la fórmula es:

$$\text{num_pulsaciones} = (220 - \text{edad})/10$$

Datos de entrada

- Edad de la persona.

Datos de salida

- Numero de pulsaciones

Definición de variables

ed: Edad de la persona

npuls: Numeros de pulsaciones

Pseudocódigo

Proceso num_pulsaciones

Definir ed Como Entero;

Definir npuls como Real;

Escribir "Ingresar la edad de la persona";

Leer ed;

 $npuls \leftarrow (220 - ed) / 10$;

Escribir "El numero de pulsaciones de la persona es: ",npuls;

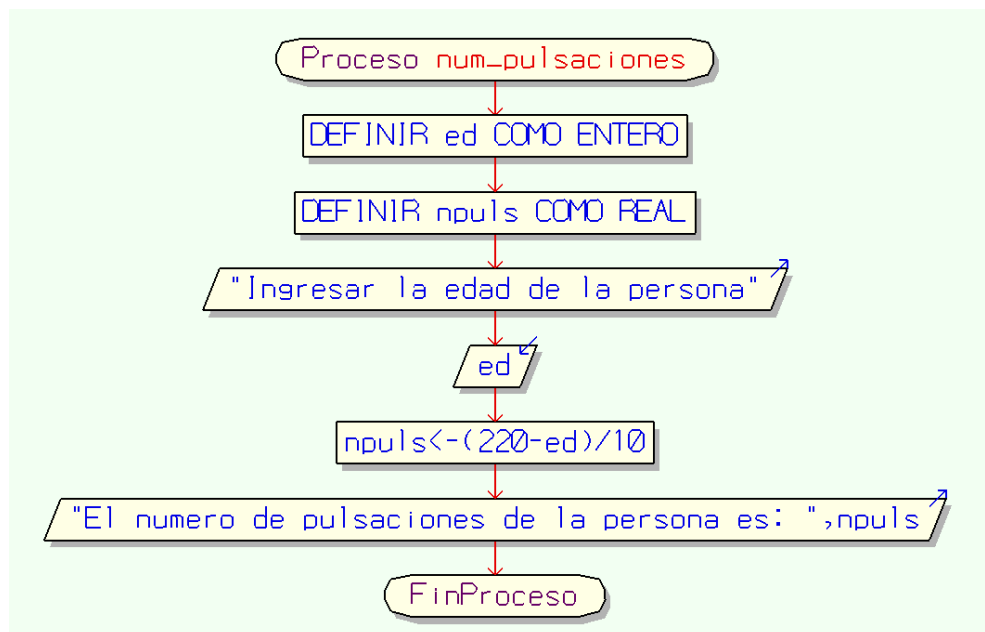
FinProceso

Salida

```

PSeInt - Ejecutando proceso NUM_PULSACIONES
*** Ejecución Iniciada. ***
Ingresar la edad de la persona
> 25
El numero de pulsaciones de la persona es: 19.5
*** Ejecución Finalizada. ***

```

Diagrama de flujo

Ejercicio resuelto

Hacer un algoritmo que, dados los dos lados diferentes de un rectángulo, encuentre el perímetro y el área del mismo. Al ser un rectángulo, conociendo los valores de los dos lados diferentes podemos obtener los resultados solicitados.

Datos de entrada

- Valor de un lado.
- Valor del otro lado.

Datos de salida

- El valor del perímetro.
- El valor del área del rectángulo.

Proceso

Los cálculos necesarios para obtener los resultados partiendo de los datos de entrada, son:

- Perímetro = suma de los cuatro lados del rectángulo
- Area = lado que representa la base * lado que representa la altura

Definición de variables

L1: Valor del lado que representa la base.
L2: Valor del lado que representa la altura.
P: Perímetro.
AREA: Área del rectángulo.

Pseudocódigo

Proceso rectangulo

```
Escribir 'Introduzca dos lados del rectangulo:';  
Leer L1, L2;  
P <- 2 * L1 + 2 * L2;  
AREA <- L1 * L2;  
Escribir 'EL PERIMETRO ES: ', P;  
Escribir 'EL AREA ES: ', AREA;
```

FinProceso

Prueba de escritorio

Si el registro de entrada es:

25 el valor de L1, 10 el valor de L2

L1	L2	P	AREA
25	10	70	250

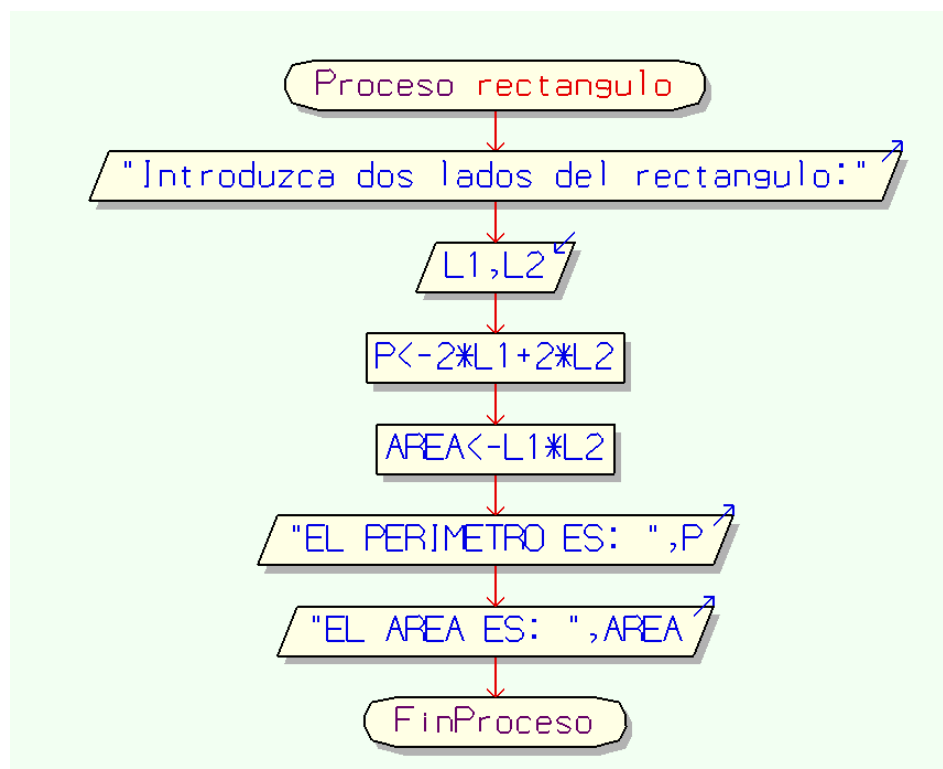
Salida

```

PSeInt - Ejecutando proceso RECTANGULO
*** Ejecución Iniciada. ***
Introduzca dos lados del rectangulo:
> 25
> 10
EL PERIMETRO ES: 70
EL AREA ES: 250
*** Ejecución Finalizada. ***

```

Diagrama de flujo



Ejercicio resuelto

Un problema que se presenta frecuentemente en las "finanzas domésticas" es el determinar cuánto dinero se acumulará en una cuenta en el banco después de n años si conocen la cantidad depositada inicialmente (P) y el tanto por ciento anual de interés compuesto aplicado (r). La respuesta se determina mediante la fórmula:

$$F = P(1 + i)^n$$

En donde F representa la cantidad futura de dinero; i es representación decimal del tanto por ciento de interés, esto es, $i = r / 100$ (por ejemplo, un interés de $r = 5$ por 100, se corresponde con $i = 0.05$).

Calcule la cantidad futura de dinero.

Datos de entrada

- Cantidad depositada inicialmente.
- Tanto por ciento anual de interés compuesto aplicado.
- Años en que estará depositado el dinero.

Datos de salida

- Cantidad futura de dinero.

Definición de variables

- p representa la cantidad depositada inicialmente.
- r representa el tanto por ciento anual de interés compuesto aplicado.
- i es representación decimal del tanto por ciento de interés
- n almacena el número de años en que estará depositado el dinero.
- f almacena la cantidad futura de dinero.

Pseudocódigo

Proceso finanzasdomesticas

```
//problema del interés compuesto
//leer datos de entrada Definir p,i,f Como real;
Definir r,n Como entero;

Escribir "Por favor, introduce la suma inicial P:";
Leer p;
Escribir "Por favor, introduce el Íteres r:";
Leer r;
Escribir "Por favor, introduce el numero de anyos n:";
Leer n;

//calcular i y f
i <- r /100;
f <- p * (1+i)^n;
```

```
//escribir salida
Escribir "El valor final (F) es:",f;
```

FinProceso

Prueba de escritorio

p	r	n	i	f
15000	6	7	0.06	22554.4538848704

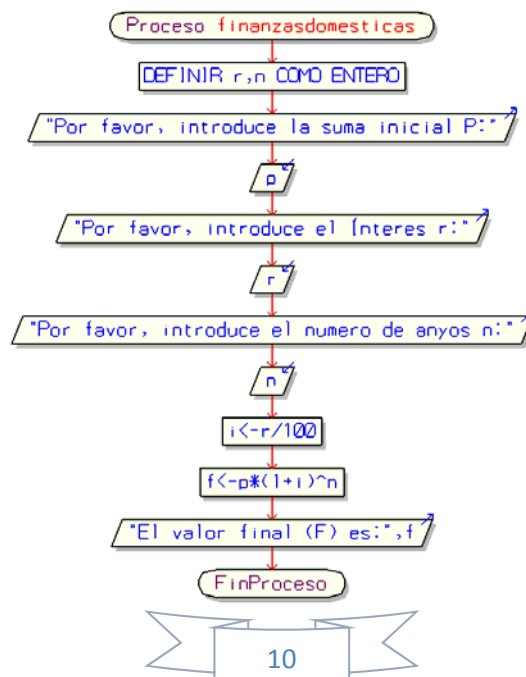
Salida

```

PSeInt - Ejecutando proceso FINANZASDOMESTICAS
*** Ejecución Iniciada. ***
Por favor, introduce la suma inicial P:
> 15000
Por favor, introduce el interés r:
> 6
Por favor, introduce el numero de anyos n:
> 7
El valor final (F) es:22554.4538848704
*** Ejecución Finalizada. ***

```

Diagrama de Flujo



Ejercicio resuelto

Construya pseudocódigo y diagrama de flujo correspondiente, que al recibir como datos la longitud y el peso de un objeto expresado en pies y libras, imprima los datos de este objeto pero expresados en metros y kilos, respectivamente.

Datos de entrada

- Longitud del objeto en pie.
- Peso del producto en libras.

Datos de salida

- Longitud del objeto en metros.
- Peso del producto en kilos.

Definición de variables

- PIE es una variable de tipo real, que representa la longitud del producto en pies.
- LIB es una variable de tipo real, que representa el peso del producto en libras.
- METRO y KIL son variables de tipo real, que almacenan los datos del objeto en metros y kilogramos, respectivamente.

Pseudocódigo**Proceso medidas**

```
//medidas.psc
//El programa al recibir como datos la longitud y el peso de un objeto
//expresados en pies y libras, calcula los datos de este objeto pero
//en metros y kilogramos, respectivamente.
Definir PIE, LIB, METRO, KIL Como Real;
Escribir "Ingrese la longitud del objeto en pie:";
Leer PIE;
Escribir "Introduzca el peso del objeto en libras:";
Leer LIB;
METRO = PIE * 0.09290;
KIL = LIB * 0.45359;
Escribir "Longitud en metros:",METRO;
Escribir "Peso en kilos: ",KIL;
```

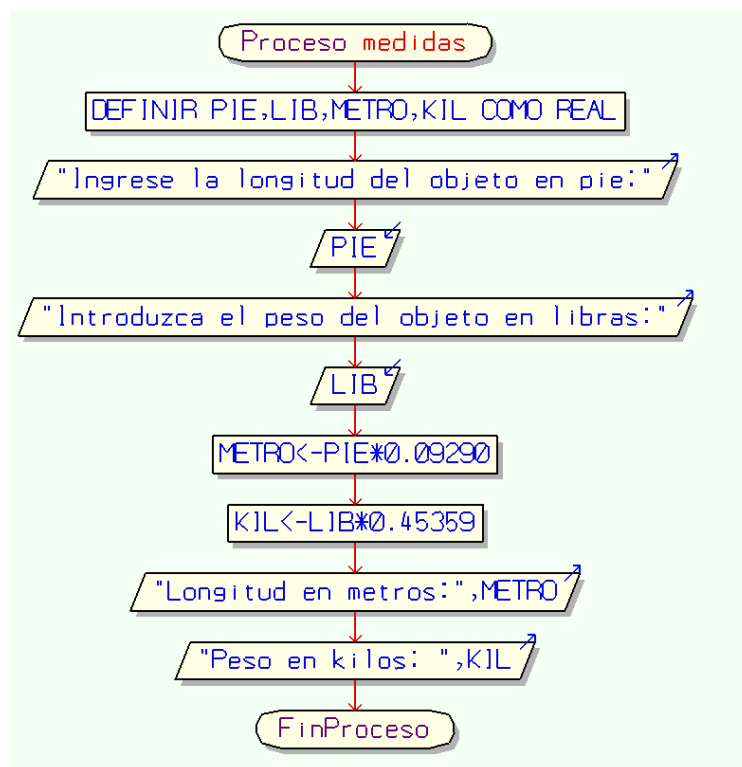
FinProceso**Prueba de escritorio**

PIE	LIB	METRO	KIL
6	300	0.5574	136.077

Salida

```
PSelInt - Ejecutando proceso MEDIDAS
*** Ejecución Iniciada. ***
Ingrese la longitud del objeto en pie:
> 6
Introduzca el peso del objeto en libras:
> 300
Longitud en metros:0.5574
Peso en kilos: 136.077
*** Ejecución Finalizada. ***
```

Diagrama de Flujo



Ejercicio resuelto

A la mamá de Juan le preguntan su edad, y contesta: tengo 3 hijos, pregúntele a Juan su edad. Alberto tiene $\frac{2}{3}$ de la edad de Juan, Ana tiene $\frac{4}{3}$ de la edad de Juan y mi edad es la suma de las tres. Hacer un algoritmo que muestre la edad de los cuatro.

Datos de entrada

- Edad de Juan

Datos de salida

- Edad de Alberto.
- Edad de Juan. (En este caso, es un dato de entrada y salida al mismo tiempo).
- Edad de Ana.
- Edad de la mamá.

Proceso

- Edad de Alberto = $\frac{2}{3}$ * edad de Juan
- Edad de Ana = $\frac{4}{3}$ * edad de Juan
- Edad de la mamá = Edad de Alberto + Edad de Juan + Edad de Ana.

Definición de variables

EDJUAN: **Edad de Juan.**
 EDALBER: **Edad de Alberto.**
 EDANA: **Edad de Ana.**
 EDMAMA: **Edad de la mamá**

Pseudocódigo**Proceso edades**

```

Escribir 'Ingrese la edad de Juan:';
Leer EDJUAN;
EDALBER <- 2 * EDJUAN / 3;
EDANA <- 4 * EDJUAN / 3;
EDMAMA <- EDJUAN + EDALBER + EDANA;
Escribir 'Las edades son:';
Escribir 'ALBERTO:', EDALBER, ' JUAN:', EDJUAN, ' ANA:', EDANA, '
MAMA: ', EDMAMA

```

FinProceso

Prueba de escritorio

Si el valor de la edad de Juan es 9, las otras edades serán:

EDJUAN	EDALBER	EDANA	EDMAMA
9	6	12	27

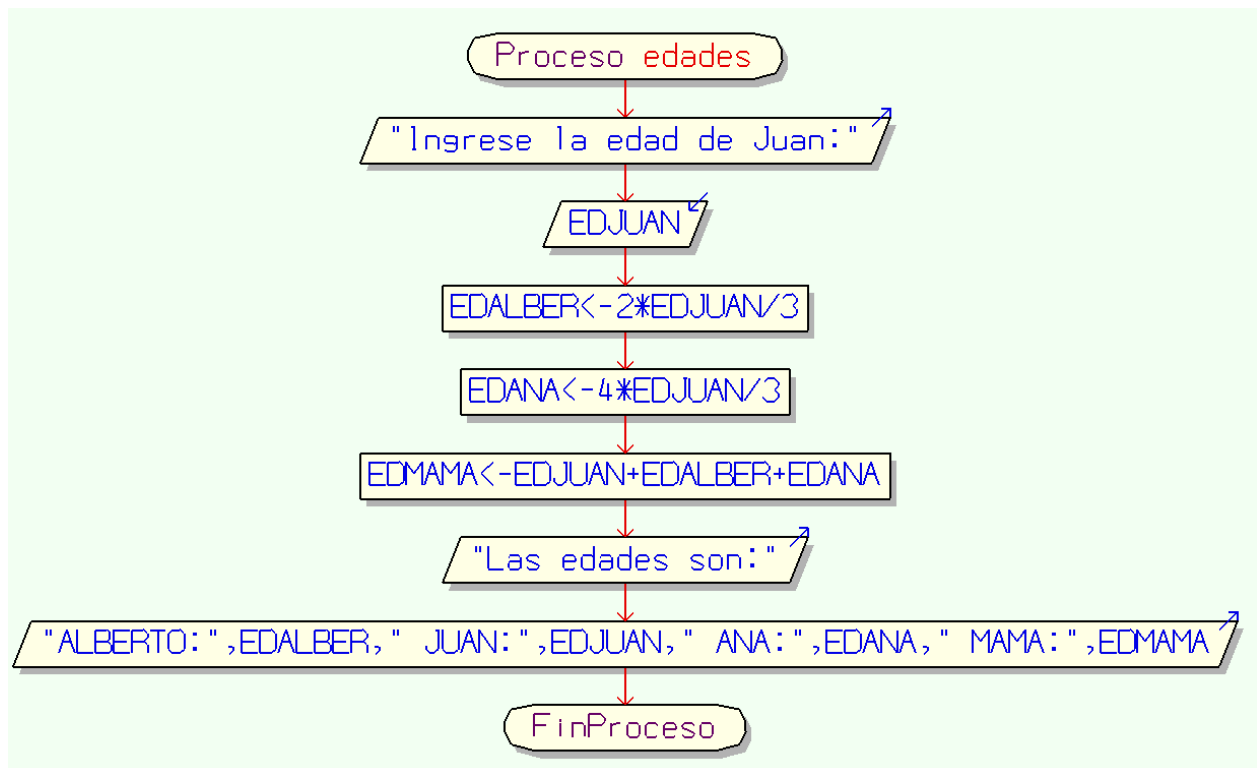
Salida

```

PSeInt - Ejecutando proceso EDADES
*** Ejecución Iniciada. ***
Ingrese la edad de Juan:
> 9
Las edades son:
ALBERTO:6 JUAN:9 ANA:12 MAMA:27
*** Ejecución Finalizada. ***

```

Diagrama de Flujo



Ejercicio resuelto

Hacer un seguimiento (prueba de escritorio) del siguiente grupo de instrucciones.

Proceso operaciones

```
SUMA <- 0;
X <- 20;
SUMA <- SUMA + X;
Z <- 40;
X <- X + Z ^ 2;
SUMA <- SUMA + X / Z;
Escribir 'EL VALOR DE LA SUMA ES: ', SUMA;
```

FinProceso

Prueba de escritorio

SUMA	X	Z
0	20	
20	1620	40
60.5		

La raya horizontal indica que el valor subrayado desaparece y es reemplazado por un nuevo valor.

Salida

```
PSeInt - Ejecutando proceso OPERACIONES
*** Ejecución Iniciada. ***
EL VALOR DE LA SUMA ES: 60.5
*** Ejecución Finalizada. ***
```

Aspectos a tener en cuenta:

- En un algoritmo a las variables se les da valor, ya sea por una asignación o por una instrucción de entrada.
- El valor que se le asigna a una variable debe ser del mismo tipo de dato que la variable.
- En la prueba de escritorio se deben mostrar los cambios que sufren todas las variables del algoritmo.
- Si una variable aparece en más de una instrucción de entrada o asignación, su valor es destruido cada vez que se ejecuta una nueva instrucción.

Más ejercicios resueltos

1. Dada la matrícula y 5 calificaciones de un alumno obtenido a lo largo del semestre; construya un algoritmo que imprima la matrícula del alumno y el promedio de sus calificaciones. Realice una prueba de escritorio con los valores siguientes:

MAT = 16500, 16650, 17225, 17240, 18240.

CAL1 = 8, 9, 9, 8.5, 7.3

CAL2 = 8.5, 8, 10, 9, 6.8

CAL3 = 9, 9, 10, 7.5, 9.5

CAL4 = 7, 7, 8, 6, 8

CAL5 = 6, 9, 9, 6.5, 8.5

Datos de entrada

- La matrícula.
- Las 5 calificaciones

Datos de salida

- La matrícula (En este caso, es un dato de entrada y salida al mismo tiempo).
- Promedio

Proceso

- PROMEDIO = suma de las 5 calificaciones divididas entre 5.

Definición de variables

MAT: Matrícula del estudiante.

CAL1: Valor que representa la calificación 1 del alumno

CAL2: Valor que representa la calificación 2 del alumno

CAL3: Valor que representa la calificación 3 del alumno

CAL4: Valor que representa la calificación 4 del alumno

CAL5: Valor que representa la calificación 5 del alumno

PROMEDIO: Valor que representa el promedio

Pseudocódigo**Proceso promedio**

Escribir 'MATRICULA DEL ALUMNO:';

Leer MAT

Escribir 'INGRESE 5 CALIFICACIONES:';

Leer CAL1, CAL2, CAL3, CAL4, CAL5;

PROMEDIO <- (CAL1 + CAL2 + CAL3 + CAL4 + CAL5)/5

Escribir 'EL ALUMNO CON MATRICULA ', MAT, ' TIENE COMO PROMEDIO ', PROMEDIO

FinProceso

Prueba de escritorio

No. De ejecución	MAT	CAL1	CAL2	CAL3	CAL4	CAL5	PROMEDIO
1	16500	8	8.5	9	7	6	7.7
2	16650	9	8	9	7	9	8.6
3	17225	9	10	10	8	9	9.2
4	17240	8.5	9	7.5	6	6.5	7.5
5	18240	7.3	6.8	9.5	8	8.5	8.02

Salida para la última ejecución

```

PSelnt - Ejecutando proceso PROMEDIO

*** Ejecución Iniciada. ***
MATRICULA DEL ALUMNO:
> 18240
INGRESE 5 CALIFICACIONES:
> 7.3
> 6.8
> 9.5
> 8
> 8.5
EL ALUMNO CON MATRICULA 18240 TIENE COMO PROMEDIO 8.02
*** Ejecución Finalizada. ***

```

Diagrama de Flujo

2. Escriba un algoritmo que resuelva el problema que tienen en una gasolinera. Los surtidores de la misma registran lo que surten en galones, pero el precio de la gasolina está fijado en litros. El algoritmo debe calcular e imprimir lo que hay que cobrarle al cliente.

Consideraciones:

- Cada galón tiene 3.785 litros.
- El precio del litro es \$1.025

Datos de entrada

- Galones surtidos

Datos de salida

- Cobro del cliente.

Procedimiento

- $\text{litros} = \text{galones surtidos} * 3.785$
- $\text{cobro} = \text{cantidad de litros} * 1.025$

Definición de variables

galones: Galones surtidos
 litros: Litros equivalentes a los galones surtidos
 cobro: Cobro al cliente.

Pseudocódigo

Proceso gasolinera

 Escribir 'Galones surtidos:';

 Leer galones

$\text{litros} \leftarrow \text{galones} * 3.785$;

$\text{cobro} \leftarrow \text{litros} * 1.025$;

 Escribir 'Cobro al cliente = ', cobro

FinProceso

Prueba de escritorio

galones	litros	cobro
10.38	39.29	40.27

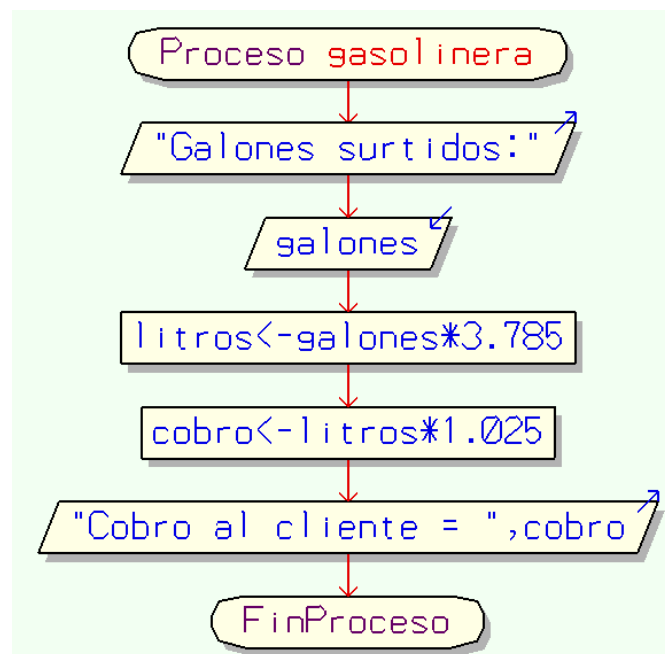
Salida

```

PSeInt - Ejecutando proceso GASOLINERA
*** Ejecución Iniciada. ***
Galones surtidos:
> 10.38
Cobro al cliente = 40.2705075
*** Ejecución Finalizada. ***

```

Diagrama de flujo



3. Construya un algoritmo tal, que dados los tres lados de un triángulo, puede determinar su área. Esta la calculamos aplicando la siguiente fórmula:

$$\text{Area} = \sqrt{S * (S - L1) * (S - L2) * (S - L3)}$$

$$S = (L1 + L2 + L3) / 2$$

Datos de entrada

- Valor de los tres lados del triángulo.

Datos de salida

- El valor del semiperímetro (éste es un dato intermedio).
- El valor del área del triángulo.

Proceso

Los cálculos necesarios para obtener los resultados partiendo de los datos de entrada, son:

- Semiperímetro = suma de los tres lados del triángulo dividido entre 2
- Area = elevar a la potencia 0.5, el resultado de la expresión para calcular el área

Definición de variables

L1: Valor del lado 1 del triángulo.

L2: Valor del lado 2 del triángulo.

L3: Valor del lado 3 del triángulo.

S: Semiperímetro del triángulo.

Area: Área del triángulo.

Pseudocódigo

Proceso areatriangulo

 Escribir 'Introduzca los tres lados del triangulo:';

 Leer L1,L2,L3;

$S \leftarrow (L1+L2+L3) / 2;$

$Area \leftarrow RC(S * (S - L1) * (S - L2) * (S - L3));$

 Escribir 'El area del triangulo es:', Area

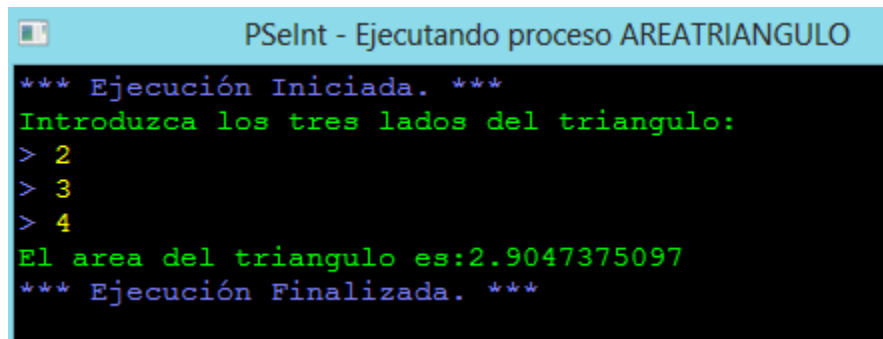
FinProceso

Prueba de escritorio

Si el registro de entrada es:

2 el valor de L1, 3 el valor de L2, 4 el valor de L3

L1	L2	L3	S	Area
2	3	4	4.5	2.90

Salida


```

*** Ejecución Iniciada. ***
Introduzca los tres lados del triangulo:
> 2
> 3
> 4
El area del triangulo es:2.9047375097
*** Ejecución Finalizada. ***

```

Ejercicios propuestos

1. Un empleado trabaja 48 horas en la semana a razón de \$5 la hora. El porcentaje de retención es del 12,5% del salario bruto. Se desea saber cuál es el salario bruto, la retención y el salario neto del trabajador.
2. Elabore un algoritmo que lea un número y obtenga su cuadrado y su cubo.
3. Diseñe un algoritmo que lea un número negativo e imprima el número y el positivo del mismo.
4. Se tiene la siguiente información de un empleado:
 - Código del empleado
 - Nombres
 - Número de horas trabajadas al mes
 - Valor de la hora trabajada
 - Porcentaje de retención

Haga un algoritmo que muestre: Código, nombres, salario bruto y salario neto.

5. Construya un algoritmo tal, que dados los datos enteros A,B, escriba el resultado de la siguiente expresión:

$$\frac{(A + B)^3}{3}$$

Realice una prueba de escritorio con los valores de A = 5, 7, 0, 12, 14 y B = 6, 10, 3, 2, -5

6. Escriba un algoritmo tal, que dados como datos dos números reales, calcule la suma, resta y multiplicación de dichos números. Simule el programa (prueba de escritorio) para los valores de N1 = 7.25, 50.60, 43.20, 9.85, 100.70 y de N2 = 18.30, 3.98, 12.60, 1.90, 87.60
7. Construya un algoritmo tal, que dado el radio de un círculo, calcule e imprima su área y la longitud de su circunferencia.

- El área de un círculo la calculamos como: **$\text{Area} = \pi * \text{radio}^2$**
- **$\text{Circunferencia} = 2 * \pi * \text{radio}$**

Realice una prueba de escritorio (simulación) con los valores de radio = 7.25, 11.14, 37.90, 1.31, 83.20.

8. Un alumno desea saber cuál será su calificación final en la materia de Algoritmos. Dicha calificación se compone de los siguientes porcentajes:

55% del promedio de sus tres calificaciones parciales.
30% de la calificación del examen final.
15% de la calificación de un trabajo final.

9. Un maestro desea saber qué porcentaje de hombres y que porcentaje de mujeres hay en un grupo de estudiantes.
10. Calcular el nuevo salario de un obrero si obtuvo un incremento del 25% sobre su salario anterior.
11. Todos los lunes, miércoles y viernes, una persona corre la misma ruta y cronometra los tiempos obtenidos. Determinar el tiempo promedio que la persona tarda en recorrer la ruta en una semana cualquiera.
12. Tres personas deciden invertir su dinero para fundar una empresa. Cada una de ellas invierte una cantidad distinta. Obtener el porcentaje que cada quien invierte con respecto a la cantidad total invertida.

Bibliografía

- **Oviedo Regino Efraín M.** Lógica de Programación. 2da edición.
- **Hernández María Lourdes.** Diseño estructurado de algoritmos. Diagramas de flujos y pseudocódigos. Universidad de Teuextepe, México.
- **Osvaldo Cairó.** Metodología de la programación: Algoritmos, diagramas de flujo y programas. Editorial COMPUTEC.