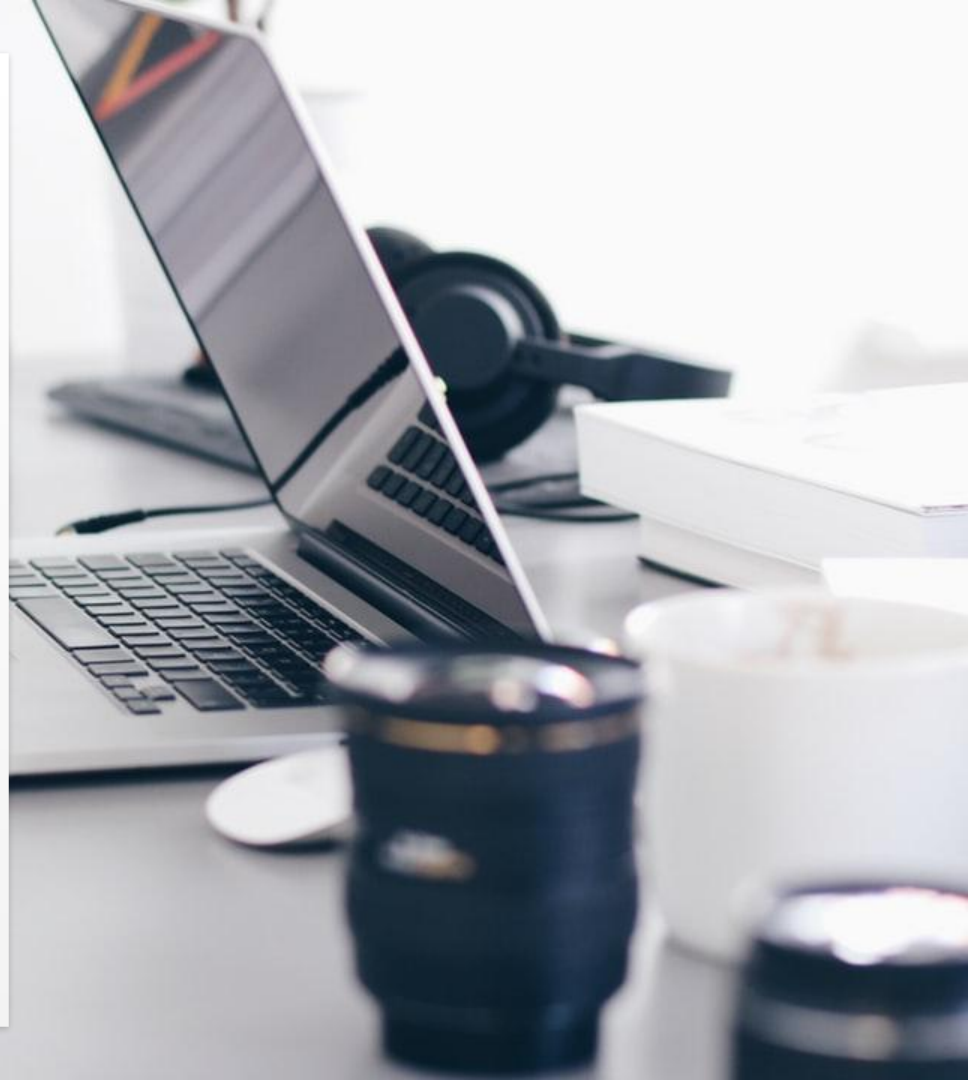


Algoritmos formales

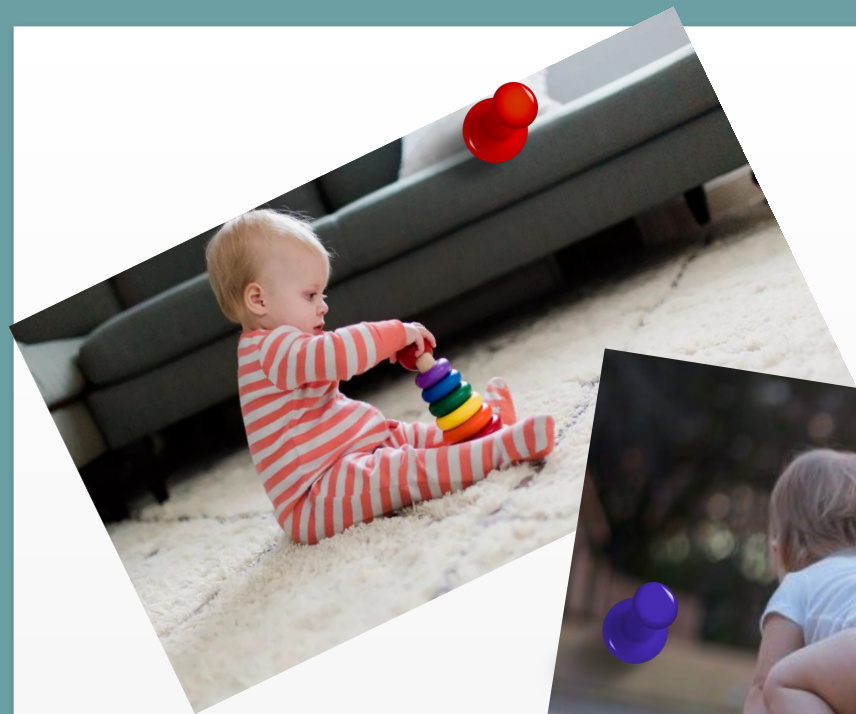


RECORDEMOS...

“Es sencillamente un conjunto de reglas para efectuar algún cálculo, bien sea a mano, o más frecuentemente en una máquina” (Brassard y Bratley 2000).

Cualquier lenguaje formal (textual o gráfico), exige tener algunas reglas mínimas de construcción:

- ❖ *Símbolos (operaciones matemáticas, lógicas)*
- ❖ *Palabras (iconos) que representan acciones*
- ❖ *Reglas para construir instrucciones (sintaxis)*





Las palabras mágicas

De la algoritmia

1

¡HOLA MUNDO!



Fortran

```
PROGRAM HOLA  
  
  PRINT *, '¡Hola,mundo!'  
  
END
```



Python 3.X

```
print('Hola mundo')
```



Javascript

```
//Caso console.log()  
console.log("Hola Mundo!");
```



C

```
#include <stdio.h>  
  
int main(void) {  
    printf("¡Hola,mundo!\n");  
    return 0;  
}
```



Ensamblador

```
.section .rodata  
    .MSG  
    .string "¡Hola Mundo!"  
  
    .section .text  
    .globl main
```



Golang

```
package main  
  
import "fmt"  
  
func main() {  
    fmt.Println("Hola mundo")  
}
```

ACCIONES

- *Leer, Entrada:* Estas se usan para recibir información por pantalla o por archivos.
- *Imprimir, Escribir, Salida:* Estas se usan para sacar información en pantalla o por archivos.
- *Si, Si no:* Estas se usan para preguntar, si pasa esto sucede aquello, sino, sucede aquello otro.
- *Para, Mientras:* Estas se usan para los ciclos dependiendo de la longitud de la repetición.

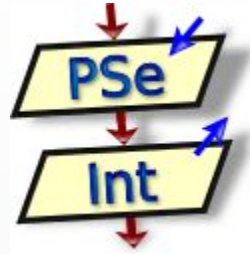
SUJETOS

En cualquier “lenguaje” la información se representa usando identificadores (Ej. masa, g, peso). Los identificadores “abstraen” las características específicas de la información. Los identificadores pueden contener:

- *Un sólo atributo.*
- *Listas o arreglos.*
- *Datos de entrada/salida.*

OPERADORES

Operadores matemáticos	Operadores de Relación	Operadores lógicos
Se usan para formulas matemáticas	Se usan en condicionales para establecer una relación	Se usan en condicionales para hacer elecciones lógicas.
+ Suma	> Mayor que	& Y
- Resta	< Menor que	o
* Multiplicación	= Igual	!/~ No
/ División	>= Mayor igual que	
a^n exp(a) Potencia	<= Menor igual que	
$\sqrt[n]{a}$ Raíz	!= Diferente	
log(a) ln(a) Logaritmo		
π		



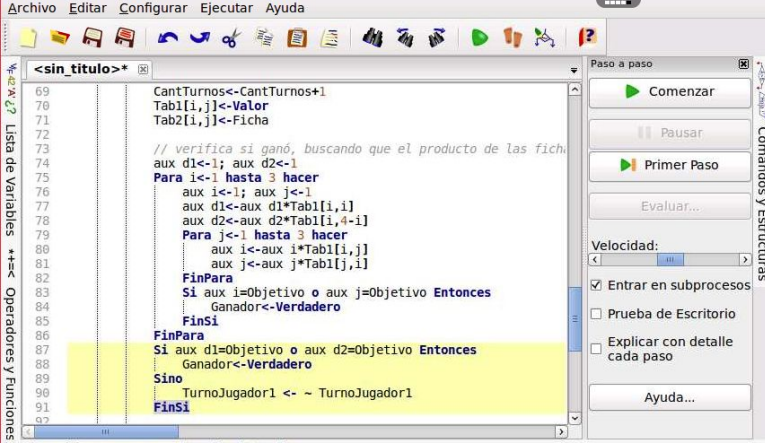
PSeInt

<http://pseint.sourceforge.net/>

<https://sourceforge.net/projects/pseint/>

LISTA DE CHEQUEO

- ➔ *Revisar el perfil 111Mil*
- ➔ *Hacer un ¡Hola Mundo!*
- ➔ *Revisar las operaciones y funciones*
- ➔ *Revisar variables*

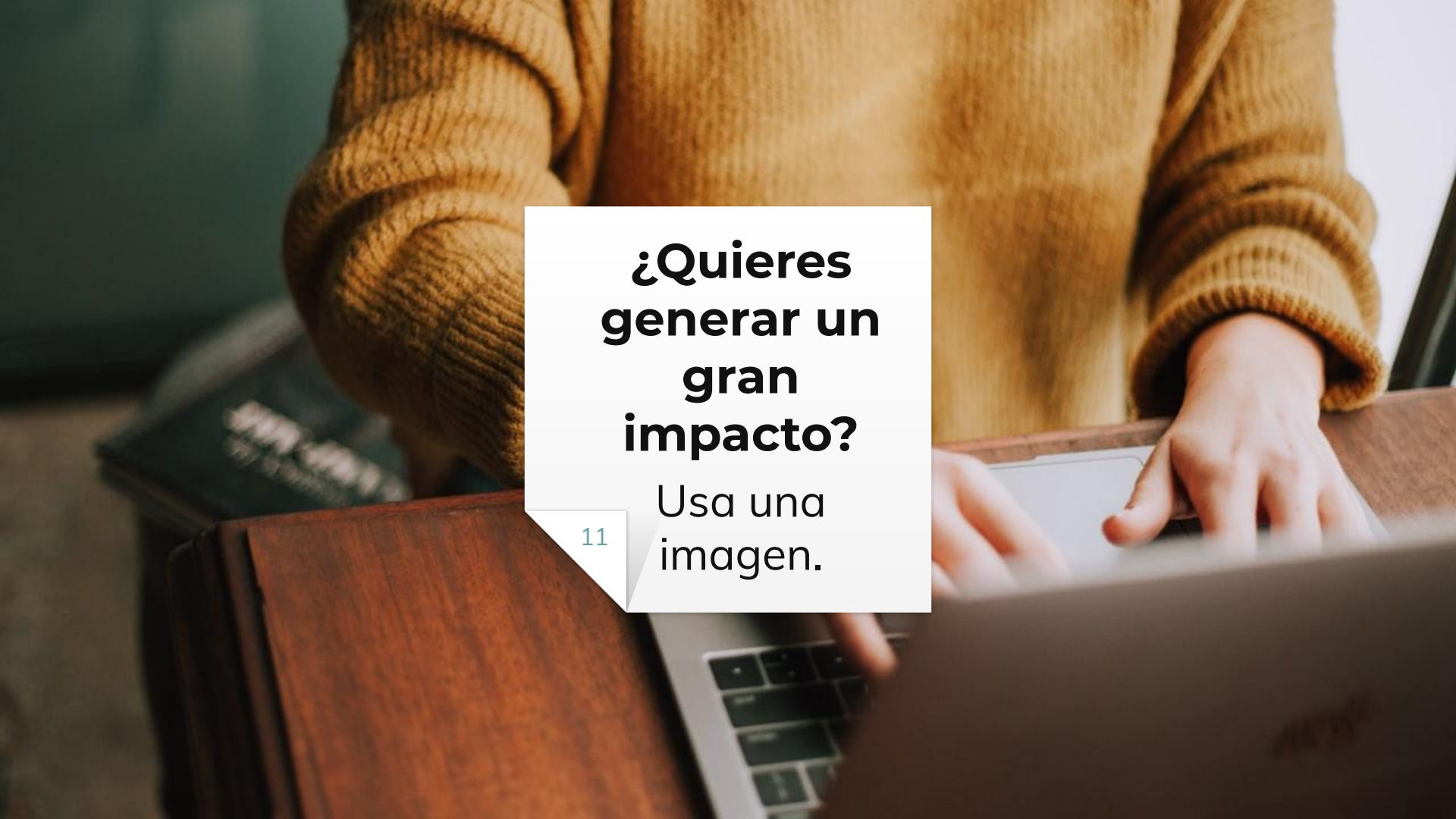


The screenshot shows the NetLogo IDE interface. The main window displays a script titled "<sin_titulo>". The script contains the following code:

```
69 CantTurnos<-CantTurnos+1
70 Tab1[i,j]<-Valor
71 Tab2[i,j]<-Ficha
72
73 // verifica si ganó, buscando que el producto de las fichas
74 aux d1<-1; aux d2<-1
75 Para i<-1 hasta 3 hacer
76   aux i<-1; aux j<-1
77   aux d1<-aux d1*Tab1[i,i]
78   aux d2<-aux d2*Tab1[i,4-i]
79   Para j<-1 hasta 3 hacer
80     aux i<-aux i*Tab1[i,j]
81     aux j<-aux j*Tab1[j,i]
82   FinPara
83   Si aux i=Objetivo o aux j=Objetivo Entonces
84     Ganador<-Verdadero
85   FinSi
86 FinPara
87 Si aux d1=Objetivo o aux d2=Objetivo Entonces
88   Ganador<-Verdadero
89 Sino
90   TurnoJugador1 <- ~ TurnoJugador1
91 FinSi
92
```

On the right side, the 'Paso a paso' (Step-by-step) panel is visible, showing buttons for 'Comenzar' (Start), 'Pausar' (Pause), 'Primer Paso' (First Step), and 'Evaluar...' (Evaluate...). Below these buttons, there is a 'Velocidad:' (Speed) slider and checkboxes for 'Entrar en subprocesos' (Enter sub-processes), 'Prueba de Escritorio' (Desktop Test), and 'Explicar con detalle cada paso' (Explain in detail each step). An 'Ayuda...' (Help) button is at the bottom of the panel.

At the bottom of the IDE window, a status bar message reads: "La ejecución paso a paso ha sido detenida." (The step-by-step execution has been stopped.)

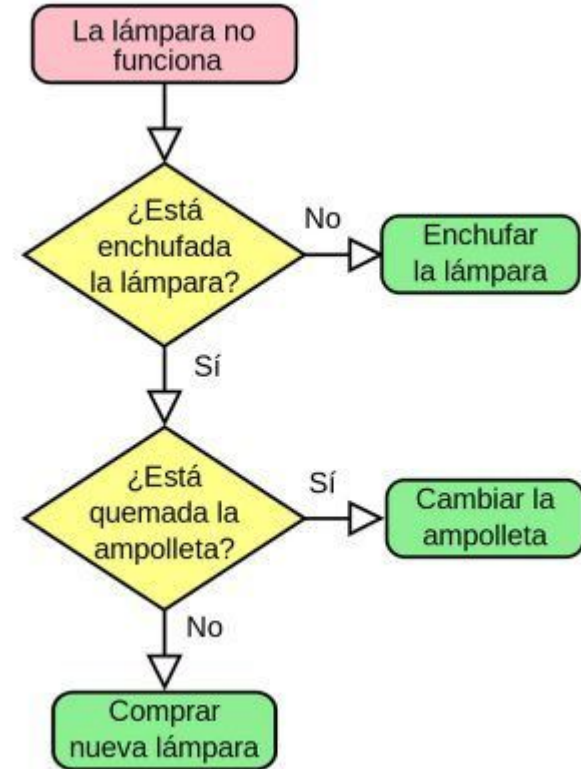
A person wearing a mustard yellow sweater is sitting at a wooden desk, using a laptop. The person's hands are visible on the laptop's trackpad and keyboard. The background is slightly blurred, showing a dark surface and some papers.

**¿Quieres
generar un
gran
impacto?**

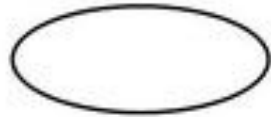
Usa una
imagen.

DIAGRAMAS DE FLUJO

Los diagramas de flujo son descripciones gráficas de algoritmos; usan símbolos conectados con flechas para indicar la secuencia de instrucciones y están regidos por cierta simbología.



DIAGRAMAS DE FLUJO



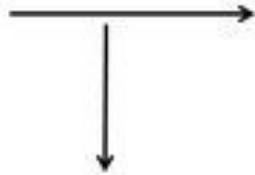
Inicio y fin



Control de datos



Conector dentro
de página



Control de flujo



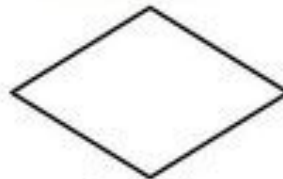
Impresión/
Documento



Conector fuera
de página



Proceso



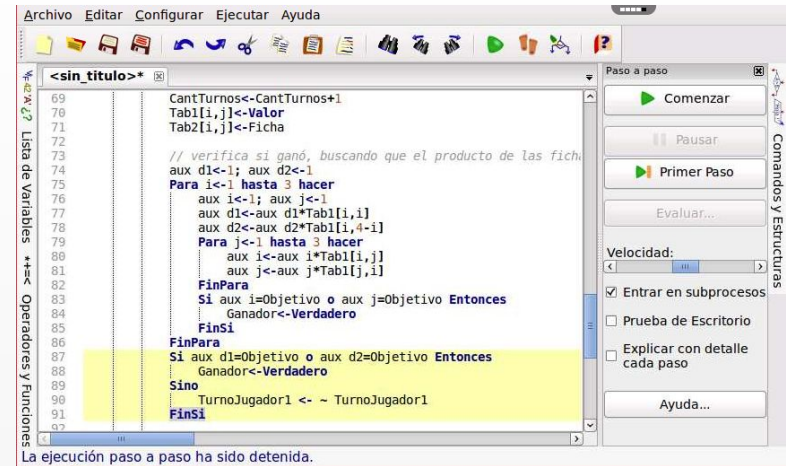
Condición



Impresión en pantalla

LISTA DE CHEQUEO

- *Revisar diagramas de flujo*
- *Hacer un ¡Hola Mundo!*
- *Utilizar todas las herramientas*
- *Exportar un diagrama*
- *Revisar la relación entre el tipo pseudo-código y el diagrama*
- *Revisar estructuras*



PRUEBA DE ESCRITORIO

Una vez se ha descrito un algoritmo... ¿cómo se prueba? ¿cómo sabemos que está bien escrito? Pues, haga una prueba de escritorio:

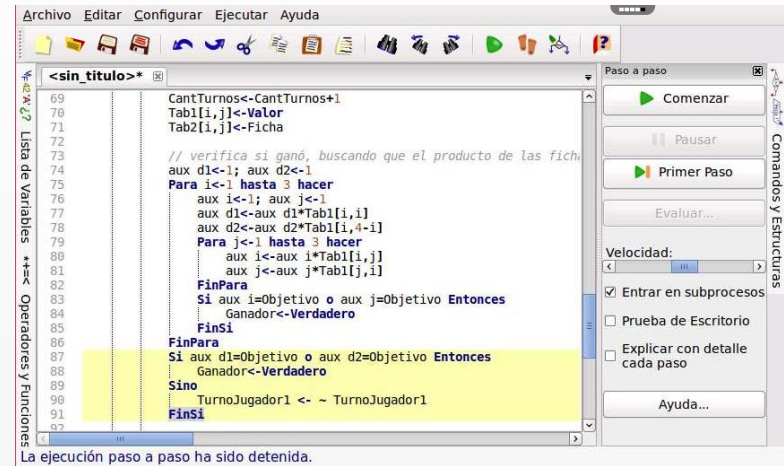
- ★ *Seleccione previamente los valores de entrada. Para ello recorra todos los caminos posibles y determine los valores de los parámetros de entrada que lo llevarían por esos caminos.*
- ★ *Ejecute la prueba de escritorio para cada conjunto de valores de entrada.*

PRUEBA DE ESCRITORIO

- ★ *Ejecute el algoritmo paso a paso, colocando el número de la línea y el resultado de la acción respectiva. El resultado de las acciones que modifiquen los valores de los identificadores se indica explícitamente.*
- ★ *Cuando se evalúe una condición coloque la condición con valores explícitos e indique si la condición se cumple. Cuando se muestre algo en pantalla indique explícitamente.*

LISTA DE CHEQUEO

- ➔ *Revisar paso a paso*
- ➔ *Revisar prueba de escritorio*
- ➔ *Revisar evaluar una variable*
- ➔ *Revisar explicaciones*
- ➔ *Realizar un prueba para la solución de una ecuación de segundo orden.*



Ejercicios usando PSeInt

1. *Escribir el pseudo-código para obtener el área de un triángulo.*
2. *Hacer el diagrama de flujo para determinar el valor en dólares de X cantidad en pesos colombianos .*
3. *Escribir el pseudo-código para obtener la edad de una persona desde el año en que nacieron.*
4. *Hacer el diagrama de flujo para determinar la hipotenusa de un triángulo*
5. *Escribir el pseudo-código para determinar la cantidad total de dinero que ahorra una persona en un año, si guarda un porcentaje X de su sueldo cada mes.*

Ejercicios usando PSeInt

1. Hacer el diagrama de flujo para obtener el día de la semana cuando se ingresa un número [1-7].
2. Siendo la entrada la fecha de nacimiento de una persona, hacer un pseudo-código que indique si esta apta para votar para una fecha X de elecciones (siendo que el trámite del documento oficial podría durar 1 semana y sin este no puede votar)
3. Se tiene el nombre y la edad de un grupo de personas no mayor a 5. Hacer el diagrama de flujo para determinar el nombre de la persona de menor edad.
4. Escribir el pseudo-código para asignar un número (desde 1) al ingresar una letra del alfabeto (en minúscula) y 0 si es otro carácter.
5. Hacer el diagrama de flujo para determinar el cobro de un estacionamiento que cobra las primeras 2 horas a 10000 c/u, las siguientes 3 a 8000 c/u, las siguientes 5 a 6000 c/u y después de 10 horas el costo es de 4000 c/u.

Ejercicios usando PSeInt

1. Escribir el pseudo-código para determinar de N cantidades cuántas mayores a cero y cuántas menores a cero.
2. Hacer el diagrama de flujo del factorial de un número natural.
3. Escribir un pseudo-código para calcular el promedio de N notas (no mayor a 6) cada una de ellas con un porcentaje asignado.
4. Hacer el diagrama de flujo para determinar el rendimiento a N años de una inversión que comienza en (X) cantidad y una ganancia del 5% mensual, cuando el saldo acumulado alcanza una cantidad (Y) la ganancia sube al 10%.
5. Escribir el pseudo-código para determinar si un número natural es primo o no.

!Gracias!

¿Preguntas?

Créditos

Special thanks to all the people who made and released these awesome resources for free:

- Presentation template by SlidesCarnival
- Photographs by Unsplash