

EJEMPLO DE DESARROLLO DE UN PROGRAMA

El jefe de contabilidad de la empresa “Perlas de Manacor” quiere un programa para calcular los salarios de los empleados de la compañía. Este programa debería aceptar el número de la seguridad social de un empleado, su nombre y apellidos, el sueldo base por hora y el número de horas trabajadas. Para cada empleado el programa debe mostrar en pantalla: su nombre y apellidos, su número de la seguridad social, el número de horas trabajadas, el sueldo base y el salario total.

1. Análisis y especificación del problema.

El primer paso es **analizar el problema** y formular una **especificación** precisa del mismo. Esta especificación debe incluir una **descripción de la entrada del problema** (qué información se da y qué elementos son importantes para resolver el problema) y una **descripción de la salida del problema** (qué información debe producirse para resolver el problema). La entrada y la salida son las dos partes más importantes de la especificación de un programa.

Identificar la entrada y salida de este problema es fácil:

Entrada	Salida
Nombre y apellidos.	Nombre y apellidos.
Número de la seguridad social.	Número de la seguridad social.
Sueldo base por hora.	Sueldo base por hora.
Número de horas trabajadas.	Número de horas trabajadas.
	Salario total.

2. Diseño.

Una vez dada la especificación de un problema se debe formular **un plan de diseño** para desarrollar un programa que cumpla las especificaciones. Dos aspectos importantes en el diseño son la selección apropiada de **estructuras para organizar y almacenar los datos** que van a ser procesados y el diseño de **procedimientos para procesar dichos datos**.

Debido a que la computadora es una máquina que carece de la capacidad inherente de resolver problemas, los procedimientos desarrollados para resolver un problema deben estar formulados en forma de **secuencia detallada de pasos sencillos**. Tales procedimientos reciben el nombre de **algoritmos**.

Los pasos de que consta un algoritmo deben estar organizados de manera lógica y clara de forma que el programa que implementa este algoritmo esté igualmente bien estructurado. Los algoritmos y los programas estructurados se diseñan usando tres métodos básicos de control:

1. Secuencial. Los pasos se ejecutan uno tras otro; cada uno de ellos se ejecuta exactamente una vez.

2. Selectiva. Se selecciona y ejecuta una entre varias acciones posibles.

3. Repetitiva. Se ejecutan repetidamente uno o más pasos.

Estas tres estructuras son individualmente bastante sencillas, pero en realidad son suficientemente potentes como para poder construir cualquier algoritmo con ellas.

Los algoritmos se describen en **pseudocódigo** y la estructura de un algoritmo se puede visualizar mediante un **diagrama estructural** que muestra las distintas tareas que pueden ejecutar y la relación entre ellas.

Para este problema se procesarán y almacenarán los datos usando **variables** parecidas a las utilizadas en matemáticas para nombrar cantidades en fórmulas algebraicas y ecuaciones (**Tipos básicos de datos: numéricos, lógicos y texto**).

Algoritmo en pseudocódigo:

INICIO Calcular Nomina.

VAR nombre, numeroSocial, sueldoBase, horasTrabajadas, salarioTotal;

ESCRIBIR "Nombre y Apellidos";

LEER nombre;

ESCRIBIR "Número Seguridad Social";

LEER numeroSocial;

ESCRIBIR "¿Cuánto cobras por hora trabajada?";

LEER sueldoBase;

ESCRIBIR "¿Cuántas horas has trabajado este mes?";

LEER horasTrabajadas;

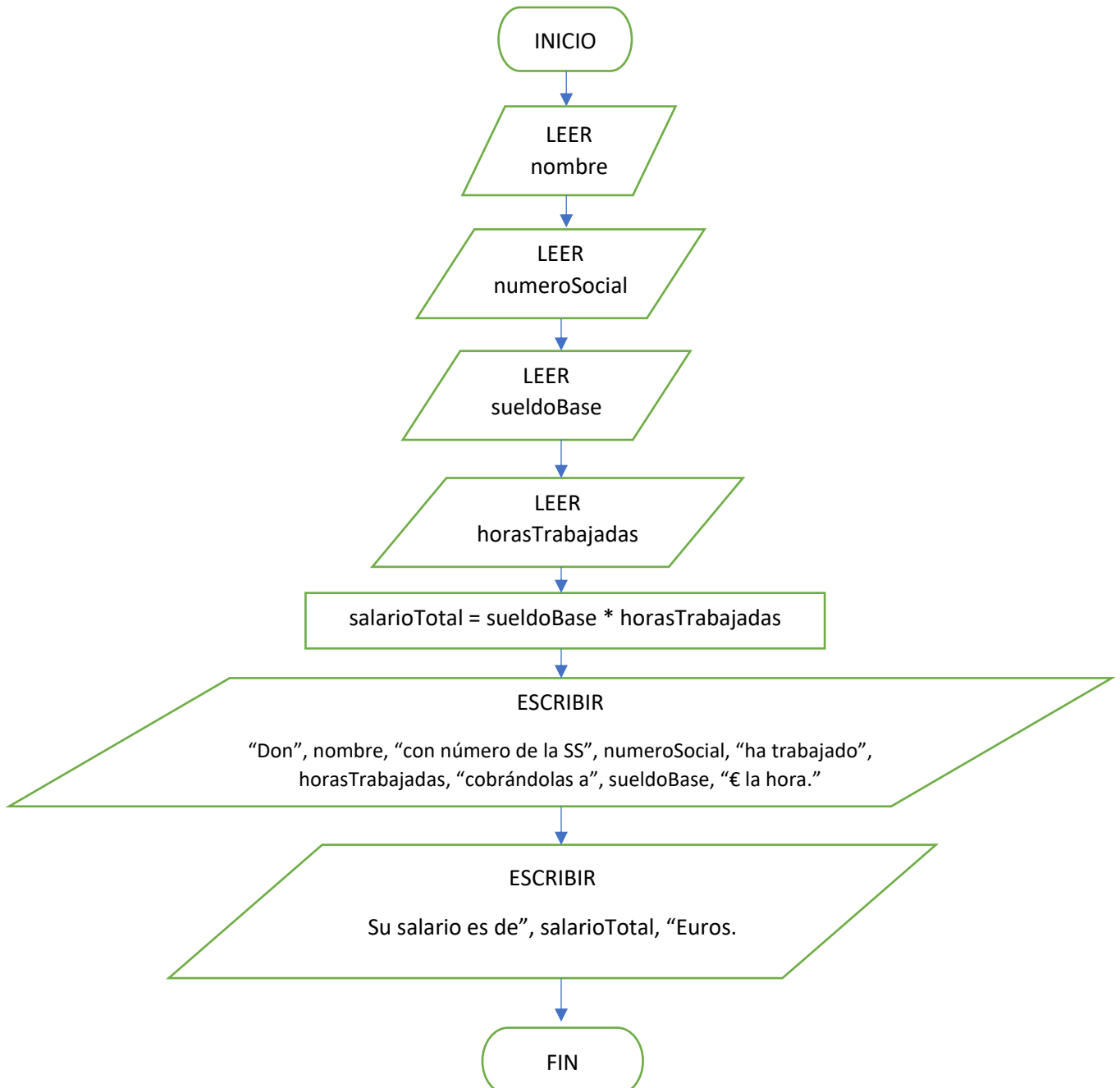
salarioTotal <- sueldoBase * horasTrabajadas;

ESCRIBIR "Don", nombre, "con número de la SS", numeroSocial, "ha trabajado", horasTrabajadas, "cobrándolas a", sueldoBase, "€ la hora.";

ESCRIBIR "Su salario es de", salarioTotal, "Euros.";

FIN

Algoritmo en diagrama de flujo:



3. Codificación.

Los dos primeros pasos del proceso de desarrollo de programas son extremadamente importantes, puesto que las siguientes fases serán mucho más complicadas si se ha pasado por alto los dos primeros pasos o se han realizado sin el suficiente cuidado. Por otro lado, si se ha analizado y especificado cuidadosamente el problema y se ha desarrollado un plan de diseño efectivo, el tercer paso, la codificación del programa, es normalmente directo.

La **codificación** es el proceso de implementar en algún lenguaje de programación las variables (y estructuras) usadas para almacenar los datos y los algoritmos para resolver el problema.

calcularNomina.py

#Declaramos las variables.

nombre = ""

numeroSocial = ""

sueldoBase = 0

horasTrabajadas = 0

salarioTotal = 0

#Pedimos los datos.

print('Nombre y Apellidos')

nombre = input()

print('Número Seguridad Social')

numeroSocial = input()

print('¿Cuánto cobras por hora trabajada?')

sueldoBase = float(input())

print('¿Cuántas horas has trabajado este mes?')

horasTrabajadas = float(input())

#Calculamos el salario

salarioTotal = sueldoBase * horasTrabajadas

#Presentamos los resultados

print('Don', nombre, 'con número de la SS', numeroSocial, 'ha trabajado', horasTrabajadas, 'cobrándolas a', sueldoBase, '€ la hora.')

print('Su salario es de', salarioTotal, 'Euros.')