

Asignatura

Programación

## UNIDAD 1

Introducción a la programación



UNIVERSAE  
Instituto Superior de FP

# El origen



## Evolución

- Inicio. Lenguaje máquina (Binario)
- Tareas manuales (Con tarjetas perforadas)
- Primeras calculadoras
- Ada Byron creo el primer algoritmo informático.
- Evolución con dispositivos electrónicos (transistores)
- Sustituir 0 y 1 por palabras. ENSAMBLADOR
- A partir del lenguaje ensamblador empiezan a surgir lenguajes de alto nivel.

## ¿Qué es la programación?

- Proceso de comunicación con el equipo informático para diseñar y codificar instrucciones.
- Objetivo dar una solución a un problema
- El conjunto de instrucciones que da una solución a un problema es un **ALGORITMO**





# Paradigma de programación

## Paradigma

- Forma de clasificar los lenguajes de programación.
- Cada paradigma tiene su reglas y estándares.



## Programación estructurada

- Estructuras básicas:
  - Secuencial
  - Alternativa
  - Iterativa
- No hay complejos saltos de líneas

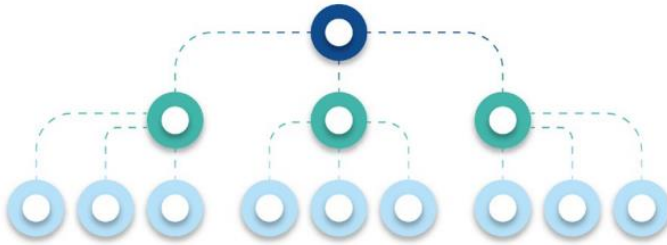


## Programación modular

- Divide el programa en problemas mas pequeños.
- Módulos y subprogramas.
- Un único punto de entrada y salida.

```
Program Saludos;  
  
Var  
    nombre:string;  
  
Begin  
  
    Write(' ¿Cómo te llamas? ');  
    ReadLn(nombre);  
    WriteLn(' Bienvenido ', nombre);  
    ReadLn  
End.
```

# Cohesión y acoplamiento en la programación modular



## Cohesión



## Acoplamiento

**Acoplamiento débil**  
(favorece el mantenimiento)



**Acoplamiento fuerte**  
(dificulta el mantenimiento)

### · Acoplamiento normal

- Acoplamiento normal por datos
- Acoplamiento normal por marca o estampado
- Acoplamiento normal de control

### · Acoplamiento externo

- Acoplamiento global
- Acoplamiento patológico o por contenido

# Pseudocódigo

```
1. Inicio
2. Inicializar variables OP1, OP2, RESULTADO, OPERACION
3. Escribir "Introduce los datos"
4. Leer OP1, OP2, OPERACION
5. Si OPERACION = "Suma" ENTONCES
6.     RESULTADO <- OP1 + OP2
7. Fin si
8. Si OPERACION = "Resta" ENTONCES
9.     RESULTADO <- OP1 - OP2
10. Fin si
11. Escribir RESULTADO
12. Fin
```

- Lenguaje intermedio o cercano al lenguaje natural
- No es un lenguaje computacional
- Sirve para desarrollar algoritmos
- Para aprender a programar

## Estructuras que se pueden usar

- Secuencial
- Alternativa
- Iterativas
- Procedimientos
- Funciones

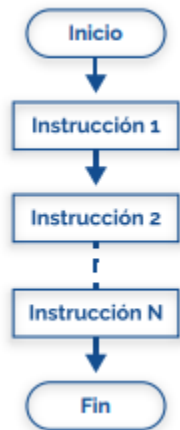


# Diagrama de flujo

- Ordinograma o flujograma
- Representación gráfica de un algoritmo
- Transcripción idéntica del pseudocódigo



## Secuencial



## Alternativa (Si)



## Iterativa (while)





# Resumen

1. El origen
2. Paradigma de programación
  - 2.1 Programación estructurada
  - 2.2 Programación modular
3. Pseudocódigo
4. Diagramas de flujo

The background is a solid blue color. Overlaid on this are several faint, light-blue geometric patterns. These include a grid of small squares that form larger, irregular shapes, and numerous small, light-blue arrows pointing in various directions, some of which are slightly blurred, giving a sense of motion or data flow.

# UNIVERSAE

— CHANGE YOUR WAY —