

# Lenguajes de Programación



Prof. José R. Rojas.

Noviembre, 2.007



# Contenido

## **LENGUAJES DE PROGRAMACION**

- A. Definición
- B. Propósitos de los Lenguajes
- C. Características de los Lenguajes
- D. Niveles de los Lenguajes de Programación
- E. Desventajas de los Lenguajes de Alto Nivel.
- F. Clasificación según su Aplicación



# Lenguajes de Programación

## Definición

Un LENGUAJE DE PROGRAMACION consiste en una serie específica de caracteres que permiten dar instrucciones a un computador, con la finalidad de lograr que ella realice ciertas acciones en la solución de algún problema computarizado. Cada lenguaje tiene características propias en cuanto a su GRAMATICA Y su SINTAXIS





# Lenguajes de Programación

## Propósito de los Lenguajes

### 1. *Naturales*

Los lenguajes naturales o bien conocidos como lenguajes humanos no son más que un medio para lograr la comunicación de pensamientos



# Lenguajes de Programación

## Propósito de los Lenguajes

### *2. Lenguajes de Programación*

Su propósito es más restringido, son limitados por la capacidad del hardware y del software. Se orientan más hacia el mandato de acciones a ejecutar por el computador





# Lenguajes de Programación

## Características de los Lenguajes de Prog

### 1. *Naturales*

- Permite la transmisión de muchos pensamientos a diferentes niveles y con diferentes significados. Una descripción de un pensamiento puede ser impersonal y objetiva o emocional y subjetiva
- Puede utilizarse la prosa y la poesía o la descripción formal, técnica, humorística, etc.



# Lenguajes de Programación

## Características de los Lenguajes de Prog

### 2. *Lenguajes de Programación*

- No se requiere que comuniquen una gran variedad de pensamientos, es suficiente que permitan la transmisión de solo cuatro pensamientos básicos (cuatro tipo de operaciones)
- Es imprescindible que la comunicación sea precisa y clara, es decir, no debe admitir ambigüedades





# Lenguajes de Programación

## Clasificación

Es posible clasificar los lenguajes de programación siguiendo diferentes criterios.

- Se considerarán tres:
  - Nivel de Abstracción
  - Finalidad del lenguaje
  - Características del lenguaje







## Clasificación

Según el Nivel de abstracción:

- Lenguajes de bajo nivel:
  - Cercanos a la máquina
  - Bajo nivel de abstracción
    - Lenguaje Máquina
    - Ensamblador
- Lenguajes de alto nivel:
  - . Cercanos al problema
  - . Alto nivel de abstracción





## Clasificación

Según la finalidad, existen lenguajes:

- Científicos: ALGOL, FORTRAN, ...
- Ingenieriles: ADA, DYNAMO, ...
- de Gestión: COBOL, dBASE, ...
- de Inteligencia Artificial: LISP, PROLOG, ...
- Multipropósito: PASCAL, MODULA-2, C, ...





# Lenguajes de Programación

## Niveles de los Lenguajes de Programación

Los lenguajes de programación se pueden clasificar según su cercanía o similitud al lenguaje máquina o lenguaje binario, a esto se le conoce como nivel del lenguaje. Mientras más parecido sea el lenguaje al lenguaje máquina o binario su nivel será menor y contrapuesto se tendrá los lenguajes de alto nivel que se asemejan a los lenguajes naturales



# Lenguajes de Programación

## Niveles de los Lenguajes de Programación

Lenguaje Natural



Alto Nivel



Lenguaje Máquina

Bajo Nivel





# Lenguajes de Programación

## Niveles de los Lenguajes de Programación

### *1. Bajo Nivel*

Este nivel corresponde al lenguaje de programación ASSEMBLER o lenguaje Ensamblador que está asociado a un respectivo lenguaje máquina. El lenguaje ensamblador es simplemente una representación simbólica del lenguaje máquina asociado, lo cual permite una programación menos tediosa que con el lenguaje máquina correspondiente

# Ejemplo (assembly code)

```
    lda $30,-32($30)
    stq $26,0($30)
    stq $15,8($30)
    bis $30,$30,$15
    bis $16,$16,$1
    stl $1,16($15)
    lds $f1,16($15)
    sts $f1,24($15)
    ldl $5,24($15)
    bis $5,$5,$2
    s4addq $2,0,$3
    ldl $4,16($15)
    mull $4,$3,$2
    ldl $3,16($15)
    addq $3,1,$4
    mull $2,$4,$2
    ldl $3,16($15)
    addq $3,1,$4
    mull $2,$4,$2
    stl $2,20($15)
    ldl $0,20($15)
    br $31,$33
$33:
    bis $15,$15,$30
    ldq $26,0($30)
    ldq $15,8($30)
    addq $30,32,$30
    ret $31,($26),1
```



# Lenguajes de Programación

## Niveles de los Lenguajes de Programación

### *1. Bajo Nivel*

Este lenguaje requiere que el programador conozca profundamente la arquitectura del equipo, además que presenta una complicación de programación mayor que con los lenguajes de alto nivel

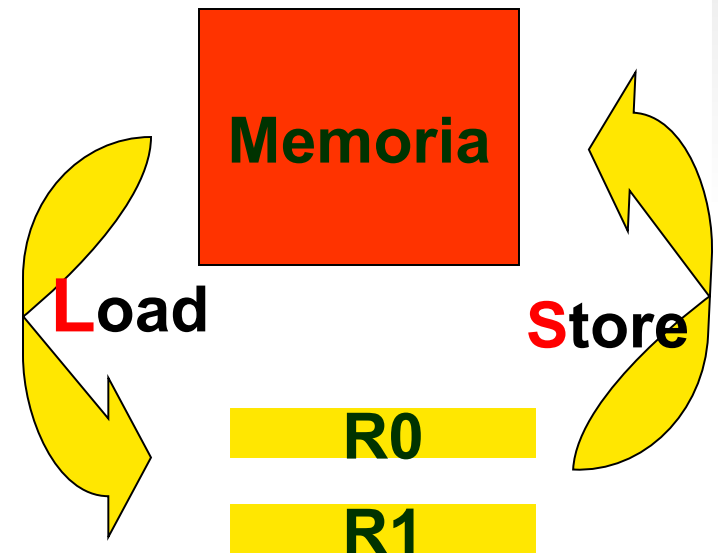


# Lenguajes de Bajo Nivel

## Arquitectura de la máquina

L      R1, operando1  
OP    *R1 o R0 , operando2*  
S      R1, resultado

<OP> ----> A | R | M | D







# Lenguajes de Programación

## Niveles de los Lenguajes de Programación

### 2. Alto Nivel

Son los lenguajes de programación más modernos, presentan una gran facilidad para el programador en cuanto a la facilidad de realizar programas debido a que poseen **MACROINSTRUCCIONES**.

Una **MACROINSTRUCCION** es una instrucción escrita en un lenguaje de diferente al máquina que mediante traducción es convertida en otro grupo de instrucciones, en lenguaje máquina



# Lenguajes de Programación

## Niveles de los Lenguajes de Programación

### 2. Alto Nivel

Los lenguajes de alto nivel pueden ser divididos en distintos grupo, siendo ellos los siguiente:

- Lenguajes de Alto Nivel de 4ta. Generación o Lenguajes Declarativos.
- Lenguajes de Alto Nivel de 3era. Generación.
- Lenguajes de Alto Nivel Especiales o de Mediano Nivel.



# Lenguajes de Programación

## Niveles de los Lenguajes de Programación

Lenguaje Natural



Alto Nivel

Lenguaje Máquina



Bajo Nivel



# Lenguajes de Programación

## Niveles de los Lenguajes de Programación

Lenguaje Natural

Declarativos

3era. Generación

Mediano Nivel

M

Lenguaje Máquina

Bajo Nivel





# Lenguajes de Programación

## Niveles de los Lenguajes de Programación

### 2.1. Lenguajes Declarativos

- Gran parecido con los lenguajes formales en cuanto a su potencia expresiva y funcional.
- Son lenguajes de ordenes, indican "lo que hay que hacer" y no "como hacerlo".



# Lenguajes de Programación

## Niveles de los Lenguajes de Programación

### 2.1. Lenguajes Declarativos

- Son fáciles de aprender y de programar, no requiere de alto conocimiento de computación.
- Requiere traductor.





# Lenguajes de Programación

## Niveles de los Lenguajes de Programación

### 2.2. Lenguajes 3<sup>era</sup>. Generación

- Fácil legibilidad y comprensión entre programadores.
- Tienen una alta transportabilidad, son independientes de la máquina.
- Programación rápida, gracias a las macroinstrucciones.



# Lenguajes de Programación

## Niveles de los Lenguajes de Programación

### 2.2. Lenguajes 3<sup>era</sup>. Generación

- Requiere traductor.
- No requiere que el programador conozca de la arquitectura de la máquina
- Como ejemplo de estos lenguajes tenemos: PASCAL, ADA, MODULA, COBOL, BASIC, ALGOL, PROLOG, LISP, ETC., en la actualidad existen más de 200 lenguajes, muchos de ellos con un elevado número de variantes o dialectos.







# Lenguajes de Programación

## Niveles de los Lenguajes de Programación

### 2.3. Lenguajes de Mediano Nivel

Estos lenguajes presentan características iguales a los lenguajes de alto nivel de 3<sup>era</sup> generación, pero incluyen una característica especial que los ha ayudado a tener una gran importancia en la actualidad como es la de permitir el acceso a la arquitectura de la máquina con un potencial similar al de los lenguajes de bajo nivel, pero con la facilidad de los de alto nivel.





# Lenguajes de Programación

## Niveles de los Lenguajes de Programación

### 2.3. Lenguajes de Mediano Nivel

Una de las aplicaciones principales que es la de construcción de sistemas operativos (UNIX), manejadores de bases de datos Y otros lenguajes de programación . Un ejemplo de este tipo de lenguajes es el *LENGUAJE C en todas sus variantes*.





# Lenguajes de Programación

## Niveles de los Lenguajes de Programación

### Desventajas de los Lenguajes De Alto Nivel

- Tienen un alto tiempo de traducción (compilación).
- No se aprovechan las posibles ventajas de la arquitectura interna de la máquina (caso especial los de mediano nivel).



# Lenguajes de Programación

## Niveles de los Lenguajes de Programación

### Desventajas de los Lenguajes De Alto Nivel

- Se incrementa la ocupación de memoria interna.
- El tiempo de ejecución es mayor puesto que las instrucciones generadas por el compilador son más numerosas que las correspondientes al mismo programa escrito directamente en ASSEMBLER o lenguaje máquina.





# Lenguajes de Programación

## Clasificación según su Aplicación

Según sea la utilización u orientación que posean la mayoría de las aplicaciones elaboradas, los lenguajes de programación se pueden clasificar de la siguiente manera:

- Científico (FORTRAN, C)
- Procesamiento de Datos (PL/1, Cobol, Leng. Xbase)
- Procesamiento de Texto (snobol, XML)
- Inteligencia Artificial (LISP, PROLOG)
- Programación de Sistemas (C, ADA, MODULA)



# Lenguajes de Programación

## Clasificación según su Aplicación

### a. Científico

Son lenguajes que permiten la solución de problemas donde existe una alta manipulación de números y arreglos de números usando principios matemáticos y estadísticos. Usualmente la cantidad de datos en tales problemas es relativamente pequeña y su estructura es relativamente simple. Frecuentemente los problemas científicos requieren de alto procesamiento y bajo proceso de E/S



# Lenguajes de Programación

## Clasificación según su Aplicación

### b. Procesamiento de Datos

Son lenguajes con el propósito de permitirle al programador elaborar programas donde el fin principal es el de crear, mantener y extraer datos contenidos en registros y archivos. El volumen de datos que se manipula es generalmente grande, pero por lo contrario la realización de procesos aritméticos es baja, incrementándose el número de procesos de E/S. Por lo general los procesos son de tipo BATCH en vez de iterativo



# Lenguajes de Programación

## Clasificación según su Aplicación

### c. Procesamiento de Texto

Su principal característica es la de procesar texto en lenguaje natural. El texto procesado no tiene ninguna restricción en cuanto al lenguaje de programación utilizado para el desarrollo de aplicaciones de textos.





# Lenguajes de Programación

## Clasificación según su Aplicación

### d. Inteligencia Artificial

La condición principal de estos lenguajes es la de permitir la implementación de aplicaciones que simulan un comportamiento inteligente. Estos lenguajes son muy utilizados en aplicaciones como juegos de ajedrez, programas de comprensión del lenguaje natural, robótica y sistemas expertos.



# Lenguajes de Programación

## Clasificación según su Aplicación

### e. Programación de Sistemas

Son lenguajes de programación que permiten el desarrollo de programas que hacen interfaz entre la computadora (hardware), el programador y/o el operador. Dentro de las aplicaciones comúnmente desarrolladas en estos lenguajes tenemos: Compiladores, Ensambladores, Traductores, Rutinas de E/S, Facilidades de Gestión, Sistemas Operativos, etc.



# Lenguajes de Programación

## Clasificación según su Aplicación

### e. Programación de Sistemas

Dos de las características que poseen las aplicaciones que se pueden desarrollar con estos lenguajes son: la necesidad de tratar con sucesos impredecibles o excepcionales y la necesidad de coordinar actividades de varios programas o tareas ejecutándose asincrónicamente.





# Lenguajes de Programación



- **ADA**
- **Basic, Visual**
- **Beta**
- **C, C++, Visual C**
- **Java**
- **Cecil** : Cecil is a new purely object-oriented language intended to support rapid construction of high-quality, extensible software
- **COBOL** : COmercial Buisness Oriented Language
- **Elisp**: Emacs lisp
- **Dylan**: Dylan is a new Object Oriented Dynamic Language (OODL). Dylan combines the features of static and dynamic languages
- **Eclipse**: combines Sepia's extended Prolog technology with MegaLog's persistent knowledge base functionality, a substantial subset of CHIP's constraints handling facilities, several new constraints libraries, and soon or-parallelism as featured in ElipSys.



# Lenguajes de Programación



- **Eiffel** is an advanced object-oriented programming language that emphasizes the design and construction of high-quality and reusable software.
- **Elf** is a constraint logic programming language based on the LF Logical Framework. It is intended as a uniform meta-language for specifying, implementing, and proving properties of programming languages and logics

**Erlang**: Concurrent functional programming language for large industrial real-time systems.

Dynamically typed.

**Forth** :Forth is an embeded stack language.

**FORTRAN**

**Lisp**

**Occam** programming language based on CSP Parallel Computing

**OZ** is a concurrent constraint programming language



# Lenguajes de Programación



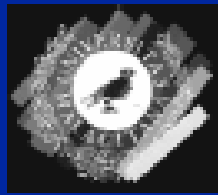
- **Perl** A powerful scripting and string manipulation language.
- **Postscript.** Internet PostScript Resources
- **Prolog** : The Logic Programming Language
- **Python** is an object-oriented scripting and prototyping language which some prefer over Perl, TCL or Scheme

**Sisal:** A high-performance functional language with implicit parallelism for scientific programming

.....

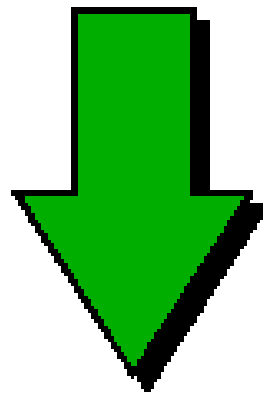
.....

# Lenguajes de Programación



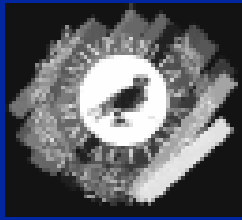
## Clasificación de los Lenguajes de Programación

Clasificación de los lenguajes de programación según sus características



Paradigmas de Programación

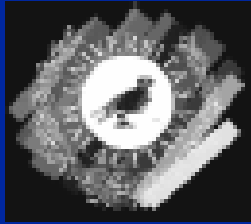
# Lenguajes de Programación



## Paradigmas de Programación

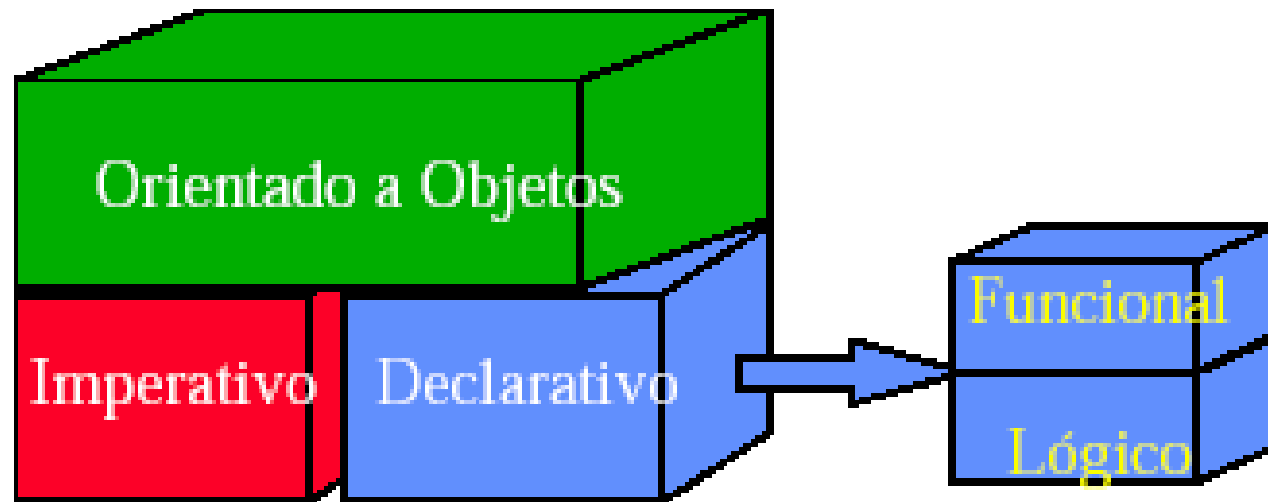
- Un paradigma de programación es un modelo que engloba a ciertos lenguajes que comparten:
  - Elementos estructurales:
    - ¿con qué se confeccionan los programas?
  - Elementos metodológicos:
    - ¿cómo se confecciona un programa?



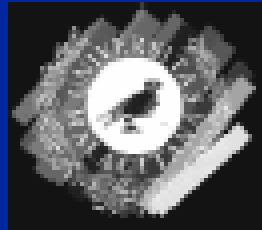


## Paradigmas de Programación

- Consideramos los siguientes paradigmas:



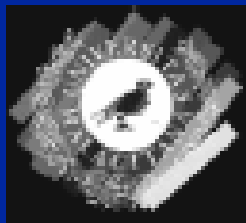
- Los paradigmas no son disjuntos.



## Paradigmas de Programación

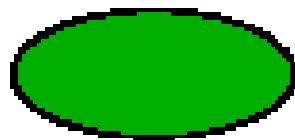
### Programación Imperativa

- Es la más antigua → máquina de Von Neumann
- Un programa es una secuencia de acciones que se realizan en orden.
- Existen herramientas para modificar el orden de ejecución de las acciones.



## Paradigmas de Programación

- Diagramas de Flujo:



Comienzo y terminación  
del algoritmo



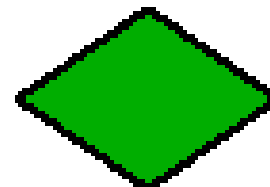
Operación de  
Entrada/Salida



Flujo

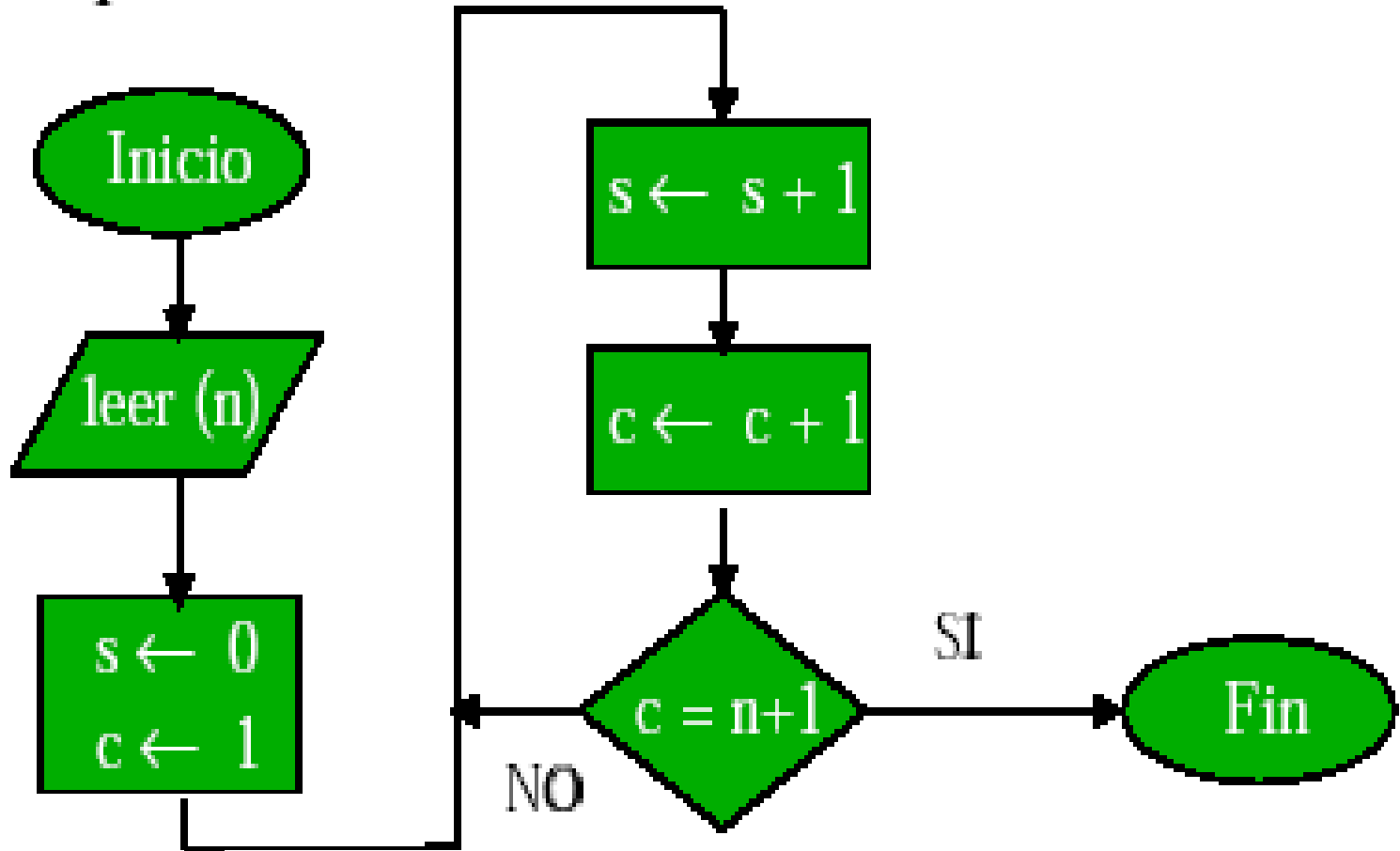


Acción primitiva

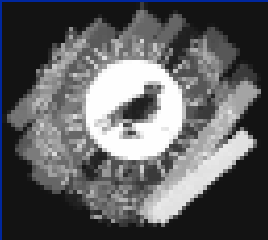


Bifurcación

- Ejemplo:







## Paradigmas de Programación

### Programación Declarativa

- **Programación Funcional:** definición de una serie de funciones.
- **Programación Lógica:** definición de hechos y relaciones lógicas entre éstos.
- No se indica el orden en el que se computa una función o se deriva un nuevo hecho.

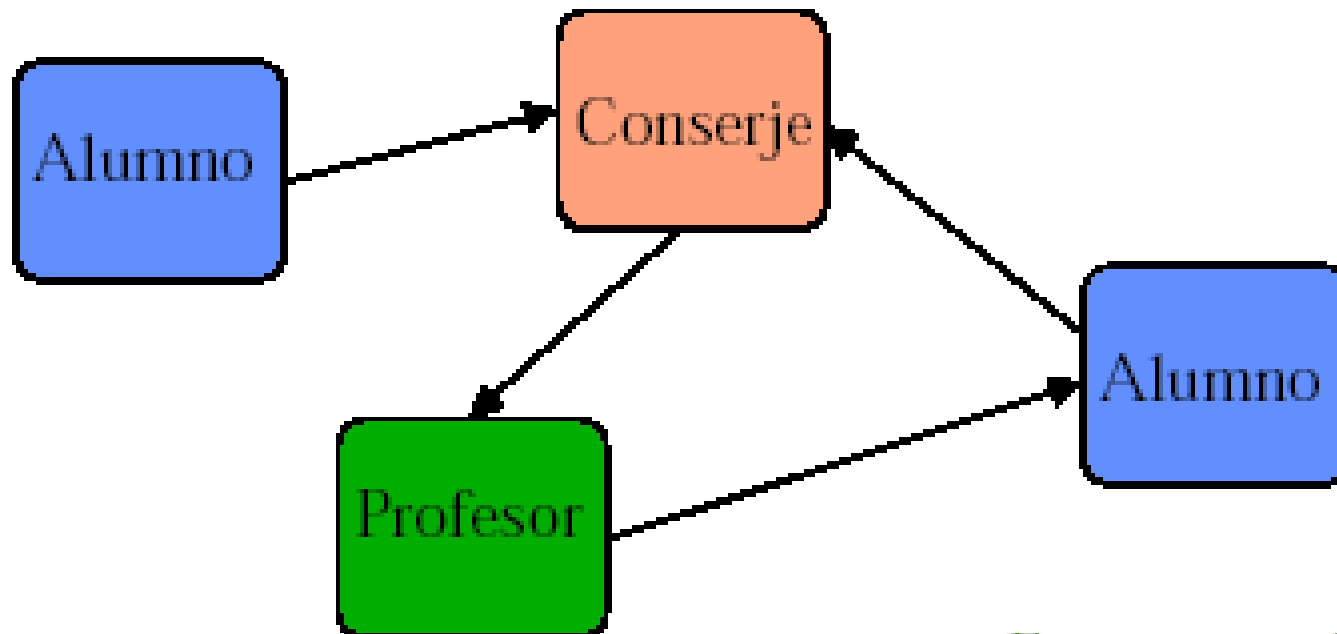




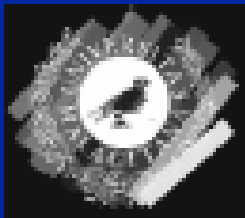
# Paradigmas de Programación

## Programación Orientada a Objetos

- Un programa consiste en una colección de **objetos** que intercambian mensajes.



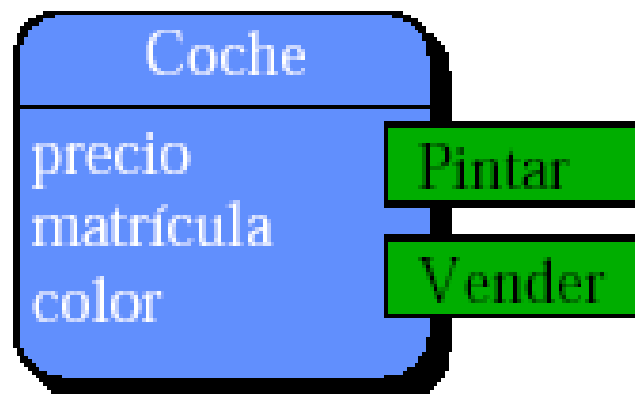




## Paradigmas de Programación

### Programación Orientada a Objetos

- Cada objeto es una entidad que agrupa una cierta información (**estado**) y un conjunto de mecanismos para manipularla (**métodos**).



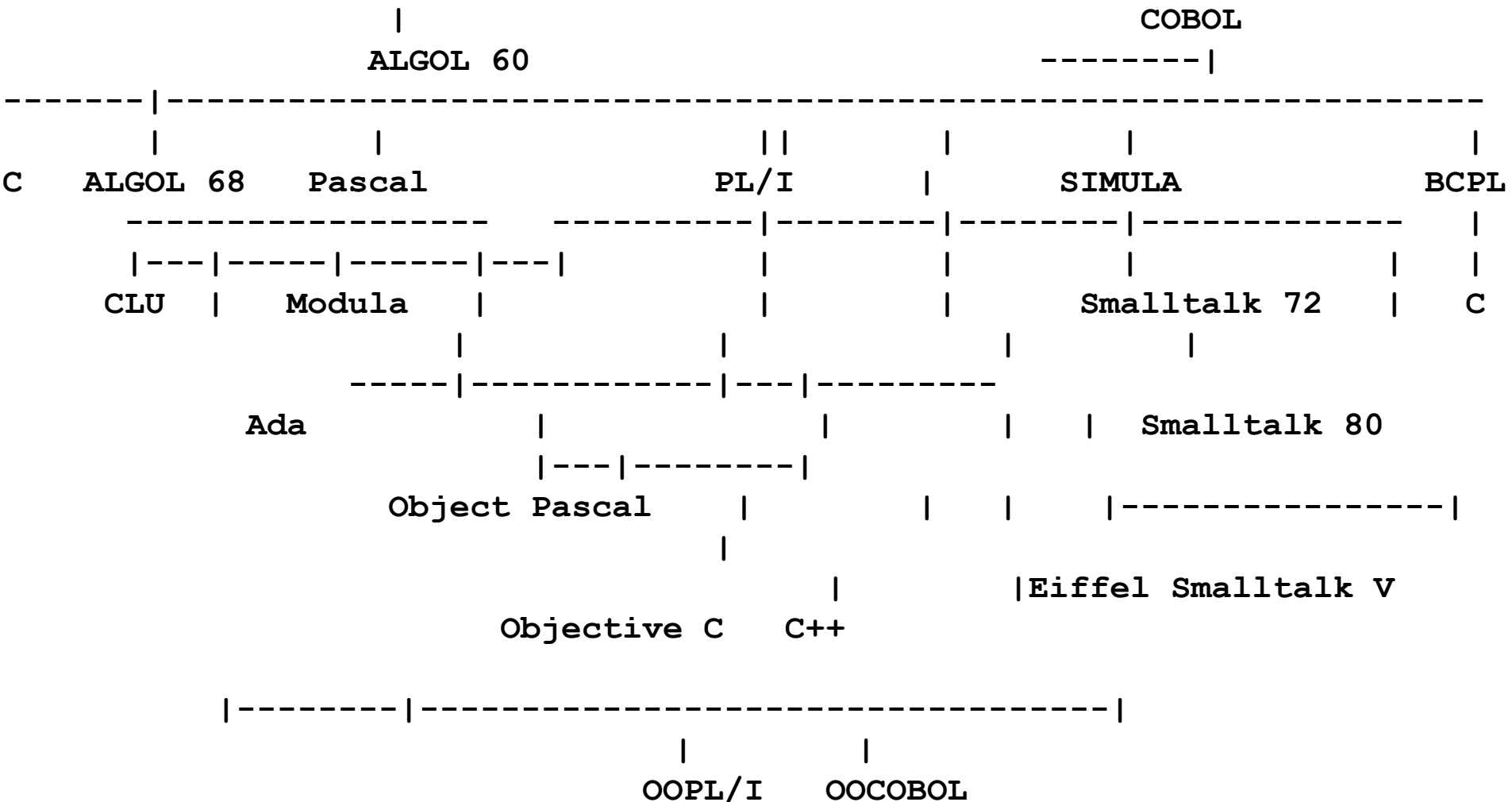




# Lenguajes de Programación

Lenguajes de alto nivel Lenguajes imperativos

## 1. FORTRAN







# Lenguajes de Programación

## Actividad Grupal

**Por equipo, conciba un computador simple hipotético, Conciba un lenguaje para darle instrucciones a esa máquina.**

**Entregar por equipo en la próxima clase.**