

# Lenguajes de programación y sus paradigmas

## Historia

Verónica E. Arriola-Rios

Facultad de Ciencias, UNAM

18 de agosto de 2025



# Antecedentes

- 1 Antecedentes
- 2 Democratización
- 3 Bibliografía

# Antecedentes

**Tabla 1: Antecedentes**

Fecha	Lenguaje	Contribución
1950s		El objetivo era obtener programas eficientes que explotaran al máximo el potencial del hardware.
~1960s		Uso de <i>mainframes</i> , primeras computadoras de uso general. Procesaban datos en lotes. Programadas con tarjetas perforables. Un programa se ejecutaba tras otro. Sus entradas y salidas estaban en archivos.
1957	Fortran	( <i>FORmula TRANslation</i> ) Fue diseñado para aplicaciones de cálculo numérico científico. Sigue en uso hoy en día.

# Antecedentes

Tabla 2: Antecedentes

Fecha	Lenguaje	Contribución
1956	ALGOL	<i>(ALGORithmic Lenguaje)</i> Diseñado como lenguaje universal. Primer lenguaje en usar la gramática BNF de Chomsky para expresar su sintaxis.  Comandos estructurados para control de secuencia usados hoy en día: <code>if then else</code> , <code>while</code> y <code>case</code> .
1960	LISP	<i>(LISt Processor)</i> Lenguaje funcional diseñado por el grupo de John McCarthy para manipular expresiones simbólicas, para Inteligencia Artificial. Usa recolector de basura.

# Antecedentes

Tabla 3: Antecedentes

Fecha	Lenguaje	Contribución
1960	COBOL	( <i>COmmon Busines Oriented Lenguaje</i> ). Aún en uso hoy en día.
1962	Simula	Primer lenguaje orientado a objetos que introduce los conceptos de <i>clase</i> , <i>objeto</i> , <i>sub-tipo</i> y <i>despacho dinámico de métodos</i> .
1970s	Aparición del microchip y las microcomputadoras.	
1970	C	Permite acceder la funcionalidad de bajo nivel de la máquina y programar sistemas que interactúan más con el usuario.

# Antecedentes

Tabla 4: Antecedentes

Fecha	Lenguaje	Contribución
1970	Pascal	Usa el P-code con la finalidad de ser portable, concepto que usa ahora el <i>bytecode</i> de Java. Permite anidar bloques y funciones con complejidad arbitraria.
1970	Smalltalk	Reglas de visibilidad: métodos públicos, variables de instancia privadas.
1970	Prolog	1er lenguaje de programación lógico. En uso hoy en día. Utiliza los algoritmos de unificación y resolución restringida para calcular los valores que satisfacen las restricciones lógicas, así la demostración provee el resultado.

# Democratización

- 1 Antecedentes
- 2 Democratización
- 3 Bibliografía

# Democratización

- En 1977, Ken Olson, presidente de Digital Equipment Corp (empresa productora de minicomputadoras) afirmó que “*no veía ninguna razón por la cual alguien pudiera querer una computadora en su casa*”.
- En 1981 IBM saca Lotus el primer programa tipo hoja de cálculo y la opinión del público empieza a cambiar.
- 1984 Apple saca la Macintosh, con el primer sistema operativo con interfaz gráfica basada en ventanas, iconos y ratón.
- Aparecen los primeros sistemas *embebidos*. Son sistemas compuestos de computadoras conectadas a sistemas físicos que realizan tareas de control como motores, maquinaria industrial, aparatos electrodomésticos, aviones, etc. Por lo que aparecen los lenguajes para respuesta en *tiempo real*.



## PCs

Tabla 5: PCs

Fecha	Lenguaje	Contribución
1980s		Dominados por el desarrollo de la computadora personal (PC).
1986	C++	Stroustrup añade orientación a objetos a C, casi sin afectar la compatibilidad con C. Uso de plantillas y herencia múltiple.
80s	Ada	Concepto de tarea, temporizadores, ejecución concurrente de tareas.
80s	CLP	( <i>Constraint Logic Programming Languages</i> ) Permiten la manipulación de relaciones sobre dominios apropiados: ecuaciones, desigualdades, restricciones sobre cadenas de caracteres, booleanos, reales o dominios finitos (lenguaje CHIP).


Tabla 6: Internet

Fecha	Lenguaje	Contribución
1990s		Aparición de Internet y la <i>World Wide Web</i> con sus lenguajes de marcado: HTML y XML.
1990s	Java	Diseñado para distribuir mejor el trabajo entre servidores y clientes usando <i>Applets</i> . Requiere portabilidad para ejecutar código en varios tipos de cliente. Uso de hilos para ejecución concurrente.
1991	Python	Se enfoca en la legibilidad del código, su filosofía es favorecer la claridad y la simplicidad sobre la complejidad.

# Bibliografía

- 1 Antecedentes
- 2 Democratización
- 3 Bibliografía**

# Bibliografía I

-  Maurizio Gabbrielli, Simone Martini (2010). *Programming Languages: Principles and Paradigms*. Springer. 440 págs. ISBN: 978-607-707-211-9. DOI: [10.1007/978-1-84882-914-5](https://doi.org/10.1007/978-1-84882-914-5).
-  Viso, Elisa y Canek Peláez V. (2012). *Introducción a las ciencias de la computación con Java*. 2a. Temas de computación. Las prensas de ciencias. 571 págs. ISBN: 978-607-02-3345-6.