**HISTORIA DE LOS LENGUAJES DE PROGRAMACION**

Alumno: Josué de Jesús Zapata Moreno

En la década de 1940 fueron creadas las primeras computadoras modernas, con alimentación eléctrica. La velocidad y capacidad de memoria limitadas forzaron a los programadores a escribir programas, en lenguaje ensamblador muy afinados. Finalmente se dieron cuenta de que la programación en lenguaje ensamblador requería de un gran esfuerzo intelectual y era muy propensa a errores.

En 1948, Konrad Zuse publicó un artículo acerca de su lenguaje de programación [Plankalkül](https://es.wikipedia.org/wiki/Plankalk%C3%BCl" \o "Plankalkül). Sin embargo, no fue implementado en su vida y sus contribuciones fueron aisladas de otros desarrollos.

Entre algunos lenguajes importantes que fueron desarrollados en este período se encuentran:

* 1943 - Plankalkül (Konrad Zuse), diseñado, pero sin implementar durante medio siglo
* 1943 - el sistema de codificación ENIAC.
* 1949 - 1954 -- una serie de conjuntos de instrucciones nemotécnicas, como la de ENIAC, comenzando en 1949 con C-10 para BINAC (que luego evolucionaría en UNIVAC). Cada conjunto de instrucciones estaba destinado a un fabricante específico.

En los cincuenta, los tres primeros lenguajes de programación modernos, cuyos descendientes aún continúan siendo utilizados, son:

* FORTRAN (1955), creado por John Backus et al.;
* LISP (1958), creado por John McCarthy et al.;
* COBOL (1959), creado por el Short Range Committee, altamente influenciado por Grace Hopper.

Otro hito a finales de 1950 fue la publicación, por un comité Americano y Europeo de científicos de la computación, de un nuevo “lenguaje para algoritmos”; el Reporte de [ALGOL 60](https://es.wikipedia.org/wiki/ALGOL_60) ("ALGOrithmic Language"). Este reporte consolidó muchas ideas que estaban circulando en aquel entonces, y proporcionó dos innovaciones importantes para los lenguajes de programación:

* [Estructuras de bloques](https://es.wikipedia.org/wiki/Bloque_de_c%C3%B3digo) anidadas: las secuencias de código y las declaraciones asociadas se pueden agrupar en [bloques](https://es.wikipedia.org/wiki/Bloque_de_c%C3%B3digo) sin tener que pertenecer explícitamente a procedimientos separados;
* Ámbito léxico: un bloque puede tener sus propias variables, procedimientos y funciones, invisible al código fuera de dicho bloque, por ejemplo, ocultamiento de información.

Otra innovación, relacionada con esto, fue cómo el lenguaje fue descrito:

* Una notación matemática exacta, [Backus-Naur Form](https://es.wikipedia.org/wiki/Backus-Naur_Form) (BNF), fue utilizada para describir la sintaxis del lenguaje. Todos los subsecuentes lenguajes de programación han utilizado una variante de BNF para describir la porción [libre del contexto](https://es.wikipedia.org/wiki/Gram%C3%A1tica_libre_de_contexto) de su sintaxis.

Algol 60 influenció particularmente en el diseño de lenguajes posteriores, de los cuales algunos se hicieron más populares. Los [grandes sistemas de Burroughs](https://es.wikipedia.org/wiki/Grandes_sistemas_de_Burroughs) fueron diseñados para ser programados en un subconjunto extendido de Algol.

Las ideas fundamentales de Algol se continuaron, produciendo Algol 68:

* la sintaxis y la semántica se hizo aún más ortogonal, con rutinas anónimas, un sistema de tipificación recursiva con funciones de orden superior, etc.;
* y no sólo la parte libre del contexto, sino que tanto la sintaxis como la semántica del lenguaje completo fueron definidos formalmente, en términos de una [gramática de Van Wijngaarden](https://es.wikipedia.org/w/index.php?title=Gram%C3%A1tica_de_Van_Wijngaarden&action=edit&redlink=1), un formalismo diseñado específicamente para este propósito.

Las variadas pero poco usadas características de Algol 68 (por ejemplo, bloques simultáneos y paralelos) y su complejo sistema de atajos sintácticos y coerciones automáticas de tipo lo hicieron impopular entre los ejecutores y se ganó una reputación de ser difícil. [Niklaus Wirth](https://es.wikipedia.org/wiki/Niklaus_Wirth" \o "Niklaus Wirth) salió del comité de diseño para crear el sencillo lenguaje [Pascal](https://es.wikipedia.org/wiki/Pascal_(lenguaje_de_programaci%C3%B3n)).

El período comprendido entre finales de 1960 y finales de 1970 trajo un gran florecimiento de lenguajes de programación. La mayoría de los paradigmas de lenguajes de programación más importantes y actualmente en uso se inventaron en este período:

* Simula, inventado en la década de 1960 por [Nygaard](https://es.wikipedia.org/wiki/Kristen_Nygaard" \o "Kristen Nygaard) y [Dahl](https://es.wikipedia.org/wiki/Ole-Johan_Dahl" \o "Ole-Johan Dahl) como un superconjunto de Algol 60, fue el primer lenguaje diseñado para apoyar la [programación orientada a objetos](https://es.wikipedia.org/wiki/Programaci%C3%B3n_orientada_a_objetos).
* [C](https://es.wikipedia.org/wiki/C_(lenguaje_de_programaci%C3%B3n)), en principio un lenguaje de programación de sistemas, fue desarrollado por [Dennis Ritchie](https://es.wikipedia.org/wiki/Dennis_Ritchie) y [Ken Thompson](https://es.wikipedia.org/wiki/Ken_Thompson) en los [Laboratorios Bell](https://es.wikipedia.org/wiki/Laboratorios_Bell) entre 1969 y 1973.
* [Smalltalk](https://es.wikipedia.org/wiki/Smalltalk) (mediados de los 70) proporcionaron un completo diseño de un lenguaje orientado a objetos.
* [Prolog](https://es.wikipedia.org/wiki/Prolog), diseñado en 1972 por [Colmerauer](https://es.wikipedia.org/w/index.php?title=Alain_Colmerauer&action=edit&redlink=1" \o "Alain Colmerauer (aún no redactado)), [Roussel](https://es.wikipedia.org/w/index.php?title=Phillipe_Roussel&action=edit&redlink=1" \o "Phillipe Roussel (aún no redactado)) y [Kowalski](https://es.wikipedia.org/w/index.php?title=Robert_Kowalski&action=edit&redlink=1" \o "Robert Kowalski (aún no redactado)), fue el primer lenguaje de [programación lógica](https://es.wikipedia.org/wiki/Programaci%C3%B3n_l%C3%B3gica).
* [ML](https://es.wikipedia.org/wiki/Meta_Lenguaje) construyó un sistema de tipos polimórfico (inventado por [Robin Milner](https://es.wikipedia.org/wiki/Robin_Milner" \o "Robin Milner) en 1973) en el tope de Lisp, pionero en los [lenguajes funcionales](https://es.wikipedia.org/wiki/Lenguajes_funcionales) de programación con [tipado estático](https://es.wikipedia.org/wiki/Tipado_est%C3%A1tico" \o "Tipado estático).

Cada uno de estos lenguajes generó toda una familia de descendientes, y los lenguajes más modernos cuentan al menos uno de ellos en su ascendencia.

En los años 60 y 70 también suscitó un debate considerable sobre los méritos de la "[programación estructurada](https://es.wikipedia.org/wiki/Programaci%C3%B3n_estructurada)", que esencialmente significaba programación sin el uso de [GOTO](https://es.wikipedia.org/wiki/GOTO). Este debate estaba estrechamente relacionado con el diseño del lenguaje: algunos lenguajes no incluían GOTO, lo que obligó a la programación estructurada en el programador. Aunque el debate se recrudecía acaloradamente en ese momento, casi todos los programadores están de acuerdo en que, incluso en lenguajes que ofrecen GOTO, es una mala práctica de programación usarlo, excepto en raras circunstancias. Como resultado de ello, las generaciones posteriores de diseñadores de lenguajes han encontrado el debate sobre programación estructurada tedioso e incluso desconcertante.

Algunos lenguajes importantes que se desarrollaron en este período fueron:

* 1968 - [Logo](https://es.wikipedia.org/wiki/Logo_(lenguaje_de_programaci%C3%B3n))
* 1969 - [B](https://es.wikipedia.org/wiki/B_(lenguaje_de_programaci%C3%B3n)) (precursor C)
* 1970 - [Pascal](https://es.wikipedia.org/wiki/Pascal_(lenguaje_de_programaci%C3%B3n))
* 1970 - [Forth](https://es.wikipedia.org/wiki/Forth" \o "Forth)
* 1972 - [C](https://es.wikipedia.org/wiki/C_(lenguaje_de_programaci%C3%B3n))
* 1972 - [Smalltalk](https://es.wikipedia.org/wiki/Smalltalk" \o "Smalltalk)
* 1972 - [Prolog](https://es.wikipedia.org/wiki/Prolog" \o "Prolog)
* 1973 - [ML](https://es.wikipedia.org/wiki/Meta_Lenguaje)
* 1975 - [Scheme](https://es.wikipedia.org/wiki/Scheme" \o "Scheme)
* 1978 - [SQL](https://es.wikipedia.org/wiki/SQL) (inicialmente sólo un lenguaje de consulta, posteriormente ampliado con construcciones de programación)
* 1978 - [Modula-2](https://es.wikipedia.org/wiki/Modula-2)

La década de 1980 fueron años de consolidación relativa en los [lenguajes imperativos](https://es.wikipedia.org/wiki/Lenguajes_imperativos). En vez de inventar nuevos paradigmas, se comenzó a trabajar a partir de las ideas inventadas en la década anterior. [C++](https://es.wikipedia.org/wiki/C%2B%2B) combinaba la programación orientada a objetos y la programación de sistemas. El gobierno de Estados Unidos estandarizó [Ada](https://es.wikipedia.org/wiki/Ada_(lenguaje_de_programaci%C3%B3n)), un lenguaje de programación de sistemas destinado a ser utilizado por contratistas de defensa. En Japón y en otras partes, se gastaron enormes sumas investigando los llamados [lenguajes de programación de quinta generación](https://es.wikipedia.org/w/index.php?title=Lenguajes_de_programaci%C3%B3n_de_quinta_generaci%C3%B3n&action=edit&redlink=1) que incorporaban construcciones de la programación lógica. La comunidad de los lenguajes funcionales llevaron a cabo la estandarización de ML y Lisp. La investigación en [Miranda](https://es.wikipedia.org/wiki/Miranda_(lenguaje_de_programaci%C3%B3n)), un lenguaje funcional, con [evaluación perezosa](https://es.wikipedia.org/wiki/Evaluaci%C3%B3n_perezosa), comenzó a tomar fuerza en esta década.

El rápido crecimiento de Internet en la década de 1990 fue el siguiente gran acontecimiento histórico para los lenguajes de programación. Con la apertura de una plataforma totalmente nueva para los sistemas informáticos, Internet creó una oportunidad adoptar nuevos lenguajes. En particular, el lenguaje de programación Java se hizo popular debido a su pronta integración con el navegador web Netscape Navigator, y varios lenguajes de scripting alcanzaron un amplio uso en el desarrollo de aplicaciones personalizadas para servidores web. La década de 1990 no vio ninguna novedad fundamental en los lenguajes imperativos, pero sí mucha recombinación y la maduración de viejas ideas. Esta era comenzó la difusión de los [lenguajes funcionales](https://es.wikipedia.org/wiki/Lenguajes_funcionales). Una filosofía de conducción grande era la productividad del programador. Surgieron muchos lenguajes de “aplicaciones de desarrollo rápido” ([RAD](https://es.wikipedia.org/wiki/RAD)), los cuales usualmente venían con un [IDE](https://es.wikipedia.org/wiki/Entorno_de_desarrollo_integrado), [recolector de basura](https://es.wikipedia.org/wiki/Recolector_de_basura), y eran descendientes de lenguajes anteriores. Todos estos lenguajes eran orientados a objeto. Estos incluían [Object Pascal](https://es.wikipedia.org/wiki/Object_Pascal" \o "Object Pascal), [Visual Basic](https://es.wikipedia.org/wiki/Visual_Basic) y [Java](https://es.wikipedia.org/wiki/Java_(lenguaje_de_programaci%C3%B3n)). Java, en particular, recibió mucha atención. Pero más radicales e innovadores que los lenguajes de RAD eran los nuevos [lenguajes de script](https://es.wikipedia.org/wiki/Script). Estos no descendían directamente de otros lenguajes y ofrecieron nuevas sintaxis e incorporación más liberal de otras características. Muchos consideran estos lenguajes de script más productivos que los lenguajes de RAD, aunque esto se debe a menudo a que es más difícil escribir y mantener largos programas que pequeños programas simples. Sin embargo, no es menos cierto que los programas de script llegaron para convertirse en los más prominentes en la conexión con la Web.

* 1990 - [Haskell](https://es.wikipedia.org/wiki/Haskell" \o "Haskell)
* 1991 - [Python](https://es.wikipedia.org/wiki/Python)
* 1991 - [Visual Basic](https://es.wikipedia.org/wiki/Visual_Basic)
* 1991 - [HTML](https://es.wikipedia.org/wiki/HTML) (lenguaje de marcado de hipertexto)
* 1993 - [Ruby](https://es.wikipedia.org/wiki/Ruby)
* 1993 - [Lua](https://es.wikipedia.org/wiki/Lua" \o "Lua)
* 1994 - [CLOS](https://es.wikipedia.org/wiki/CLOS) (parte del ANSI Common Lisp)
* 1995 - [Java](https://es.wikipedia.org/wiki/Java_(lenguaje_de_programaci%C3%B3n))
* 1995 - [Delphi (Object Pascal)](https://es.wikipedia.org/wiki/Embarcadero_Delphi)
* 1995 - [JavaScript](https://es.wikipedia.org/wiki/JavaScript)
* 1995 - [PHP](https://es.wikipedia.org/wiki/PHP)
* 1996 - [WebDNA](https://es.wikipedia.org/w/index.php?title=WebDNA&action=edit&redlink=1" \o "WebDNA (aún no redactado))
* 1997 - [Rebol](https://es.wikipedia.org/w/index.php?title=Rebol&action=edit&redlink=1" \o "Rebol (aún no redactado))
* 1999 - D