Buenas ! Parece que he llegado un poco tarde y no he podido aportar mucho de los que más conozco. Pero debido a ello, he podido sacar una parte buena que es investigar y encontrar uno que no sabía ni de su existencia.

Os vengo a hablar del lenguaje de programación **Pascal.**

Originario de los años 70 por **Niklaus Wirth** y empleado para enseñar a los universitarios en las asignaturas de programación, aunque después se utilizó también para la programación en el día a día de los profesionales y estaba orientado a cualquier tipo de ordenador (teniendo en cuenta que hasta los años 80 no había ordenadores personales).

De hecho, Pascal fue el lenguaje utilizado para el Apple Lisa y los primeros Macintosh, aunque tuvo que competir con el lenguaje Basic que en aquella época era predominante.

Fue diseñado para programar de una forma “elegante y disciplinada”, para adquirir buenas prácticas de programación, ya que es un lenguaje que obliga a escribir con claridad, apenas sin errores y donde debe primar la organización.

Poco a poco, como la mayoría de lenguajes evolucionó y surgió Object Pascal, (versión de Pascal orientada a objetos) que tuvo más importancia ya que posibilitó la creación de Delphi, entorno de programación para la creación de programas Windows.

De los compiladores a destacar para programar en este lenguaje, son:

* Free Pascal: multiplataforma y posiblemente sea el que se use más hoy en día. Puede integrase en cualquier editor de texto o usarse directamente en la consola
* Lazarus: basado en el anterior e igualmente para crear aplicaciones de escritorio

Ciertamente ha sido desplazado por otros lenguajes como C, pero sigue persistiendo ya que es más fácil que el anterior.

Puestos en situación, vamos a destacar tres características principales de este lenguaje:

* **Claridad** **y Simplicidad** para que otro programador lo entienda sin esfuerzo.
* **Legibilidad y modularidad.** Aunque el código sea infinito debe poder leerse y al poder dividir el programa en procedimientos y funciones, facilita la reutilización y mantenimiento del código.
* **Eficacia** a la hora de ejecutarse y del uso de los mismos recursos del ordenador.
* **Compatibilidad multiplataforma** para ejecutar programas en Windows, macOS y Linux.

Obviamente tiene más características como rapidez de compilación y una buena gestión de librerías, como sucede con otros lenguajes, por eso voy a hacer una pequeña comparativa de ventajas y desventajas respecto a C y C++ con la cual tuvo que competir antaño y que finalmente estas ganaron pero por publicidad.

Ventajas respecto a C

* Rapidez de compilación, facilidad y limpieza, mejor gestión de ficheros de código y un entorno de programación gráfico sencillo y efectivo (Lazarus y Delphi).

Desventajas con respecto a C

* Código generado más lento pero porque GCC está más mantenido, no porque su capacidad sea más lenta. Aún así la diferencia es muy poca.
* Acceso más directo a las librerías del sistema.

Ventajas con respecto a C++

* Sistema de objetos mucho más claro

Desventajas con respecto a C

* Templates

A modo general, las desventajas son claras:

* Menor cantidad de recursos y soporte (poca documentación, herramientas y bibliotecas desactualizadas).
* Puede llegar a ser un código más largo y tedioso al ser más detallado.
* En muchos casos no es adecuado para el desarrollo de aplicaciones y webs en la actualidad, por lo que pocas empresas lo utilizan.