



**UNIVERSIDAD  
DEL QUINDÍO**

**Unidad 5 - E.A. 1  
CLASIFICACIÓN Y TIPOS DE SOFTWARE - LENGUAJES DE PROGRAMACIÓN**

Autor  
**Robinson Pulgarín Giraldo**



# Clasificación y tipos de software - Lenguajes de programación



- ☰ Competencias y Resultados de Aprendizaje
- ☰ Ruta Metodológica
- ☰ Introducción a la Temática
- ☰ Enseñanzas
- ☰ Resumen de la Temática
- ☰ Glosario
- ☰ Referencias

# Competencias y Resultados de Aprendizaje



- Identifica la clasificación y los diferentes tipos de software existentes en el mercado para así, poder usarlos de acuerdo a sus funcionalidades y características.
- Reconoce la tecnología como soporte básico en la solución u optimización de problemas, abordada desde las herramientas que el mundo digital ofrece para así, poder plantear soluciones acordes a los requerimientos que se presenten.



# Ruta Metodológica

## Recomendaciones Generales:

Para lograr el éxito en el desarrollo de este espacio de aprendizaje, por favor, siga las siguientes recomendaciones:

 **Nota:** para interactuar con este recurso de clic sobre los íconos (+) que se encuentran titilando dentro de la imagen.



## Requisitos:

Historia y evolución de la computadora y el procesamiento de datos.



- **Para cualquier tipo de consulta sobre el proceso, haga uso del foro de dudas.**
- Realice las actividades autónomas, si bien es cierto que no generan nota, son fundamentales para autoevaluarse en cuanto a la apropiación del tema estudiado.
- Revise y estudie detenidamente cada uno de los materiales (lecturas, videos, imágenes, infografías, etc.) que se le ofrecen para garantizar su aprendizaje y apropiación de la temática.
- Tenga presente la fecha de inicio y cierre de cada una de las actividades planteadas.
- Antes de iniciar el desarrollo de cada una de las actividades propuestas, por favor lea con cuidado todos los momentos que a continuación se describen, para tener primero una visión global de la pretensión de este espacio de aprendizaje, las actividades que debe realizar y los tiempos programados para ellas.
-

# Introducción a la Temática



Esta temática es complementaria a otros espacios de aprendizaje que se han desarrollado hasta el momento, entre los cuales, se encuentran la Historia de la computación, las Clases de dispositivos y las Tendencias de la ingeniería de sistemas y computación.

Previamente se abordó el tema de hardware y su relación con el software, en él se pudo evidenciar que ambos no pueden funcionar de forma independiente, ellos se complementan. De la misma forma, se han nombrado algunos lenguajes de Programación, pero sin profundizar en ellos, ya que se estudiarán con más detalle en los espacios académicos que corresponden al área de Algoritmia y Programación del plan de estudios del Programa.

Ahora bien, a partir de este momento se abordarán estas temáticas, partiendo desde su definición, características y diferentes clasificaciones. Inicialmente se estudiará el software en general y al finalizar se hará un análisis detallado de que son los lenguajes de programación, su desarrollo y su importancia.

## 1. CLASIFICACIÓN Y TIPOS DE SOFTWARE

Según el estándar 729 de la IEEE, software “es el conjunto de los programas de cómputo, procedimientos, reglas, documentación y datos asociados que forman parte de las operaciones de un sistema de computación”.

**Figura 1.** Software y CD



**Fuente:** Internet (Licencia libre).



En esta definición debe entenderse que, un programa es un conjunto de instrucciones escritas en un lenguaje de programación que permiten que la computadora ejecute una tarea determinada, de igual manera, un sistema de computación, no se limita solamente a las computadoras tradicionales, en ese sistema deben tenerse en cuenta los nuevos teléfonos celulares, las tablets, calculadoras y hasta otros dispositivos que también hace uso de software: televisores inteligentes, autos, equipos electrodomésticos, entre muchos más.

A parte de esta definición técnica, se tienen otras menos formales, tales como:

- Parte no tangible o blanda de la computadora.
- Equipamiento lógico de la computadora.
- Es el moderador entre la computadora y el usuario, el cual permite controlar y coordinar los dispositivos de hardware.
- Todos las aplicaciones o programas que se instalan en un equipo de procesamiento de datos.

### **1.1 CLASIFICACIÓN DEL SOFTWARE**

El software se clasifica de acuerdo a la función que ejecute dentro de un sistema de cómputo. Hay tres categorías básicas:

- Software del sistema.
- Software de aplicación.
- Software de desarrollo o de programación.

## **SOFTWARE DEL SISTEMA**

El software del sistema es un conjunto de programas escritos para dar servicio a otros programas. Se caracteriza por: gran interacción con el hardware de la computadora, uso intensivo por parte de usuarios múltiples, operación concurrente que requiere la secuenciación, recursos compartidos y administración de un proceso sofisticado (Pressman, 2010).

Dicho de otra manera, este software es el encargado de interactuar y gestionar los dispositivos de hardware de un sistema de cómputo (microprocesador, periféricos, dispositivos de comunicación).

El Software del sistema se subdivide en tres categorías:

- Sistemas operativos
- Controladores de dispositivos o drivers
- Herramientas de diagnóstico

**Figura 2.** Categorías software



**Fuente:** Pulgarín, R. (2020).

**Sistema Operativo (S.O):** es el software básico que presenta una interfaz entre los programas del computador, los dispositivos de hardware y el usuario. Todo equipo de cómputo debe tener instalado un S.O para su funcionamiento (Portalsenior, 2010).

El S.O gestiona y controla las actividades de los diferentes dispositivos del equipo, tales como: microprocesador, memoria principal y secundaria, dispositivos de entrada y de salida, GPS, entre otros.

El conjunto de programas que conforman el S.O. indican como usar los circuitos primarios y los dispositivos conectados al equipo de cómputo. Además, permiten la transferencia de datos entre programas y el hardware (Vasconcelos, 2018).

En el mercado existen varios sistemas operativos, algunos de ellos usados en computadoras, otros en dispositivos móviles y en dispositivos inteligentes como televisores o relojes:

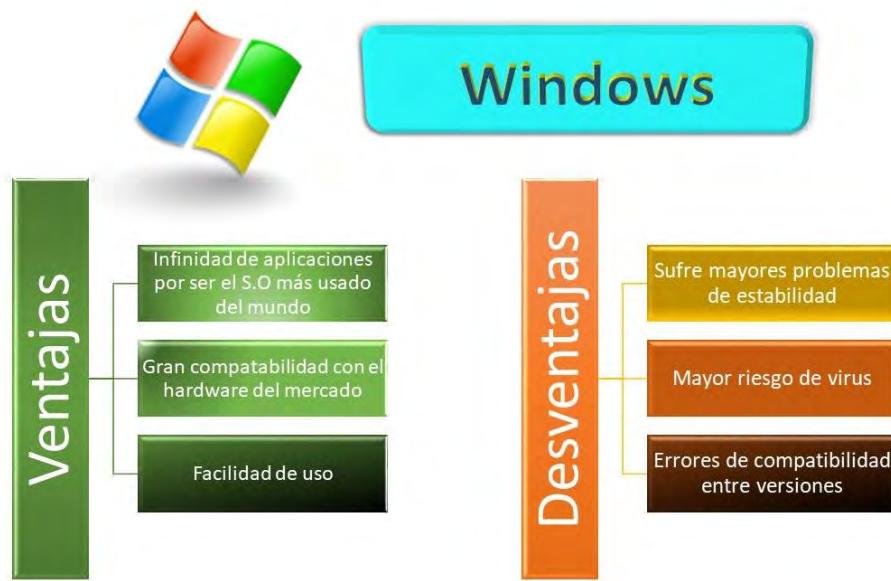
- Solaris
- Unix
- OpenBSD
- Wave OS
- webOS
- Haiku
- ReactOS
- BeOS
- Android
- iOS
- Windows
- Phone Symbian
- OS WatchOS
- Android Wear

En computadoras, los sistemas operativos más comunes son:

- Windows
- Linux y sus diferentes distribuciones
- Mac Os

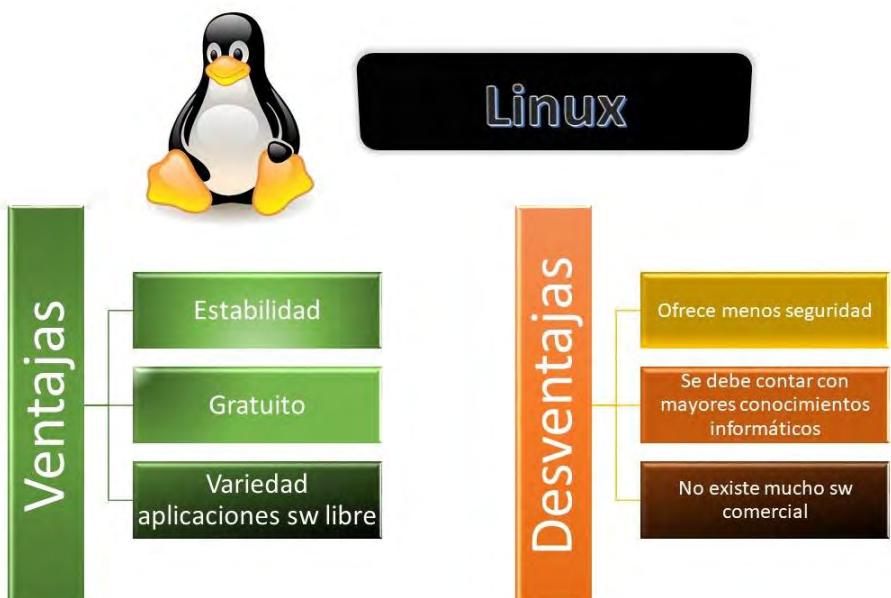
En las figuras 1, 2 y 3 se muestran las principales ventajas y desventajas de estos tres S.O.

**Figura 3.** Ventajas y desventajas del S.O Windows



**Fuente:** Adaptación, (Universidad Politécnica de Valencia, 2010).

**Figura 4:** Ventajas y desventajas del S.O Linux



**Fuente:** Adaptación, (Universidad Politécnica de Valencia, 2010).

Figura 5: Ventajas y desventajas del Mac OS

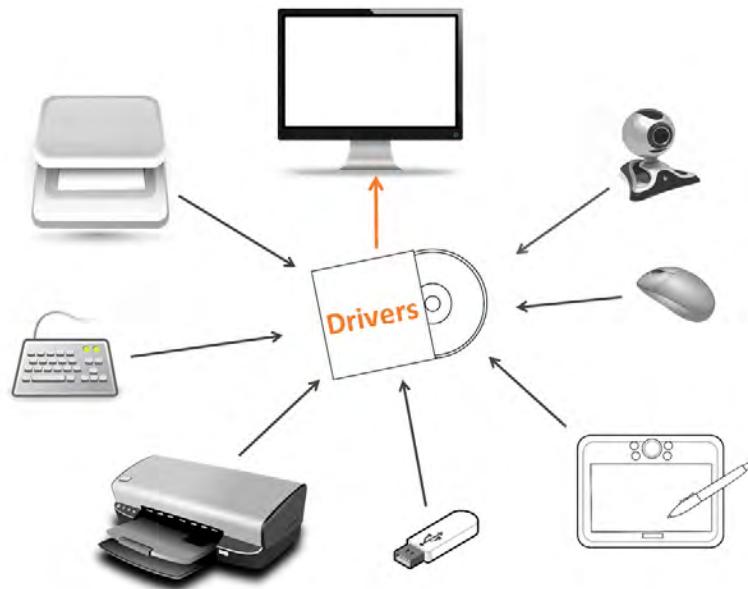


**Fuente:** Adaptación, (Universidad Politécnica de Valencia, 2010).

**Controladores de dispositivos o drivers:** son programas que trabajan directamente sobre el hardware y permiten la comunicación entre los diferentes dispositivos del equipo de cómputo y el sistema operativo.

Cada vez que el usuario requiera el servicio de alguno de los dispositivos del equipo, el SO establecerá la conexión a través del driver adecuado (Vasconcelos, 2018).

**Figura 6.** Controladores o Drivers



**Fuente:** Pulgarín, R. (2020).

**Herramientas de diagnóstico:** han sido creadas con el fin de monitorear, buscar y hasta resolver determinados problemas en el funcionamiento de los dispositivos de la computadora: discos duros, tarjetas de red, tarjetas de memoria, procesador, etc.

### **SOFTWARE DE APLICACIÓN**

Este tipo de software está construido con el propósito de atender necesidades comunes de un grupo de usuarios. Es desarrollado por programadores individuales o grandes compañías productoras de software.

Dentro de este grupo de software se encuentran aplicaciones como:

- Procesamiento de texto.
- Reproductores de multimedia.
- Hojas de cálculo.
- Presentaciones.
- Software educativo.
- Videojuegos.
- Navegadores para internet.
- Tiendas virtuales.
- Software Web.
- Diseño gráfico.

**Figura 7.** Software Aplicación



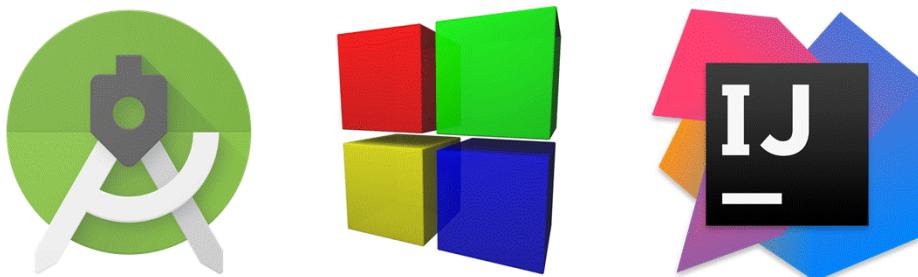
**Fuente:** Pulgarín, R. (2020).

## **SOFTWARE DE DESARROLLO O DE PROGRAMACIÓN**

Son herramientas con las que los programadores construyen los diferentes programas. En esta categoría se encuentran:

- Editor de texto: permite la edición del código fuente de los programas.
- Compilador: traduce un programa completo, escrito en un lenguaje de programación, a código de máquina.
- Intérprete: es similar a un compilador, es decir, traduce un programa a código de máquina, pero solo lo hace a medida que se van ejecutando las instrucciones.
- Depurador (debugger): es un programa que se usa para buscar y corregir errores en la programación.
- IDE Integrated Development Environment o Entorno de Desarrollo Integrado: es un software que agrupa más de una de las categorías anteriores y de esta manera facilita el trabajo de los programadores. Los más básicos proporcionan un editor de código fuente, una herramienta asistente de construcción de código y un depurador.

**Figura 8.** Software programación



**Fuente:** Pulgarín, R. (2020).

Todos estos tipos de software pueden conseguirse de diferentes formas o presentaciones, la más convencional y que está en vía de desaparición es en medios de almacenamiento como CD, DVD o Memorias USB. De igual forma, hay software que viene integrado dentro de equipos o artefactos de diferente clase, por ejemplo: en un horno microondas, en un vehículo, en un televisor inteligente; a este tipo de software se le conoce como software embebido o empotrado. La forma más reciente es conseguirlo es la nube; es decir, en los servidores de las empresas que lo crean y comercializan, ellas lo ofrecen para que sea usado desde la nube o descargado e instalado en los equipos de cómputo.

Con las nuevas tecnologías de equipos Smart (teléfonos, tablets, televisores, etc) se usan las tiendas de aplicaciones para adquirir todo este tipo de software.

**Figura 9. APPS**



**Fuente:** Internet (Licencia libre).

## 1.2 TIPOS DE LICENCIA EXISTENTES DE SOFTWARE

Citando a (Vercelli, 2009) quien definió el concepto de licencia en su tesis doctoral

Repensando los bienes intelectuales comunes, se tiene:

Una licencia es un instrumento jurídico de carácter privado [declaración, acuerdo, contrato] que permite expresar los derechos de autor y los derechos de copia sobre una obra intelectual. Es decir, las licencias indican qué se puede y qué no se puede hacer con una obra intelectual.

En el mencionado instrumento jurídico existe un licenciante (titular de los derechos) y un licenciatario (usuario final del software). Decir que se adquiere un software es una afirmación incorrecta, lo que se está adquiriendo realmente, es una licencia de uso. Cuando Usted baja una app para instalar en su dispositivo móvil, no se está convirtiendo en el dueño de esa aplicación, así haya pagado por ella, lo que está adquiriendo es el derecho a instalarla en su dispositivo y a usar dicha app.

**Figura 10.** Licencias software



**Fuente:** Pulgarín, R. (2020).

En la actualidad existen varias categorías principales de acuerdo a la forma de distribución o licencia que se tenga del software:

- Freeware.
- Shareware.
- Propietario o privativo.
- Free.
- Open source o código abierto.

Las licencias de tipo **Freeware** se usa en aquel software que no tienen ningún costo para el usuario final. Puede ser usado por tiempo ilimitado.

Cuando un software está catalogado como **Shareware**, indica que fue concebido para que el usuario final puede evaluar su desempeño, puede estar limitado por un tiempo de uso o en sus funciones, es decir, no todas las funcionalidades están disponibles, una vez el usuario decida comprarlo ya no habrá limitaciones.

**Software Propietario o Privativo.** El usuario final solamente tiene derecho a usarlo. Generalmente es software de pago, aunque hay algunas aplicaciones que son gratuitas. No se tiene acceso al código fuente.

**Software Libre (free software).** Este tipo de licencia faculta al usuario para que pueda realizar diferentes operaciones sobre el software: usarlo, estudiarlo, modificarlo y distribuirlo. Debido a que la palabra free del inglés, suele traducirse como libre o gratis, el término libre ha ocasionado algunas ambigüedades, por lo que se ha venido empleando indistintamente el término Software Open Source, es decir, Software de código abierto, pero como se verá más adelante existe una diferencia entre ellos.

Para que un programa pueda estar en la categoría de software libre debe cumplir con los cuatro grados de libertad del software libre:

- **Libertad 0:** libertad de usar el programa con cualquier propósito.
- **Libertad 1:** libertad de estudiar cómo funciona el programa y si es necesario modificarlo, adaptándolo a las necesidades del usuario.
- **Libertad 2:** libertad de distribuir copias del software, gratis o con pago.
- **Libertad 3:** libertad de realizar mejoras y hacerlas públicas a los demás, de tal forma que la comunidad se beneficie.

De lo anterior se deduce que, para poder cumplir con los grados de libertad 1 y 3, es imprescindible contar con el código fuente.

Hay que tener en cuenta que la palabra libre no significa gratis. En el mercado existe software gratis que no cumple con las 4 libertades antes enunciadas.

A continuación, los invito a revisar el siguiente video donde se explica que es Software Libre.

#### **Video. El software libre, de las TIC en un CLIC**

Fundación CTIC (2008, 04, 28). El software libre, de las TIC en un CLIC [Archivo en video]. Disponible en: [https://www.youtube.com/watch?v=9NrPGDtqv50&list=PL9Gv-fXZqn8BefVkj7FFQGkZvjDYuxYL\\_&index=14](https://www.youtube.com/watch?v=9NrPGDtqv50&list=PL9Gv-fXZqn8BefVkj7FFQGkZvjDYuxYL_&index=14)

**VER VIDEO**

**Open source o código abierto.** Su principal característica es que el código fuente está disponible sin ninguna restricción. Inicialmente, como se dijo anteriormente, se usó como sinónimo del software libre entendiéndose libre como libertad y no como gratuidad. El tener código abierto, no es sinónimo de libertad.

Aunque el software libre y el software de código abierto, tienen muchas similitudes en lo práctico, sus filosofías son completamente diferentes. Para (Stallman, 2008) creador del movimiento del software libre ambos conceptos son totalmente diferentes:

Las expresiones «software libre» y «código abierto» se refieren casi al mismo conjunto de programas. No obstante, dicen cosas muy diferentes acerca de dichos programas, basándose en valores diferentes. El movimiento del software libre defiende la libertad de los usuarios de ordenadores, en un movimiento en pro de la libertad y la justicia. Por contra, la idea del código abierto valora principalmente las ventajas prácticas y no defiende principios. Esta es la razón por la que estamos en desacuerdo con el movimiento del código abierto y no empleamos esa expresión.

Continúa en su escrito, afirmando que:

Ambos describen casi la misma categoría de software, pero representan puntos de vista basados en valores fundamentalmente diferentes. Para el movimiento del software libre, el software libre es un imperativo ético, respeto esencial por la libertad de los usuarios. En cambio, la filosofía del código abierto plantea las cuestiones en términos de cómo «mejorar» el software, en sentido meramente práctico. Sostiene que el software privativo no es una solución óptima para los problemas prácticos que hay que resolver.

Para terminar con el tema de licencias de software, es importante tener presente que, en los actuales modelos de negocio, en el software en el que se debe pagar por su uso, existe la posibilidad de realizar el pago una sola vez y usarlo indefinidamente, o también existe el pago por mensualidades o por diferentes períodos de tiempo, así mismo, se puede limitar al número de usuarios finales que tendrá.

Con el propósito de tener mayor claridad de la diferencia entre Open Source y Software libre, los invito a revisar el siguiente video.

### **Video. Diferencia entre Open Source y Software libre por Esteban Azofeifa**

Espectro Canal UCR. (2016, 11, 17). Diferencia entre Open Source y Software libre por Esteban Azofeifa [Archivo en video]. Disponible en: <https://www.youtube.com/watch?v=zgfw2hIIoIA>

**VER VIDEO**

## 2. LENGUAJES DE PROGRAMACIÓN

**Figura 11.** Lenguajes



**Fuente:** Internet (Licencia libre).

Aunque este tema será tratado en profundidad en los espacios académicos del área de Programación, en este espacio de aprendizaje se dará una teoría general con el propósito de ambientar al estudiante en este campo de la computación.

En los libros de texto Lógica de Programación e Introducción a la Programación en C, (Gutiérrez, Herrera, Pulgarín, 2018), los autores plasman unas definiciones muy precisas y unos hitos históricos en la evolución de algunos de los más importantes lenguajes usados en la Programación. A continuación, se reproducen algunos apartes de estos textos.

Un lenguaje de programación, proporciona un conjunto de instrucciones que deben respetar ciertas reglas (Sintaxis) y que expresan un significado (Semántica). Por medio de estas instrucciones, es posible crear un programa que de la solución a un problema determinado. Mediante estos lenguajes de Programación es que se construyen los diferentes tipos de software que se estudiaron en el apartado anterior.

Los programas, son una forma de escribir un algoritmo que pueda ser ejecutado por una máquina, ya sea un computador de escritorio, portátil o incluso un dispositivo móvil como un teléfono inteligente, entre otros dispositivos programables. Mediante un programa es posible controlar tanto elementos físicos (hardware) como lógicos (software) de la máquina.

Para que una máquina pueda ser programada, tal y como lo entendemos hoy en día, es necesario que ella posea al menos un procesador.

Los procesadores son circuitos integrados que pueden ser programados mediante un lenguaje específico de cada uno denominado lenguaje de máquina. El lenguaje de máquina es ideal para las máquinas; pero, es difícil de comprender para las personas, ya que se escribe únicamente mediante números. Allí comienza la necesidad de facilitar la programación de la máquina y comienza todo un desarrollo en la línea de los lenguajes de programación.

Un lenguaje no numérico, pero que es traducido directamente al lenguaje de máquina, es el llamado lenguaje Ensamblador. Por ser un lenguaje tan cercano al lenguaje de máquina se considera que es un lenguaje de bajo nivel. Sin embargo, realizar todo tipo de tareas en lenguaje Ensamblador, aunque posible, es dispendioso debido a sus propias características propias, cuyas instrucciones son poco expresivas, en comparación con otros lenguajes de más alto nivel; complicando la labor de programar tareas no triviales. Aunque al final, todos los lenguajes de programación deben ser traducidos al único lenguaje que el procesador comprende, el lenguaje de máquina; no obstante, esta es una tarea oculta para las personas.

Todo lo descrito en los párrafos anteriores se puede visualizar en la siguiente infografía, por lo tanto, los invito a analizarla y comprender este proceso que es mostrado paso a paso:

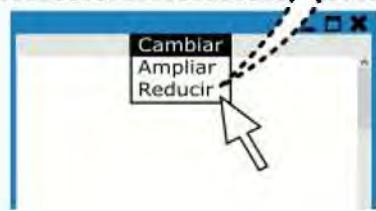
Figura 12. Código fuente

 [WWW.PROYECTOAUTODIDACTA.COM](http://WWW.PROYECTOAUTODIDACTA.COM) POR IVÁN LASSO CLEMENTE

## CÓDIGO FUENTE

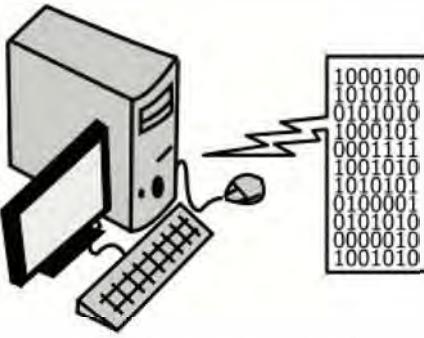
UN PROGRAMA ES UN CONJUNTO DE INSTRUCCIONES QUE SE LE DA A UNA COMPUTADORA PARA QUE HAGA ALGO.

abre el archivo - mira el tamaño que tiene - si el archivo es mayor que 50 Kb y si el archivo es mayor que 1024 pixeles entonces reducir el ancho a 800 pixeles y reducir la calidad hasta que el archivo pese 25 Kb



PARA HACER UN PROGRAMA SE EMPLEA UN LENGUAJE DE PROGRAMACIÓN.

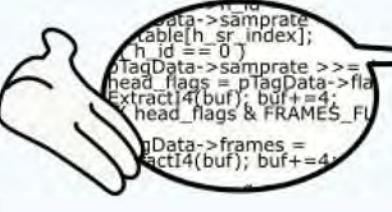
LA COMPUTADORA SÓLO ENTIENDE UN LENGUAJE: EL SUYO. Y SÓLO PODEMOS HABLARLE CON ÉL.



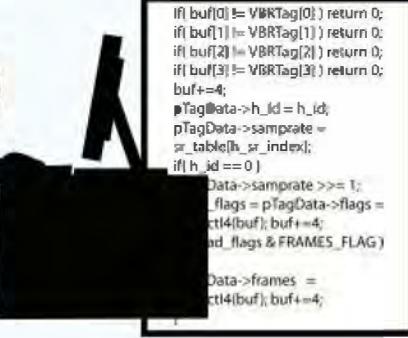
SU LENGUAJE ES EL CÓDIGO MÁQUINA (O BINARIO), UNA SERIE DE 1 Y 0.

COMO ESO ES MUY COMPLICADO, SE INVENTARON LOS LLAMADOS LENGUAJES DE PROGRAMACIÓN.

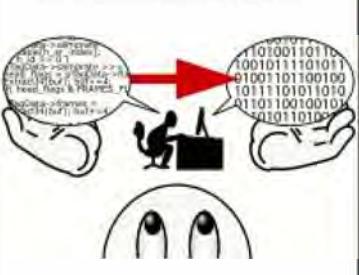
TODAS ESAS ORDENES PARA LA COMPUTADORA QUE FORMAN EL PROGRAMA Y QUE ESTÁN ESCRITAS EN UN LENGUAJE DE PROGRAMACIÓN SE LLAMAN CÓDIGO FUENTE.



SON LENGUAJES PARECIDOS A LOS NATURALES (LOS QUE USAMOS LOS HUMANOS), PERO CON UNA SERIE DE PALABRAS ESPECIALES CON UNA ESTRUCTURA ESPECIAL.



UNA VEZ QUE ESTÁ ESCRITO EL CÓDIGO FUENTE DE UN PROGRAMA, ES NECESARIO SOTENERLO A UN PROCESO QUE SE LLAMA "COMPILACIÓN".



COMPILAR VIENE A SER TRADUCIR EL LENGUAJE DE PROGRAMACIÓN EN EL QUE ESTÁ ESCRITO A OTRO QUE LA COMPI SI ENTIENDE: EL CÓDIGO MÁQUINA.

ESTE ARCHIVO COMPILADO EN CÓDIGO MÁQUINA SE LLAMA ARCHIVO BINARIO O EJECUTABLE: EL PROPIO PROGRAMA.



UNA VEZ UN PROGRAMA ESTÁ COMPILADO EN UN BINARIO, YA PUEDE UTILIZARSE.

PERO YA NO PUEDE VOLVERSE A TRADUCIR A UN LENGUAJE DE PROGRAMACIÓN PARA QUE OTRO PROGRAMADOR PUEDA VER COMO ESTÁ HECHO.





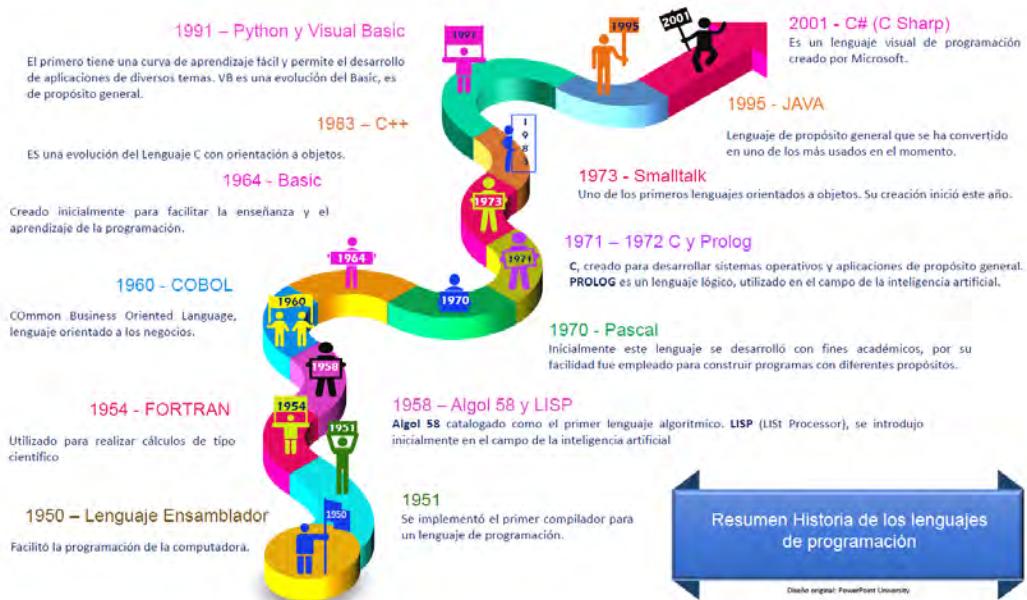
Para escribir código fuente, existen muchos lenguajes de programación, unos más fáciles de comprender y utilizar que otros; algunos con propósitos específicos y otros de propósito general.

Un buen programador, debe comprender este universo de lenguajes y usar a su favor los beneficios de cada lenguaje según la necesidad actual en la solución de un problema. Los lenguajes son herramientas y como tal, deben ser utilizadas para resolver las tareas para las que fueron diseñadas.

El primer lenguaje de programación fue Plankalkul (plan de cálculo) propuesto por Konrad Zuse en 1946, no obstante, permaneció en la teoría hasta el año 2000, año en el que fue implementado en la Universidad Libre de Berlín. Otro lenguaje importante es Fortran, creado por John Backus en 1953. En la actualidad, existen versiones modernas de este lenguaje, que es utilizado principalmente en el campo científico, ya que está especializado en cálculos matemáticos.

Luego de la aparición de estos lenguajes, se empezaron a desarrollar otros con características y propósitos diferentes. En la figura 5, se aprecian algunos hitos representativos que marcaron la historia de la programación.

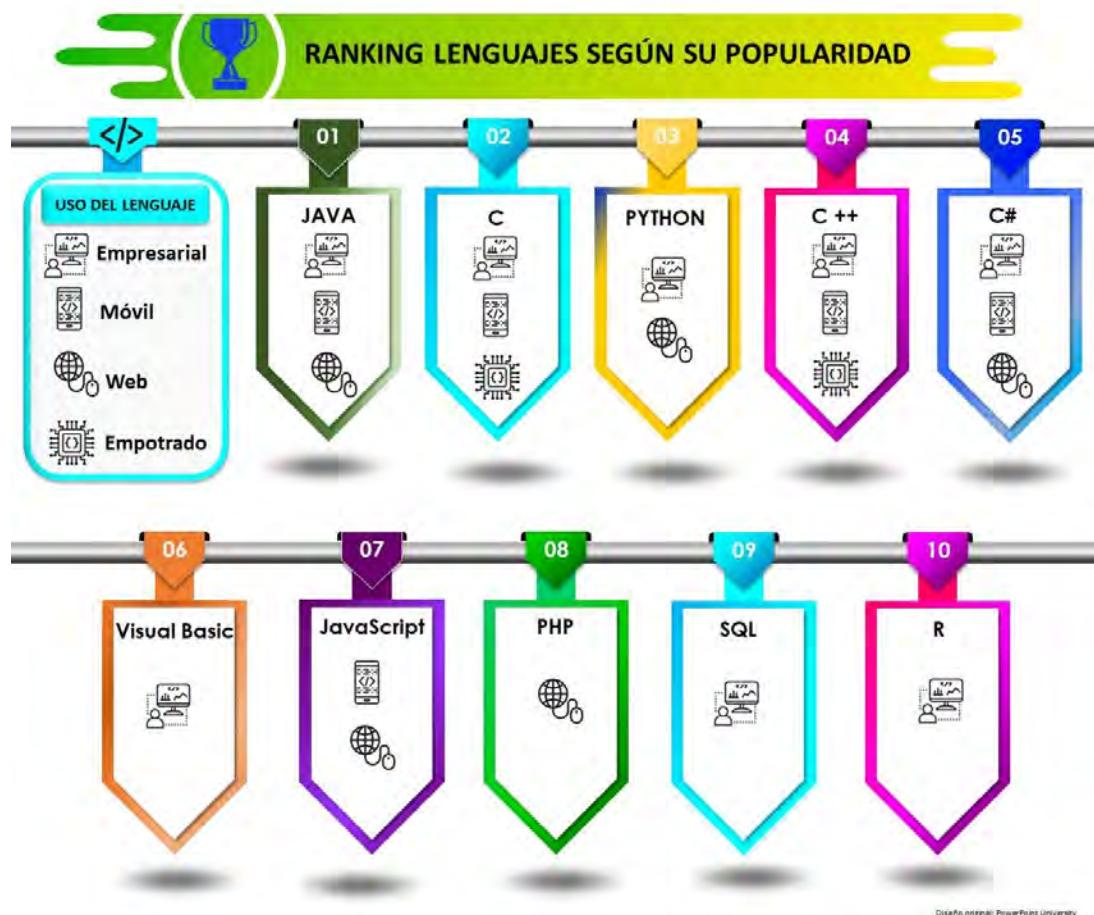
**Figura 13.** Hitos en el desarrollo de la programación



Fuente: Pulgarín, R. (2020).

Varios de estos lenguajes han perdurado hasta el momento y han sido muchos los que se han derivado de ellos. En la figura 6, se muestra el ranking de popularidad que se estableció para el año 2020 de los lenguajes actuales, según TIOBE y PYPL, dos sitios que ofrecen datos estadísticos en internet.

**Figura 14.** Ranking lenguajes según su popularidad



**Fuente:** Pulgarín, R. (2020).

Tengan presente, que estas posiciones varían de una manera bastante dinámica, acorde a las tendencias que se vayan dando en el mundo de la tecnología.

Los invito a ver en un corto video cómo puede variar esta escala a través del tiempo, con él se pueden dar una idea de lo rápido que los lenguajes ganan y pierden popularidad. Es por eso, que lo fundamental para programar, no es aprender un lenguaje, sino tener una sólida formación en lógica, así solamente necesitará aprender la sintaxis de cualquier lenguaje y construir programas con gran facilidad.

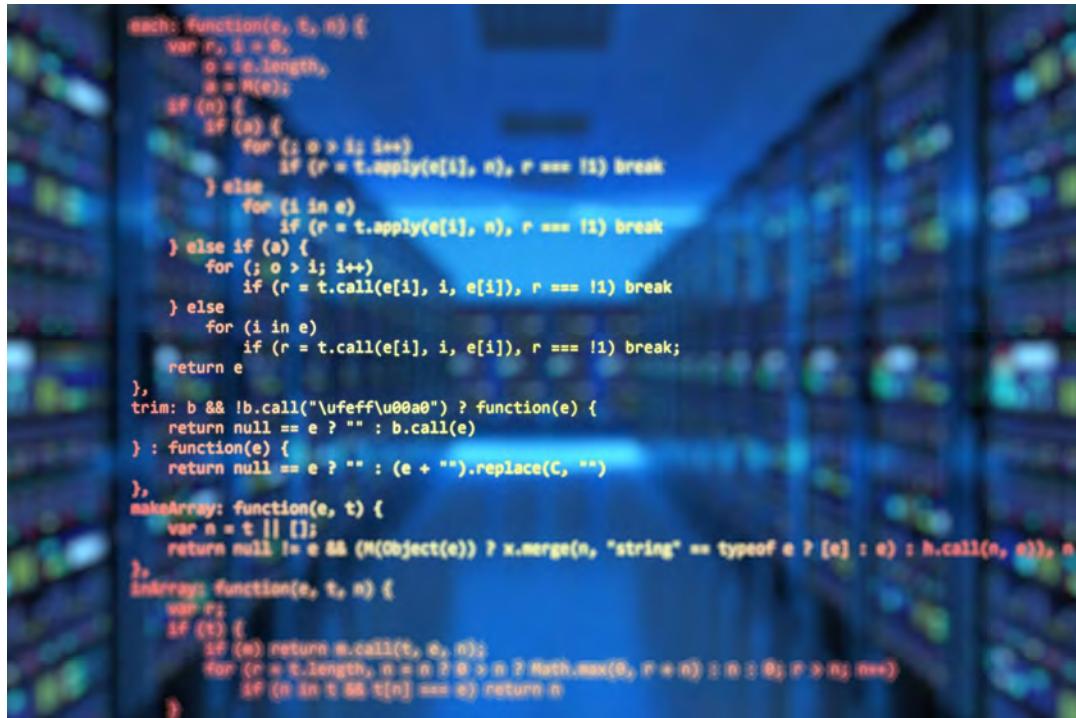
#### **Video. Lenguajes de programación más populares 1965 – 2020**

Información por minuto. (2020, 02, 10). Lenguajes de programación más populares 1965 – 2020 [Archivo de video]. Consultado el: 2020, 04, 16 en: <https://youtu.be/dYBttoXxNf8>

**VER VIDEO**

Es imposible que alguien conozca todos los lenguajes que existen, la lista de ellos es demasiado extensa. Para que se formen una idea de los diferentes lenguajes que existen en el momento, los invito a que visiten el siguiente enlace de Wikipedia, en la que usando la inteligencia colectiva se ha realizado una muy completa relación de los lenguajes de programación más relevantes.

**Figura 15.** Programación



**Fuente:** Wikipedia (2014)

#### Anexo: Lenguajes de Programación

Wikipedia. (2014, 07, 26). Anexo: Lenguajes de Programación [Sitio Web]. Disponible en: [https://es.wikipedia.org/wiki/Anexo:Lenguajes\\_de\\_programaci%C3%B3n](https://es.wikipedia.org/wiki/Anexo:Lenguajes_de_programaci%C3%B3n)

VER

Aunque este tema de los Lenguajes de Programación es bastante extenso, hemos llegado al final de este espacio de aprendizaje y tal como se expresó anteriormente, dentro de la carrera tendrán la línea de Algoritmia y Programación en la que profundizarán en muchos más aspectos relacionados, teniendo la oportunidad de llevar la programación a la práctica.

# Actividades



Apreciado estudiante, para entregar las actividades, por favor, diríjase a la pestaña evaluaciones, ubicada en la parte superior derecha.

Presentación

Unidad 1

Unidad 2

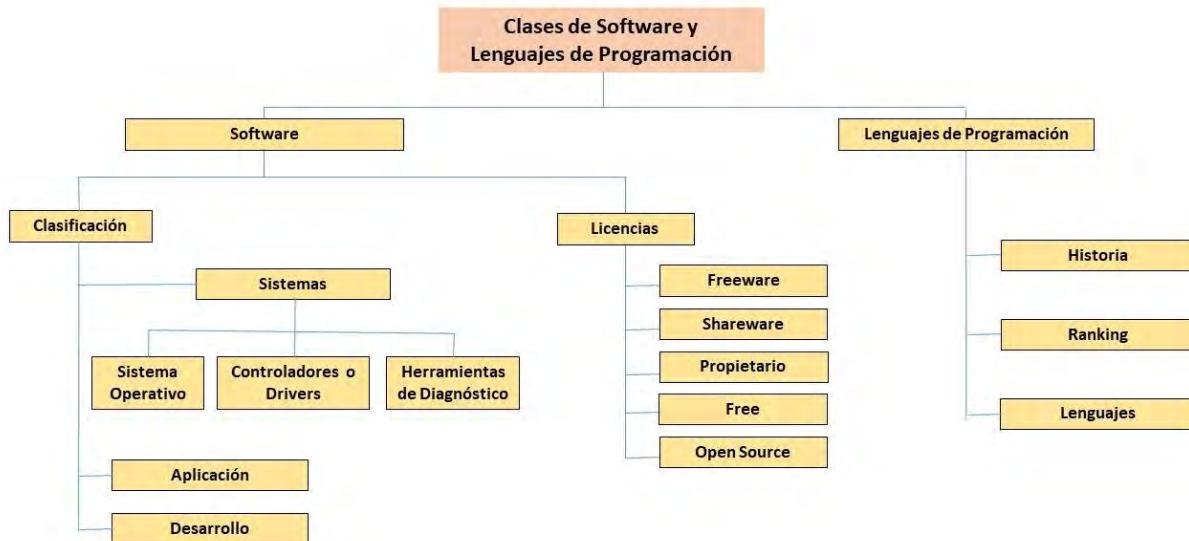
Unidad 3

Unidad 4

**Evaluaciones**

Servicios

# Resumen de la Temática



Estimado estudiante. Si está observando este mensaje, es porque acaba de finalizar el recorrido por las lecturas y los recursos de esta sección. Por tanto, para salir de aquí, y continuar con el desarrollo del curso, vaya a la parte superior y dé clic en:

[Salir de la actividad](#)





- **COMPILADOR:** traduce un programa completo, escrito en un lenguaje de programación, a código de máquina.
- **DEPURADOR (DEBUGGER):** es un programa que se usa para buscar y corregir errores en la programación.
- **EDITOR DE TEXTO:** permite la edición del código fuente de los programas.
- **IDE - INTEGRATED DEVELOPMENT ENVIRONMENT O ENTORNO DE DESARROLLO INTEGRADO:** es un software que agrupa más de una de las categorías anteriores y de esta manera facilita el trabajo de los programadores. Los más básicos proporcionan un editor de código fuente, una herramienta asistente de construcción de código y un depurador.
- **INTÉRPRETE:** es similar a un compilador, es decir, traduce un programa a código de máquina, pero solo lo hace a medida que se van ejecutando las instrucciones.



# Referencias

---

422737. (2014). Software Programación Programa [Archivo de imagen]. Recuperado el 2020, 04, 25 en: <https://pixabay.com/es/illustrations/software-programaci%C3%B3n-programa-486705/>
- Bookdragon. (2019). Desarrollo Web Php Java [Archivo de imagen]. Recuperado el 2020, 04, 25 en: <https://pixabay.com/es/illustrations/desarrollo-web-php-java-c%C3%B3digo-4202909/>
- Clker-Free-Vector-Images. (2012). Escáner Flatbed De Hardware [Archivo de imagen]. Recuperado el 2020, 04, 25 en: <https://pixabay.com/es/vectors/esc%C3%A1ner-flatbed-de-hardware-23359/>
- Clker-Free-Vector-Images. (2012). Paquete Dvd Cuadro [Archivo de imagen]. Recuperado el 2020, 04, 25 en: <https://pixabay.com/es/vectors/paquete-dvd-cuadro-cart%C3%B3n-36037/>
- Clker-Free-Vector-Images. (2012). Pingüino Ave Animales La vida [Archivo de imagen]. Recuperado el 2020, 04, 25 en: <https://pixabay.com/es/vectors/ping%C3%BCino-ave-animales-42936/>
- Clker-Free-Vector-Images. (2012). Software De Cd Caja [Archivo de imagen]. Recuperado el 2020, 04, 25 en: <https://pixabay.com/es/vectors/logotipo-microsoft-ventanas-27046/>
- Clker-Free-Vector-Images. (2014). Webcam Video Cámara [Archivo de imagen]. Recuperado el 2020, 04, 25 en: <https://pixabay.com/es/vectors/webcam-video-c%C3%A1mara-pel%C3%A1cula-295440/>
- Elchinator. (2019). Source Código Software [Archivo de imagen]. Recuperado el 2020, 04, 25 en: <https://pixabay.com/es/photos/source-c%C3%B3digo-software-equipo-4280758/>
- Geralt. (2014). Software Cd Dvd [Archivo de imagen]. Recuperado el 2020, 04, 25 en: <https://pixabay.com/es/illustrations/software-cd-dvd-digital-disco-417880/>
- Geralt. (2015). Explorador Web Www [Archivo de imagen]. Recuperado el 2020, 04, 25 en: <https://pixabay.com/es/illustrations/explorador-web-www-equipo-773215/>
- Gutiérrez, J., Herrera, O., Pulgarín, R. (2018). Introducción a la Programación en C. Armenia, Colombia. Editorial Elizcom.
- Janjf93. (2017). Plana Diseño símbolo [Archivo de imagen]. Recuperado el 2020, 04, 25 en: <https://pixabay.com/es/vectors/plana-dise%C3%B1o-s%C3%ADmbolo-icono-www-2126883/>
- Kritsadaj. (2015). Atx2 Microcontrolador Arduino Caja [Archivo de imagen]. Recuperado el 2020, 04, 25 en: <https://pixabay.com/es/illustrations/atx2-microcontrolador-arduino-950511/>
- Lasso, I. (2015, 04, 15). Qué es el código fuente [Archivo en imagen]. Disponible en: <http://www.proyectoautodidacta.com/comics/que-es-el-codigo-fuente/>
- Mmi9. (2016). Pantalla Monitor Equipo [Archivo de imagen]. Recuperado el 2020, 04, 25 en: <https://pixabay.com/es/vectors/pantalla-monitor-equipo-pc-1315650/>



OpenClipart-Vectors. (2013). Cuadro Software De 3D [Archivo de imagen]. Recuperado el 2020, 04, 25 en: <https://pixabay.com/es/vectors/cuadro-software-de-3d-azul-150962/>

OpenClipart-Vectors. (2013). Cuadro Cartón 3D Software [Archivo de imagen]. Recuperado el 2020, 04, 25 en: <https://pixabay.com/es/vectors/cuadro-cart%C3%B3n-3d-software-de-149481/>

OpenClipart-Vectors. (2013). Del Ratón de La Computadora [Archivo de imagen]. Recuperado el 2020, 04, 25 en: <https://pixabay.com/es/vectors/del-rat%C3%B3n-rat%C3%B3n-de-la-computadora-160592/>

OpenClipart-Vectors. (2013). Gráfica Comprimido Dibujo [Archivo de imagen]. Recuperado el 2020, 04, 25 en: <https://pixabay.com/es/vectors/gr%C3%A1fica-comprimido-dibujo-155176/>

OpenClipart-Vectors. (2013). Motor Ingeniería Artes [Archivo de imagen]. Recuperado el 2020, 04, 25 en: <https://pixabay.com/es/vectors/motor-ingenier%C3%ADa-artes-m%C3%A1quina-153649/>

OpenClipart-Vectors. (2013). Usb Ubs Stick Equipo [Archivo de imagen]. Recuperado el 2020, 04, 25 en: <https://pixabay.com/es/vectors/usb-ubs-stick-equipo-disco-memoria-157656/>

OpenIcons. (2013). Teclado Entrada Dispositivo De [Archivo de imagen]. Recuperado el 2020, 04, 25 en: <https://pixabay.com/es/vectors/teclado-entrada-98410/>

PaliGraficas. (2017). Dvd Cd Música Escribir [Archivo de imagen]. Recuperado el 2020, 04, 25 en: <https://pixabay.com/es/vectors/dvd-cd-m%C3%BAtica-escribir-cd-cubierta-2831541/>

Peggy\_Marco. (2019). Equipo Internet Wifi [Archivo de imagen]. Recuperado el 2020, 04, 25 en: <https://pixabay.com/es/illustrations/equipo-internet-wifi-nube-4674870/>

Pixaline. (2016). Medios De Comunicación Social [Archivo de imagen]. Recuperado el 2020, 04, 25 en: <https://pixabay.com/es/vectors/medios-de-comunicaci%C3%BDn-social-1177293/>

Pixaline. (2016). Oficina Windows Palabra [Archivo de imagen]. Recuperado el 2020, 04, 25 en: <https://pixabay.com/es/illustrations/oficina-windows-palabra-sobresalir-1356793/>

Portalsenior. [Universidad Politécnica de Valencia]. (2010). Los sistemas operativos ¿Qué son? [Archivo en video]. Disponible en: [https://youtu.be/Cli\\_PMTUwkE](https://youtu.be/Cli_PMTUwkE)

Pressman, R. (2010). Ingeniería del software – Un enfoque práctico. Ciudad de México, México. Mc Graw Hill.

Rosiette-Stock. (2017). Software Tecnología Internet [Archivo de imagen]. Recuperado el 2020, 04, 25 en: <https://pixabay.com/es/illustrations/software-tecnolog%C3%ADa-internet-2646757/>

Stallman, R. (2008). Por qué el «código abierto» pierde de vista lo esencial del software libre [Sitio Web]. Disponible en: <https://www.gnu.org/philosophy/open-source-misses-the-point.es.html>

TheDigitalArtist. (2014). En Blanco Caja De Cuadro Cartón [Archivo de imagen]. Recuperado el 2020, 04, 25 en: <https://pixabay.com/es/illustrations/en-blanco-caja-de-software-cart%C3%B3n-338176/>

TheDigitalArtist. (2015). Caja De Software Cuadro Cartón [Archivo de imagen]. Recuperado el 2020, 04, 25 en: <https://pixabay.com/es/illustrations/caja-de-software-cuadro-cart%C3%B3n-1076811/>

TheDigitalArtist. (2016). Software De Cd Caja [Archivo de imagen]. Recuperado el 2020, 04, 25 en: <https://pixabay.com/es/illustrations/software-de-cd-caja-de-software-1566455/>

Vasconcelos Santillán, J. (2018). Introducción a la computación. Ciudad de México, México. Grupo Editorial Patria.

Vercelli, A. ArielVercilli.org. (2009). Repesando los bienes intelectuales comunes [Archivo en línea]. Disponible en: <http://www.arielvercelli.org/rlbic.pdf>

**Bibliografía:**

Gutiérrez, J., Herrera, O., Pulgarín, R. (2018). Introducción a la Programación en C. Armenia, Colombia. Editorial Elizcom.

Vasconcelos S, J. (2018). Introducción a la computación. Ciudad de México, México. Grupo Editorial Patria.



UNIVERSIDAD  
DEL QUINDÍO



ESTRATEGIA VIRTUAL

## UNIDAD DE VIRTUALIZACIÓN

unidaddevirtualizacion@uniquindio.edu.co

Tel: (57) 6 7 35 9300 Ext 400

Universidad del Quindío

Carrera 15 Calle 12 Norte

Bloque de Ciencias Básicas - Primer Piso

Armenia, Quindío - Colombia