

## **Importancia de la disciplina Lenguajes y Técnicas de Programación en la carrera**

### **Educación Informática**

#### **Importance of the programming languages and techniques discipline in the computer**

#### **science education program**

*Alberto Betancourt Almaguer<sup>1</sup>*

*Yilian Peña Rodríguez<sup>2</sup>*

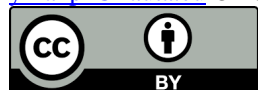
### **Resumen**

En las Ciencias Informáticas si hubiese que elegir una disciplina técnica como la más importante e influyente de este contexto (aunque todas son importantes y están interrelacionadas), existe amplia probabilidad de que la mayoría de expertos en el tema seleccionaría a la programación, esta afirmación puede argumentarse con varios elementos. El presente artículo tiene por objetivo fundamentar la importancia de la cultura de la programación en la carrera Educación Informática del Centro Universitario Municipal (CUM) Jobabo, argumentando el roll protagónico que juega la disciplina Lenguajes y Técnicas de Programación en la formación de estos profesionales. El trabajo de investigación está estructurado en dos secciones principales, la primera, dedicada a la sistematización de los fundamentos teóricos relacionados con el tema, y la segunda, a realizar un análisis crítico de algunos aspectos de la disciplina antes mencionada y del modelo del profesional, a la vez que se ofrece un criterio sobre las experiencias prácticas acumuladas hasta la fecha y se proponen algunas tecnologías, las cuales se creen sean las más adecuadas para el desarrollo exitoso de la cultura de esta gran área del conocimiento científico-técnico que es indispensable para el desarrollo de las Nuevas Tecnología de la Informática y las

---

<sup>1</sup> Ingeniero Industrial. Máster en Dirección. Centro Universitario Municipal Jobabo, Las Tunas, Cuba. Email: [albertoba@ult.edu.cu](mailto:albertoba@ult.edu.cu). ORCID: <https://orcid.org/0000-0003-0943-0085>

<sup>2</sup> Ingeniera en Ciencias Informáticas. Centro Universitario Municipal Jobabo, Las Tunas, Cuba. Email: [yilianpr@nauta.cu](mailto:yilianpr@nauta.cu) ORCID: <https://orcid.org/0000-0002-7463-2145>



Comunicaciones. En el proceso investigativo se han diseñado y aplicado instrumentos del nivel empírico (guías de observación al proceso docente, encuestas a estudiantes y entrevistas especialistas) en aras de sustentar el carácter científico de este trabajo desde los hechos (lo factual), contrastando las teorías e hipótesis que existen sobre este campo.

*Palabras clave:* programación, software, informática, pensamiento lógico

### **Abstract**

In Computer Science, if one had to choose a technical discipline as the most important and influential in this context (although all are important and interrelated), there is a high probability that most experts in the field would select programming, this statement can be argued with several elements. The purpose of this article is to support the importance of programming culture in the Computer Education career of the Jobabo Municipal University Center, arguing the leading role played by the Programming Languages and Techniques discipline in the training of these professionals. The research work is structured in two main sections, the first one, dedicated to the systematization of the theoretical foundations related to the subject, and the second one, to carry out a critical analysis of some aspects of the above mentioned discipline and of the professional's model, while offering a criterion on the practical experiences accumulated so far and proposing some technologies, which are believed to be the most adequate for the successful development of the culture of this great area of scientific-technical knowledge that is indispensable for the development of the New Technologies of Informatics and Communications. In the research process, empirical instruments have been designed and applied (observation guides to the teaching process, student surveys and interviews to specialists) in order to support the scientific character of this work from the facts, contrasting the theories and hypotheses that exist in this field.

*Keywords:* programming, software, computer science, logical thinking

### **Introducción**

En el mundo de la informática existen diversas disciplinas científico-técnicas que se articulan para dar soporte a las nuevas tecnologías y hacer posible la gran variedad de tareas que en la actualidad desarrollan las computadoras, es impensable referirse al desarrollo que ha alcanzado la civilización humana en todas sus dimensiones (científica, tecnológica, económica, social y cultural) sin la intervención de estos artefactos. Algunas de las materias más influyentes en este campo son las matemáticas, la electrónica, las bases de datos, la inteligencia artificial y la ingeniería de software por citar algunas, pero cuando se habla de informática o sistema informático la mayoría de las personas no especializadas entienden que se refiere al software y al hardware, y en parte tienen razón, entre el software y el hardware se da una relación tan estrecha similar quizás a la presente entre la mente y el cerebro si es válida esta analogía.

El hardware puede existir sin el software, pero no a la inversa, sin embargo, la importancia del software radica esencialmente en que el hardware sin este último, sería solo chatarra inservible, por muy sofisticado que este fuese, no podría desempeñar ninguna función sin órdenes, instrucciones, algoritmos y datos que permitan resolver problemas y realizar acciones útiles. El software puede ser entendido como la parte intangible de la informática, como los programas y datos que se escriben en un lenguaje comprensible para las Unidades Centrales de Procesamiento (CPU) de las computadoras.

Las carreras universitarias afines a la informática o que se especializan en ella estudiadas en nuestro país como es el caso particular de la Licenciatura en Educación: Especialidad Informática, aunque estudian el hardware en asignaturas como arquitectura de computadoras y redes, es solo de manera superficial, las disciplinas técnicas están enfocadas fundamentalmente

## IMPORTANCIA DE LA DISCIPLINA LENGUAJES Y TÉCNICAS DE PROGRAMACIÓN

en el software, lo cual es bastante razonable ya que el hardware es más cercano a especialidades relacionadas con la electrónica.

Respecto a esto cabe destacar que, con toda certeza, la disciplina más influyente e imprescindible en el desarrollo del software, es la programación. Sin pretender dar una definición rigurosa de la programación, a groso modo, esta puede entenderse como el conjunto de reglas sintácticas y semánticas que permiten escribir instrucciones en un lenguaje específico que luego será compilado o interpretado para ser ejecutado de manera eficaz por el ordenador, básicamente de esta forma es que se construyen las aplicaciones informáticas que hacen posible la existencia del software.

La especialidad de Educación Informática en su diseño curricular no está pensada para el desarrollo de software empresarial que puede llegar a ser de gran complejidad, pero sí que se enfatiza mucho en la creación del software educativo, en este sentido existen multitud de herramientas de autor como es el caso de Matchware Mediator, líder en el mercado de este tipo de productos, que presentan la característica de exigir pocos e incluso nulos conocimientos de programación, pero la práctica demuestra de manera fehaciente que cuando se desea realizar un trabajo de características muy específicas y al menos mediana complejidad técnica, se requiere necesariamente la programación.

En cambio, la Ingeniería informática es una especialidad estrictamente tecnológica y presenta otras exigencias y particularidades en su formación, a saber, según Maridueña (2014) “se conceptualiza en una carrera universitaria enfocada al dominio de las ciencias y tecnologías de la información, necesarias para planificar, analizar, diseñar, programar, operar, mantener, evaluar y optimizar sistemas informáticos de diversa índole con énfasis en la actividad humana” (p. 5).

El objetivo que persigue el presente trabajo de investigación es fundamentar la importancia de la programación en la formación del profesional de la carrera Licenciatura en Educación: Especialidad Informática, como disciplina rectora de la parte técnica de la carrera, realizando un análisis crítico de algunos aspectos del programa de la disciplina Lenguajes y Técnicas de Programación y del Modelo del Profesional, abordando los paradigmas y lenguajes más populares y recomendables en la actualidad, en aras de hacer propuestas plausibles que contribuyan al perfeccionamiento del proceso de enseñanza-aprendizaje de esta materia fundamental para la carrera en cuestión.

### **Desarrollo**

#### **Consideraciones teóricas sobre la importancia de la programación y su enseñanza en las especialidades afines a la informática**

Es importante aclarar que la programación en sentido general se suele dividir en dos grandes grupos o categorías, a los que también se le llama paradigmas, modelos o formas de programar, por un lado, se encuentra la programación imperativa que actualmente es la que más se utiliza en la práctica y se enseña en las universidades, por el otro, y ganando en popularidad debido al auge de la inteligencia artificial, nos encontramos con la programación declarativa. En el primer caso, el programador se centra en escribir los pasos o algoritmo que resuelven el problema objeto de análisis, en cambio, en el segundo paradigma general, los programadores dan más importancia a describir el problema en cuestión y relegan el método de solución a la propia máquina.

La disciplina Lenguajes y Técnicas de Programación de la carrera de Educación Informática está orientada en su totalidad hacia la programación imperativa, no abordando las técnicas funcionales y lógicas tratadas en lenguajes como Haskell y Prolog respectivamente en el

## IMPORTANCIA DE LA DISCIPLINA LENGUAJES Y TÉCNICAS DE PROGRAMACIÓN

contexto de la inteligencia artificial. Dentro de la programación imperativa existen varios subparadigmas, algunos de los más influyentes y populares hoy en día son: la programación estructurada, la programación orientada a objetos (imperante en la actualidad) y la programación conducida/dirigida por eventos (también llamada programación visual), precisamente en estos paradigmas se encuentra enmarcada la disciplina que se analiza.

La evidencia estadística demuestra que a la mayoría de las personas les resulta complejo aprender los fundamentos básicos de la programación, esto se debe fundamentalmente al pensamiento abstracto y lógico que hay que desarrollar para la escritura de los algoritmos, una cosa es saber o comprender como hay que proceder para resolver determinado problema y otra bien distinta es decirle a la máquina computadora cómo debe obrar para alcanzar el mismo fin. Sin embargo, una vez asimilados los conocimientos y habilidades esenciales de la programación, tales como el trabajo con variables, tipos de datos, arreglos y estructuras de control del flujo cíclicas y condicionales, es más expedito comprender conceptos como clases, objetos, herencia y polimorfismo.

La técnica de la programación estructurada es la más simple y básica estudiada en la carrera Educación Informática, este paradigma constituye la base indiscutible del resto de técnicas que se abordan en la disciplina, técnica utilizada ampliamente en los años 60 y 70 en la denominada “Crisis del software”, en relación a esto, autores como Ferreira y Rojo (2006) plantean que:

Han pasado más de 35 años y sin embargo la programación estructurada sigue teniendo enorme importancia en el campo de la enseñanza. Aunque ha habido muchos cambios y avances desde ese entonces, por ejemplo la aparición y consolidación de la programación

orientada a objetos, coexisten varios enfoques y tendencias pero sin consenso sobre cuál es la mejor manera de enseñar los conceptos básicos de la programación. (p. 1)

Estos autores señalan que un punto de inflexión lo marca la aparición del lenguaje PASCAL, destinado a la enseñanza de la técnica estructurada que se convirtió en un estándar de facto en el mundo de la programación. Un aspecto esencial en este sentido lo constituyen los métodos y formas de enseñanza de la programación en los cursos introductorios de esta materia, en aras de crear las bases necesarias para poder incursionar en otras temáticas más complejas y avanzadas.

Moroni y Señas (2005) son del criterio que la complejidad de los programas que se desarrollan actualmente produce la necesidad de iniciar a los alumnos en un camino que los conduzca a utilizar efectivas técnicas de programación, señalando que es importante para ello poner énfasis en el diseño previo, donde una estrategia validera es comenzar a enseñar programación utilizando los algoritmos como recursos esquemáticos para plasmar el modelo de la resolución de un problema.

En tanto, Pears (2007, como se citó en García, 2017) señala que “uno de los elementos clave en la introducción a la programación es el lenguaje escogido; siendo los otros elementos: el currículo, la pedagogía y las herramientas empleadas en el proceso de enseñanza-aprendizaje” (p. 4). Esta cuestión es ampliamente respaldada por la mayoría de los autores aún más tratándose de una especialidad de corte pedagógico.

En correspondencia con el autor citado anteriormente, Ortiz (2010) hace alusión a que:

En tiempos recientes, el primer curso de programación, como se enseña en muchas instituciones de educación superior en México y el mundo, ha tomado un fuerte énfasis hacia la enseñanza de lenguajes de programación como C, C++, Java o C#. Dichos

lenguajes son relevantes para la construcción de software de producción, pero no fueron diseñados para enseñar a programar. Lo que ha provocado que muchos alumnos tengan la percepción de que la computación es una disciplina árida y con un alto nivel de dificultad técnica. Quizás esto no sea un gran problema cuando se está en una época de bonanza en cuanto a captación, pero cuando la matrícula se contrae esta percepción negativa puede tener un efecto devastador que no puede ser ignorado. (p. 1)

Esto es importante tenerlo en cuenta, ya que cuando se analiza la bibliografía básica de la disciplina Lenguajes y Técnicas de Programación, puede apreciarse que la primera literatura que aparece, hace referencia al lenguaje C++, el cual es válido destacar que es muy utilizado hoy en día a pesar de su antigüedad, entre otras cosas, por su potencia y fiabilidad esencialmente en la construcción de software de sistema, y además de ello, muchos otros lenguajes de gran popularidad y éxito se derivan directamente de este, tal es el caso de PHP, Java y C#, esto permite con toda claridad realizar varias interpretaciones favorables al lenguaje C++. Sin embargo, también es cierto que desde el punto de vista sintáctico y semántico es un lenguaje relativamente engorroso si lo comparamos con otros, tales como, Python o Javascript.

Por tal razón, es que el autor citado con anterioridad considera que Python es una tecnología más recomendable para cursos introductorios de programación, donde básicamente en lo que se enfatiza es en el pensamiento lógico y la algoritmia. A decir de este autor, Python es un lenguaje idóneo porque presenta las siguientes características:

- Es orientado a objetos, pero soporta también los estilos de programación procedural y funcional.
- Corre en múltiples plataformas, incluyendo Windows, Mac OS y Linux.
- Su sintaxis y semántica es sencilla y consistente. Utiliza tipos dinámicos.



## IMPORTANCIA DE LA DISCIPLINA LENGUAJES Y TÉCNICAS DE PROGRAMACIÓN

- Es adecuado tanto para programar scripts como aplicaciones de gran tamaño.
- Es muy modular.
- Cuenta con administración automática de memoria a través de recolección de basura.
- Incluye una poderosa y extensa biblioteca de clases.
- Cuenta con una gran comunidad que se dedica a promover su desarrollo y adopción.
- Por su naturaleza interactiva, resulta ideal para llevar a cabo programación experimental y desarrollo rápido.

En el caso particular de la carrera objeto de análisis, es importante significar que como se está hablando de futuros profesionales de la educación, hay que llevar en equilibrio y sinergia tanto los conocimientos técnicos de la disciplina de la programación, como los relacionados con la pedagogía y la didáctica de la informática en general, Díaz et al. (2018), argumentan al respecto:

El objetivo principal de la enseñanza de la programación en la formación inicial es resolver problemas, con el empleo de un lenguaje de programación y se necesita para su cumplimiento, que el estudiante desarrolle el sistema de habilidades específicas de la programación y se apropie del sistema de conocimientos y procedimientos en correspondencia con las técnicas de programación y las herramientas de desarrollo en estudio. Sin embargo, el modelo de profesional de la especialidad demanda, además, que el profesor en formación sea capaz de asimilar modos de actuación que le faciliten su futuro desempeño en la escuela. (p. 3)

Aun así, la programación de computadoras es una tarea compleja, Jiménez et al. (2019) reafirma esta idea, al expresar que:

Programar computadores requiere mucha dedicación, lo cual se refleja en el tiempo invertido para esta actividad; además, en este proceso tanto el estudio de temáticas como el desarrollo de ejercicios puntuales no garantizan su efectivo aprendizaje. En consecuencia, es necesario incorporar este nuevo modelo mental, que exige una constante actualización de acuerdo con las tendencias experimentadas en los recientes enfoques emergentes en este campo. (p. 3)

El nuevo modelo mental al que se refiere la cita anterior, hace referencia al conjunto de reglas sintácticas y semánticas que representan a un lenguaje de programación, a partir del cual se le pueden suministrar órdenes, comandos e instrucciones al ordenador, favoreciendo de esta manera la manipulación y transformación de datos, orientado a dar solución a diversos problemas que demanda el desarrollo de la humanidad.

Retomando el tema de la elección del lenguaje de programación, cuestión sin lugar a dudas a tener en cuenta, otro candidato nada despreciable es Java, un lenguaje cuya sintaxis deriva en gran parte de C++ (siendo más simple y claro sintácticamente) y a la vez ha influido significativamente en el lenguaje líder de la tecnología .NET de Microsoft Corporation, nos referimos a C#.





En palabras de Machado (2020), “Java es un lenguaje con una de las mayores comunidades de desarrolladores y mayor aplicación en proyectos de las grandes empresas tecnológicas” (p. 6). Este autor afirma que este lenguaje es un fuerte competido de Python, y que este último solo lleva una ligera ventaja en las técnicas de inteligencia artificial, específicamente en el Big Data y el Machine Learning, pero que en términos prácticos sería difícil decidir cuál es el mejor de ambos.

## IMPORTANCIA DE LA DISCIPLINA LENGUAJES Y TÉCNICAS DE PROGRAMACIÓN

Python es un lenguaje muy recurrente en los últimos cinco a ocho años, ha ganado mucho en popularidad, el índice Tiobe con url <https://www.tiobe.com>, es uno de los rankings más fiables sobre los lenguajes de programación más utilizados en la actualidad, el mismo presenta el siguiente gráfico (Figura 1).

**Figura 1**

*Popularidad de los lenguajes de programación (cuatro primeros lugares)*

Sep 2022	Sep 2021	Change	Programming Language
1	2	▲	 Python
2	1	▼	 C
3	3		 Java
4	4		 C++

Lo cual es evidencia suficiente para elegir una buena alternativa en la impartición de docencia. En correspondencia con lo anteriormente planteado, Soto (2021) propone incluir Python como lenguaje complementario en la Universidad de Ciencias Informáticas en todos los ámbitos, debido a las características favorables y deseadas que presenta este lenguaje en términos multidisciplinar (para el procesamiento matemático, para el scripting en la automatización de tareas repetitivas y para el desarrollo de aplicaciones Webs del lado del servidor).

### **Análisis de los aspectos esenciales de la disciplina y propuesta de lenguajes para su implementación**

La disciplina Lenguajes y Técnicas de Programación (LTP) de la carrera Licenciatura en Educación: Especialidad Informática, está integrada por cuatro asignaturas, abarcando desde el segundo periodo del tercer año hasta el primer periodo del quinto año, en estos cursos se aborda la programación en sentido general y básica, comenzando por la algoritmia y la lógica de

## IMPORTANCIA DE LA DISCIPLINA LENGUAJES Y TÉCNICAS DE PROGRAMACIÓN

programación representada por el cálculo proposicional, tablas de la verdad, diagramas de flujo y pseudocódigo, pasando por la programación estructurada con el uso de variables y sus tipos de datos, estructuras condicionales e iterativas para el control del flujo del programa y el uso de arreglos unidimensionales, hasta llegar a la programación orientada a objetos (paradigma imperante en la actualidad), donde se tratan temas como la encapsulación, la herencia y el polimorfismo, para terminar con la programación conducida por eventos o programación visual, la cual está estrechamente relacionada con la programación orientada a objetos.

En la disciplina se trabaja con lenguajes de propósito general, creando aplicaciones sencillas para ejecutar en modo consola o en modo de interfaz gráfica de usuario bajo entornos Windows o Linux. Algunos de estos lenguajes son C++, Java, Basic, Pascal, Delphi o Python, pero de ninguna manera se orienta el uso de estos lenguajes hacia el desarrollo Web (en el caso de los lenguajes que procedan para tal fin como es el caso de Java y Python), ya que esta es un área de la programación muy particular y especializada.

El desarrollo Web tiene otras características, habría que referirse al backend (lado del servidor) y frontend (lado del cliente o navegador) y hacer uso de lenguajes como PHP y Javascript respectivamente, lenguajes que no se incluyen en la bibliografía de la disciplina. Además de ello, en la actualidad es prácticamente imposible hablar de desarrollo web sin la intervención de frameworks (marcos de trabajo que ofrecen un proyecto pre-elaborado y el patrón de diseño modelo-vista-controlador en la mayoría de los casos), en tal sentido tenemos a Laravel para PHP y AngularJS para Javascript, temas tampoco abordados en la carrera.

Un aspecto de gran importancia a tener en cuenta si se desea cumplir satisfactoriamente con los objetivos de la disciplina, lo constituye ineludiblemente la elección del lenguaje introductorio con el cual se codificarán los primeros algoritmos y resolverán los problemas más

## IMPORTANCIA DE LA DISCIPLINA LENGUAJES Y TÉCNICAS DE PROGRAMACIÓN

básicos. Esta decisión le corresponde al colectivo de la disciplina, el cual radica en la Sede Central, donde, tradicionalmente lo que se ha estilado hasta el momento es elegir un único lenguaje que transcurra en todas las asignaturas, y generalmente este lenguaje es C++. Esta opción tiene ventajas y desventajas, analicemos detalladamente (Tabla 1):

**Tabla 1**

*Tabla resumen de las ventajas y desventajas de utilizar únicamente el lenguaje C++ en toda la disciplina*

<b>Ventajas</b>	<b>Desventajas</b>
Solo hay que aprender una única sintaxis.	No es recomendable aprender a programar en un solo lenguaje, ya que limita las capacidades y posibilidades del programador.
Es un lenguaje multiparadigma, facilitando el cambio entre paradigmas.	La sintaxis del lenguaje C++ es una de las más complejas.
Una vez asimilada esta sintaxis es simple aprender otros lenguajes derivados de C++ tales como PHP, Java y C#.	Se hace uso directamente de punteros por parte del programador, este aspecto es eliminado en lenguajes más modernos como Java, Python y C#.
C++ es un lenguaje muy potente para aplicaciones de escritorio.	En C++ existe la herencia múltiple y esto es criticado por la mayoría de programadores ya que tiende a generar problemas y confusiones.
C++ es un lenguaje más influyente y actual que otros como Basic y Pascal.	Aunque es un lenguaje con plena vigencia, es relativamente antiguo, ya que surge en los años ochenta.
Existen IDEs tales como el Embarcadero C++ Builder que integran la VCL (biblioteca de componentes visuales) que optimizar y hacen que el desarrollo de las interfaces gráficas de usuario sea fácil y rápido.	Es un lenguaje que su principal uso está enmarcado en la construcción de aplicaciones de sistema y de escritorio.

Si bien la carrera de Educación Informática no pretende formar especialistas en desarrollo de software de gran complejidad, o sea, no está pensada para crear programadores expertos, sí que es necesario fomentar una cultura de la programación lo más amplia y adecuada posible, que le permita al futuro egresado desarrollar un pensamiento lógico avanzado y amplia capacidad para analizar y resolver problemas. Además de ello, en la elaboración de software educativo, cuestión bastante exigida a estos profesionales, les será de gran utilidad contar con los conocimientos técnicos y experiencias relacionadas con la disciplina en cuestión. La enseñanza de la programación no puede ser entendida como mero formalismo, para tener un mayor entendimiento de los sistemas de software es necesario conocer cómo se codifican los algoritmos

en un lenguaje real, lo cual hace de los profesionales de esta carrera, un verdadero especialista en informática.

La práctica y experiencias acumuladas a lo largo de cuatro años de existencia de la carrera en el CUM Jobabo en la modalidad de curso encuentro, muestran que a la gran mayoría de estudiantes les resulta trabajoso desarrollar el pensamiento abstracto, la algoritmia y la modelación que exigen las soluciones informáticas de los problemas que se tratan en la disciplina. En las condiciones actuales de nuestro País y del territorio en particular, no se dispone de abundantes recursos tecnológicos (lo cual limita la consulta bibliográfica en Internet) y mucho menos de bibliografía impresa actualizada, en tales circunstancias hay que acudir a recursos metodológicos efectivos desde el punto de vista didáctico y a la experiencia y profesionalidad de los docentes, articulando lo anterior con una adecuada estrategia de selección de tecnologías.

La propuesta que se hace en aras de contribuir al fortalecimiento del proceso de enseñanza-aprendizaje de la programación en la carrera Educación Informática del CUM Jobabo, no desde un enfoque didáctico, sino más bien tecnológico, es utilizar al menos dos lenguajes, uno principal y otro complementario, donde se recomienda que ambos tengan una sintaxis marcadamente distinta pero simple, para que se note el contraste en la escritura, pero a la vez, se vea lo singular del algoritmo y los pasos lógicos. Además de esto, los lenguajes seleccionados deben ser actuales, de gran popularidad y con la mayor versatilidad posible.

Teniendo en cuenta lo anteriormente descrito, se es de la opinión que, dentro de un abanico amplio de posibilidades, dos candidatos nada despreciables son Java como lenguaje principal y Python como lenguaje complementario. El lenguaje principal debe ser utilizado en todos los ejercicios, en tanto, el complementario solo en aquellos seleccionados por el profesor para mostrar la diversidad de posibilidades que existe en el mundo de la programación.

### Conclusiones

La consulta y sistematización de la literatura especializada permitió apreciar los diversos criterios y experiencias que existen entre variados autores, aunque existe una tendencia hacia el consenso respecto a los métodos de enseñanza y la importancia de la programación, así como los lenguajes a utilizar.

La disciplina Lenguajes y Técnicas de Programación es de gran importancia para la carrera Educación Informática y la actuación del futuro profesional.

Aunque los contenidos de la programación son relativamente complejos, una adecuada conjugación de los lenguajes Java y Python contribuirá en el perfeccionamiento del proceso de enseñanza-aprendizaje de la programación.

### Referencias

- Díaz Tejera, K. I., Fierro Martín, E., & Muñoz Pentón, M. A. (2018). La enseñanza de la programación: una experiencia en la formación de profesores de informática. *Educación*, 27(53), 73-91.
- Ferreira Szpiniak, A. y Rojo, G. A. (2006). Enseñanza de la programación. *TE & ET: Revista Iberoamericana de Tecnología en Educación y Educación en Tecnología*, (1), 9-17.
- García Monsálvez, J. C. (2017). Python como primer lenguaje de programación textual en la Enseñanza Secundaria. *Education in the knowledge society (EKS)*, 18(2), 147-162.
- Jiménez-Toledo, J. A., Collazos, C., & Revelo-Sánchez, O. (2019). Consideraciones en los procesos de enseñanza-aprendizaje para un primer curso de programación de computadores: una revisión sistemática de la literatura. *TecnoLógicas*, 22(SPE), 82-116.

## IMPORTANCIA DE LA DISCIPLINA LENGUAJES Y TÉCNICAS DE PROGRAMACIÓN

Machado, A. M. (2020). *Por qué aprender Python y cuáles son sus ventajas.*

<https://www.pragma.com.co/academia/lecciones/descubre-por-que-aprender-python-y-cuales-son-sus-ventajas>

Maridueña, A. M. R. (2014). *Programación de computadoras: una habilidad desarrollada concéntrica y radialmente en el currículo del ingeniero en sistemas computacionales.*

<https://biblio.ecotec.edu.ec/revista/edicion5/PROGRAMACION%20DE%20COMPUTADORAS.pdf>

Moroni, N., & Señas, P. (2005). Estrategias para la enseñanza de la programación. In *I Jornadas de Educación en Informática y TICs en Argentina.*

<http://sedici.unlp.edu.ar/handle/10915/18901>

Ortiz Ramirez, A. (2010). Python como primer lenguaje de programación. *Publicación interna del Tecnológico de Monterrey, Campus Estado de México.*

[https://arielortiz.info/publicaciones/primer\\_lenguaje\\_30\\_jun\\_2010.pdf](https://arielortiz.info/publicaciones/primer_lenguaje_30_jun_2010.pdf)

Soto Gómez, E. (2021). Python en Ingeniería en Ciencias Informáticas. *Serie Científica de la Universidad de las Ciencias Informáticas, 14(12), 1-15.*