



## Script de visualización de grafos para Matemática Aplicada a Sistemas de Información en Salud

### Graph Visualization Script for Applied Mathematics in Health Information Systems

Alejandro Araujo Inastrilla <sup>1</sup> , Carlos Rafael Araujo Inastrilla <sup>1</sup> , María del Carmen Roche Madrigal <sup>1</sup> , Anet López Chacón <sup>1</sup> , Delia María Gálvez Medina <sup>2</sup> 

<sup>1</sup> Universidad de Ciencias Médicas de la Habana. Facultad de Tecnología de la Salud. La Habana, Cuba.

<sup>2</sup> Hospital Universitario "General Calixto García". La Habana, Cuba.

**\*Autor para la correspondencia:**  
[inastrilla2004@gmail.com](mailto:inastrilla2004@gmail.com)

**Recibido:** 18 de abril del 2024  
**Aceptado:** 19 de abril del 2024

#### Citar como:

Araujo-Inastrilla A, Araujo-Inastrilla CR, Roche-Madrigal MC, López-Chacón A, Gálvez-Medina DM. Script de visualización de grafos para Matemática Aplicada a Sistemas de Información en Salud. Rev. Cubana Tecnol. Salud. 2024; 15(1):e4267. <http://www.revtecnologia.sld.cu/index.php/tec/article/view/4267>

#### RESUMEN

**Introducción:** la teoría de grafos en la formación de los estudiantes del campo de los Sistemas de Información en Salud es crucial para desarrollar algoritmos y métodos para la toma de decisiones en sistemas complejos. Sin embargo, no se disponen de herramientas para visualizar los grafos durante el proceso de enseñanza-aprendizaje de este contenido en la asignatura de Matemática Aplicada. **Objetivo:** diseñar un script de la visualización de grafos para la asignatura de Matemática Aplicada en Sistemas de Información en Salud. **Método:** se realizó un estudio de innovación tecnológica, donde se elabora un código en Python de visualización de grafos, a fin de facilitar el aprendizaje de teoría de grafos en la asignatura de Matemática Aplicada. Se utilizó el software *Visual Studio Code* para programar y ejecutar el script. **Resultados:** se elaboró el pseudocódigo y el diagrama de flujo del algoritmo del código de visualización a partir de la abstracción y modelación de las bases teóricas de la teoría de grafos. Se obtuvo el código en lenguaje Python. A modo de prueba se generó un grafo de ocho nodos y 24 aristas que demostró la efectividad del código elaborado. **Conclusiones:** el script Python propuesto es un paso hacia la elaboración de un software de generación de grafos con mejores prestaciones, que facilite el proceso de aprendizaje de este tema mediante solución de problemas prácticos.

**Palabras clave:** Grafos, Matemática Aplicada, Python, Sistemas de Información en Salud

## ABSTRACT

*Introduction:* graph theory in the training of students in the field of Health Information Systems is crucial to develop algorithms and methods for decision making in complex systems. However, there are no tools available to visualize graphs during the teaching-learning process of this content in the Applied Mathematics course. *Objective:* to design a graph visualization script for the Applied Mathematics in Health Information Systems course. *Method:* a technological innovation study was carried out, where a Python code for graph visualization was elaborated, in order to facilitate the learning of graph theory in the Applied Mathematics course. Visual Studio Code software was used to program and execute the script. *Results:* the pseudo code and the flowchart of the visualization code algorithm were elaborated from the abstraction and modeling of the theoretical bases of graph theory. The code was obtained in Python language. As a test, a graph with eight nodes and 24 edges was generated, which demonstrated the effectiveness of the elaborated code. *Conclusions:* the proposed Python script is a step towards the development of a better performing graph generation software, which facilitates the learning process of this subject by solving practical problems.

**Keywords:** *Graphs; Applied Mathematics; Python; Health Information Systems*

## INTRODUCCIÓN

Para adaptarse a la sociedad de la información actual, el sector de la salud adopta una estrategia de informatización que requiere de recursos humanos capacitados para gestionar información y conocimiento a través de las Tecnologías de la Información y las Comunicaciones (TIC)<sup>1</sup>. Son necesarios los profesionales con alta alfabetización y fluidez de datos que sustenten la toma de decisiones basada en la evidencia <sup>2,3,4</sup>.

Ante este panorama, la formación de Licenciados en Sistemas de Información en Salud (SIS) es fundamental. El profesional de esta área gestiona la información en salud con competencias técnicas, docentes y científicas, mediante métodos y tecnologías avanzadas, dentro del enfoque social del Sistema de Salud cubano <sup>5,6</sup>.

La asignatura de Matemática Aplicada ofrece a los estudiantes de SIS las herramientas necesarias para el entendimiento de la informática. Según Roche-Madrugal <sup>7</sup>, autora del texto básico de la asignatura tiene un gran peso en el desarrollo de un estilo de pensamiento lógico y de habilidades para la planificación de las estructuras de acciones en la solución de tareas.

El estudio de la Matemática Aplicada incita en los estudiantes la capacidad de ser exactos y organizados en los cálculos, el trabajo con datos, y la gestión de información en sistemas computarizados. A partir de estos conocimientos se adquieren habilidades para construir modelos informáticos y realizar procedimientos tecnológicos desde el entendimiento de los elementos que componen los sistemas de procesamiento de información <sup>7</sup>.

En la distribución de la asignatura se encuentra un tema dedicado al estudio de la teoría de grafos. Los conocimientos en esta área facilitan el tratamiento de algoritmos y métodos de resolución eficaces para la toma de decisiones <sup>8,9</sup>. Los grafos establecen a partir de un proceso de abstracción, modelar relaciones; de ahí la utilidad del recurso en el análisis y la comprensión de asociaciones entre elementos de un conjunto.

Para estudiantes y profesionales de los SIS, resulta relevante aplicar la teoría de grafos en la solución de problemas concretos en el ámbito donde laboran. Los grafos están presentes en la representación de redes de computadoras, en los análisis cuantitativos, la planificación de proyectos o la presentación de relaciones de estructuras complejas. Por tanto, resulta importante dentro de los procedimientos tecnológicos a dilucidar en esta área.

Sin embargo, no se cuenta con las herramientas didácticas que permitan la visualización de los grafos más allá del trazado a mano, lo cual es complejo en caso de un elevado número de relaciones a representar. Además, las herramientas de visualización de grafos existentes a nivel mundial requieren de otras habilidades, condiciones y conocimientos técnicos que no se adquieren a través de la asignatura, por lo que no solucionan esta carencia.

Los autores consideran oportuno disponer de herramientas que coadyuven al desarrollo de habilidades en la utilización de grafos. Por tanto, el objetivo del presente artículo es diseñar un script de la visualización de grafos para la asignatura de Matemática Aplicada en SIS.

## MÉTODO

Se realizó un estudio de innovación tecnológica, mediante la generación de códigos con el fin de facilitar el aprendizaje de teoría de grafos en la asignatura de Matemática Aplicada. El artículo se centra en la concepción de un script, para articular un software a partir del mismo.

Se realizó un análisis del texto básico de la asignatura para comprender los aspectos conceptuales y teóricos de los grafos. Del análisis se establecieron las reglas de representación de relaciones. Luego de la comprensión de las reglas, mediante el método de la modelación, se programaron las operaciones lógicas necesarias para la visualización de grafos.

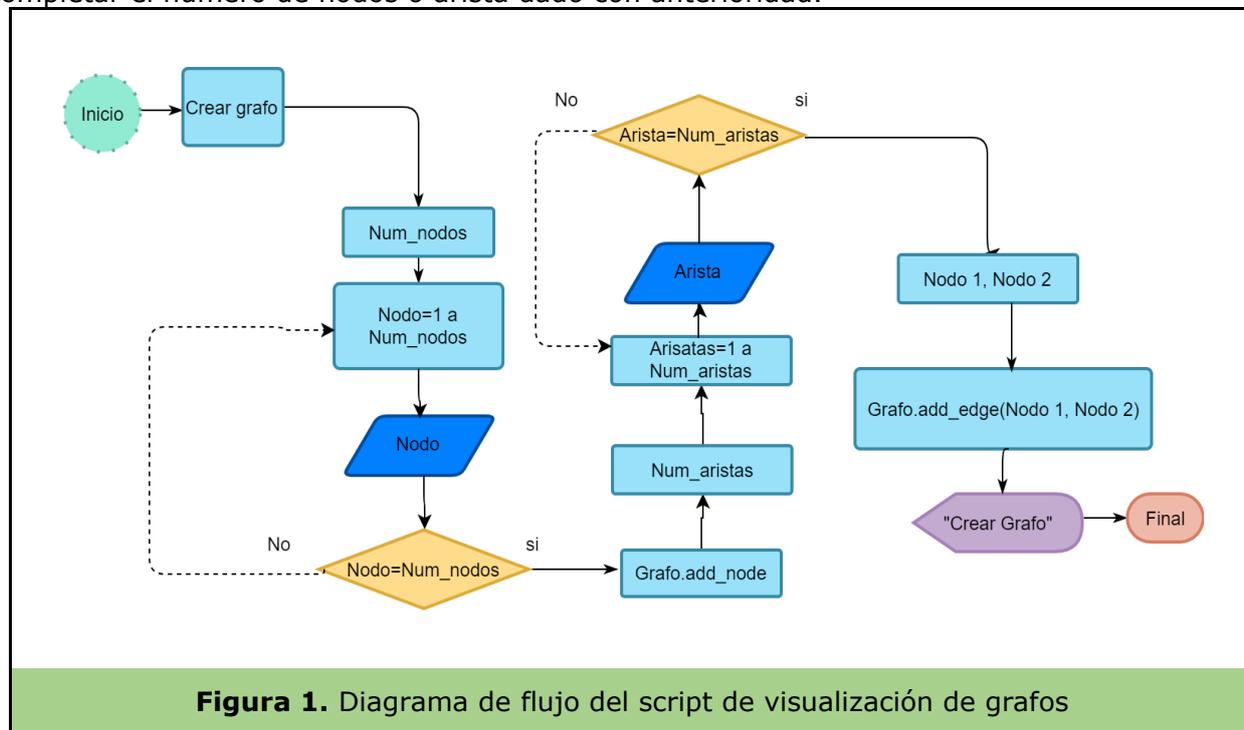
En la elaboración de este script se empleó el lenguaje de programación Python. Se utilizó el software *Visual Studio Code* al programar y ejecutar el script. En la programación fue imprescindible la librería de Python "*matplotlib*" y la extensión "*networkx*". Solo se emplearon operaciones y condiciones para la ejecución del proceso de visualización.

## RESULTADOS

A partir del análisis de los elementos que permiten la construcción de grafos, se elaboró el pseudocódigo del algoritmo del script. De esta manera se representó el funcionamiento del producto a desarrollar en un lenguaje similar al código a utilizar durante la programación. Se obtuvo el pseudocódigo que se presenta a continuación:

```
1 Inicio
2 Definir Crear_grafo
3 Crear grafo vacío
4 Entrar "Ingrese número de nodos" num_nodo
5 Para i en el rango num_nodo
    Entrar "Inserte nombre del nodo"
    Agregar nodo a grafo
6 Entrar "Ingrese número de aristas" num_aristas
7 Para i en el rango num_aristas
    Entrar "Ingrese nombre del primer nodo" Nodo 1
    Entrar "Ingrese nombre del segundo nodo" Nodo 2
    Agregar (Nodo 1, Nodo 2) a grafo
8 Grafo
9 Fin Crear_grafo
10 Mostrar: Grafo
11 Fin
```

Para un mejor entendimiento del proceso que debe ejecutar el script, también se modeló el algoritmo a través de un diagrama de flujo (Figura 2). En el mismo se establecen dos iteraciones: introducir los nodos y las aristas. Al declarar una cantidad de nodos o aristas, el algoritmo repite la operación de introducir el nombre de cada uno de estos elementos hasta completar el número de nodos o arista dado con anterioridad.



**Figura 1.** Diagrama de flujo del script de visualización de grafos

Luego del modelado del algoritmo del script se desarrolló el mismo en lenguaje de programación Python con apoyo del software *Visual Studio Code*. El script obtenido se expone a continuación:

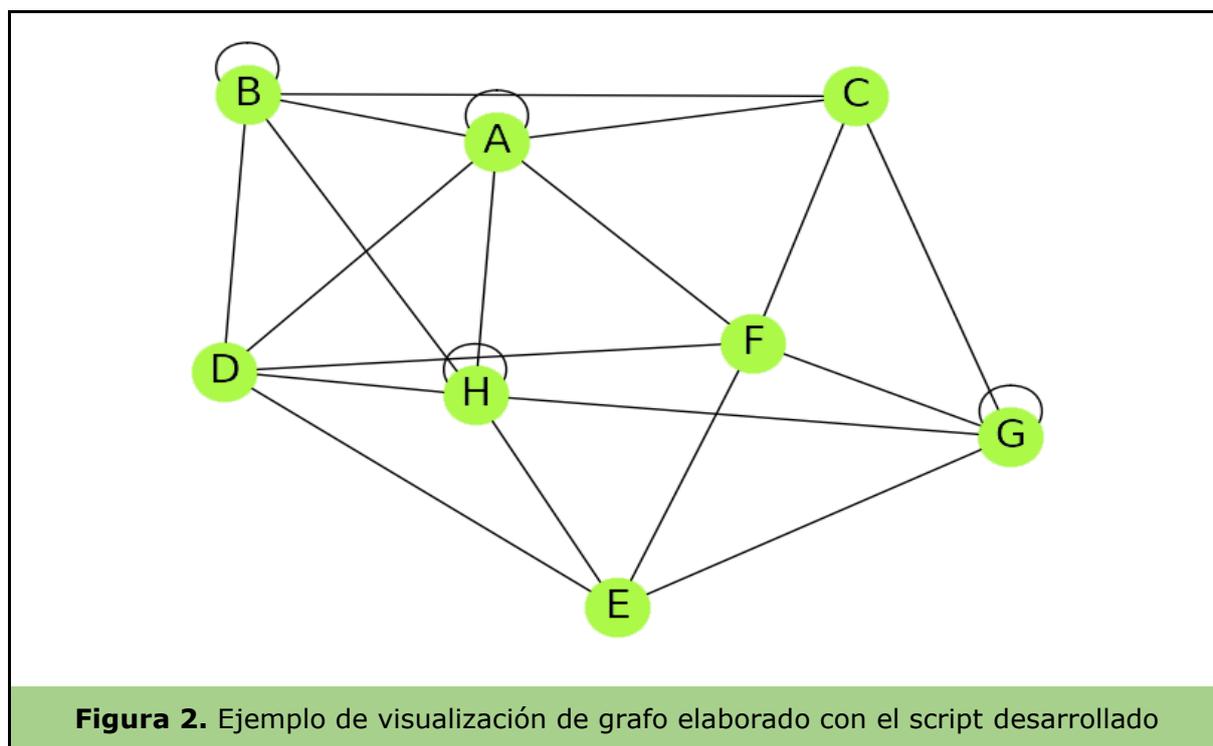
```

1  import networkx as nx
2  import matplotlib.pyplot as plt
3
4  def crear_grafo():
5      # Crear un grafo vacío
6      grafo = nx.Graph()
7
8      # Insertar nodos desde la entrada del usuario
9      num_nodos = int(input("Ingresa el número de nodos: "))
10     for i in range(num_nodos):
11         nodo = input(f"Ingresa el nombre del nodo {i + 1}: ")
12         grafo.add_node(nodo)
13
14     # Agregar aristas desde la entrada del usuario
15     num_aristas = int(input("Ingresa el número de aristas: "))
16     for i in range(num_aristas):
17         nodo1 = input(f"Ingresa el nombre del primer nodo de la arista {i + 1}: ")
18         nodo2 = input(f"Ingresa el nombre del segundo nodo de la arista {i + 1}: ")
19         grafo.add_edge(nodo1, nodo2)
20
21     return grafo
22

```

```
23 # Ejemplo de uso
24 mi_grafo = crear_grafo()
25
26 # Visualizar el grafo
27 nx.draw(mi_grafo, with_labels=True, node_size=900, font_size=20,
28         node_color='skyblue', font_color='black')
29 plt.title("Grafo de Nodos y Aristas")
30 plt.show()
```

En la programación y ejecución del algoritmo se utilizó el software *Visual Studio Code*, en lenguaje de programación *Python*. Al ejecutarlo se deben introducir los datos que solicita el script a través de la consola de comandos. Al finalizar el proceso se visualiza el grafo. La figura 2 muestra un ejemplo de grafo elaborado mediante el script.



En el ejemplo de la figura 2 se solicitó al script un grafo de ocho nodos (denominados A, B, C, D, E, F, G y H), y 24 aristas, de estas cuatro eran bucles (en los nodos A, B, G y H). Se puede comprobar que la visualización ofrece un grafo donde se aprecian todas las relaciones de manera efectiva sin la aparición de errores.

## DISCUSIÓN

En los últimos años se ha experimentado un crecimiento en las aplicaciones de la teoría de grafos, herramienta para modelizar problemas muy variados del ámbito científico y en la vida cotidiana <sup>8,9</sup>. La literatura evidencia sobradas experiencias en la implementación de los grafos en diversos contextos.

Delgado-Fernández et al.<sup>10</sup> propone el empleo de grafos en el apoyo a los estudios epidemiológicos realizados en el contexto de la pandemia de COVID-19. Los resultados simulaban escenarios de brotes producidos en la provincia de Artemisa, Cuba, a partir de datos extraídos de la web. Esto significó un paradigma flexible de representación de conocimientos que contribuyó al procesamiento de información para la toma de decisiones.

Otro estudio de Chaves et al.<sup>11</sup> desarrolló un modelo de grafo bipartito unidireccional para la identificación del plagio en ejercicios de programación en lenguaje Python. Esta herramienta resultó útil, al facilitar el reconocimiento de similitudes y diferencias entre códigos de programación.

Allauca-Melena<sup>12</sup> modeló una red de transporte en una empresa en forma de grafo para encontrar las rutas más óptimas y minimizar tiempos y costos. Los resultados identificaron oportunidades de mejora en la asignación de rutas, reducir los tiempos de entrega promedio en un 8% y costos en un 5%, lo que demostró la utilidad de la teoría de grafos en la optimización de redes de transporte.

En el área de la inteligencia artificial son conocidos los beneficios del uso de los grafos para mejorar aspectos de los sistemas. Los grafos potencian la precisión y la eficacia de los algoritmos de aprendizaje automático y los sistemas recomendadores. Se ha comprobado la efectividad de los mismos para solucionar problemas de dominio de las redes neuronales de aprendizaje automático<sup>4,13</sup>.

Se hace evidente la necesidad de aprendizaje de manera óptima de la teoría de grafos en estudiantes de SIS. Esto requiere que sean capaces de aplicarla en la solución de problemas prácticos, donde realicen análisis de la proyección de las relaciones en el grafo. Por ello resulta ineludible explotar las ventajas de las herramientas digitales para el aprendizaje de la teoría de grafos.

Existen experiencias previas en la carrera de SIS del uso de recursos digitales didácticos para el aprendizaje. Ejemplos de esto han sido la creación de videos didácticos para la enseñanza-aprendizaje de programación y gestión de bases de datos<sup>14</sup>; medios de enseñanza para la asignatura de Diseño de Bases de Datos<sup>15</sup>; o el desarrollo de una aplicación móvil para el cálculo de las tasas en estadística descriptiva<sup>16</sup>.

Esto da la medida de que es imprescindible el uso de herramientas que permitan a los estudiantes profundizar en los contenidos de la asignatura de Matemática Aplicada y la teoría de grafos. Los procesos de modelización de grafos resultan complejos, por ello se requiere que sean capaces de visualizar el producto de la aplicación de los conocimientos en este contenido, y no solo desarrollen ejercicios de forma manual.

Otro aspecto importante en el desarrollo del script fue la utilización de Python. Este es un lenguaje de programación de alto nivel y multiparadigma. Creado con el propósito de generar soluciones de cómputo para una amplia variedad de aplicaciones; entre ellas la programación de sistemas, aplicaciones WEB, la ciencia de datos y el aprendizaje automático<sup>8</sup>.

La decisión de utilizar Python para la implementación de un visualizador de grafos responde a la sencillez de la sintaxis de este lenguaje. El aprendizaje del lenguaje resulta potable, y coadyuva a la expresión en formato de códigos los conceptos y procedimientos involucrados en la visualización<sup>9</sup>. Otros autores han utilizado este mismo lenguaje para generar soluciones que implican la creación de grafos, lo cual también influyó en la elección.

Farías et al.<sup>17</sup> refiere que no existen lenguajes que establezcan una compatibilidad entre las consultas sobre bases de datos de grafos y la aplicación de diversos análisis algorítmicos en dichos grafos. A partir de algoritmos experimentales, se elaboraron grafos con Python de la modelación propuesta.

En el área de la información científica, campo de acción de los profesionales de SIS, se ha diseñado un script para la generación de grafos a partir de metadatos de revistas científicas, obtenidos del sistema OAI-PMH y códigos en Python. González-Argote et al.<sup>18</sup> considera que esta solución supone un enfoque poderoso y versátil en el análisis de la producción académica.

El script en lenguaje Python elaborado en el presente estudio crea la base para el desarrollo futuro de un software con mayores prestaciones y facilidades de uso. Se deben realizar otros estudios que aborden las fases de diseño físico, la implementación y la evaluación del impacto del producto.

## CONCLUSIONES

Se enfatiza la importancia de imbricar la teoría de grafos con herramientas de visualización efectivas en la formación de los estudiantes de SIS. El script Python diseñado proporciona una forma favorable de generar grafos y mejora el proceso de aprendizaje que permite a los estudiantes aplicar la teoría de grafos en situaciones prácticas.

## REFERENCIAS BIBLIOGRÁFICAS

1. García-Verdecia H, Méndez-Valdés RL. La transformación digital: estrategia para su implementación. Rev. científica SEg. y SAb. [Internet]. 9 de enero de 2024 [citado 2 de febrero de 2024];3(4). Disponible en: <https://seguridadysaber.seisa.cu/index.php/seisa/article/view/34>
2. González-García TR, García-Savón Y, López-Chacón A, Olazabal-Guerra DJ. Formación integral de estudiantes angolanos en la carrera Sistemas de Información en Salud. 2015-2016. Rev. Cub. Tecnol. Salud. [Internet]. 2022 [citado 2 Feb 2024];13(3). Disponible en: <https://revtecnologia.sld.cu/index.php/tec/article/view/3941>
3. García-Savón Y, Medina-Borges RM, López-Chacón A. Importancia del desarrollo de las competencias informacionales en estudiantes de la carrera de Sistemas de Información en Salud. Revista Cubana de Educación Superior. [Internet]. 2022 [citado 2 Feb 2024];41(2). Disponible en: [http://scielo.sld.cu/scielo.php?script=sci\\_arttext&pid=S0257-43142022000200027&lng=es&tlng=es](http://scielo.sld.cu/scielo.php?script=sci_arttext&pid=S0257-43142022000200027&lng=es&tlng=es).
4. Araujo-Inastrilla CR. Data Visualization in the Information Society. Seminars in Medical Writing and Education [Internet]. 2023 [citado 2 Feb 2024];2:25. <https://doi.org/10.56294/mw202325>.
5. Licenciatura en Sistemas de Información en Salud. Universidad de Ciencias Médicas de Villa Clara [Internet]. Santa Clara: Universidad de Ciencias Médicas de Villa Clara; 2024 [citado 2 de febrero de 2024]. Disponible en: [https://www.universidad.cav.sld.cu/?page\\_id=350](https://www.universidad.cav.sld.cu/?page_id=350)
6. Perche-Alvarez AE, Medina-Borges RM, López-Chacón A, García-Savón Y, Roche-Madrugal MC, Valdés-Velázquez G. El desarrollo de las habilidades profesionales en la carrera de sistemas de información en salud. Rev. Cub. Tecnol. Salud. [Internet]. 2019 [citado 2 Feb 2024];10(4). Disponible en: <https://revtecnologia.sld.cu/index.php/tec/article/view/1328>
7. Roche-Madrugal MC. Matemática Aplicada. La Habana: ECIMED; 2007.
8. Cáceres-González AE. Matemáticas Discretas Una perspectiva funcional con Python 3.x. Cunduacán, Tabasco: Universidad Juárez Autónoma de Tabasco; 2023.
9. Jordán-Lluch C, Sanabria-Codeçal E. Aprendiendo a modelizar con grafos. Revista de Investigación. 2021;11(2):055–066.
10. Delgado-Fernández T, Stuart-Cárdenas ML. Grafos de conocimiento para gestionar información epidemiológica sobre COVID-19. Revista Cubana de Información en Ciencias de la Salud. 2021;32(4):e1686.
11. Chaves RC, Albuquerque KM, Chaves RC, Gouveia T. Modelo para Deteccao de Plagio em Exercicios de Programacao na Linguagem Python. Brasil: Instituto Federal da Paraiba (IFPB); 2023.
12. Allauca-Melena JE. Aplicación de la teoría de grafos en la optimización de redes de transporte. Revista CINTE. 2023;1(1).
13. Vilanova-González C. Redes Neuronales sobre Grafos (GNN): una Prueba de Concepto para sistemas recomendadores "Graph Neural Networks (GNN): A Proof of Concept for

- Recommender Systems". [Trabajo de Fin de Grado en Ingeniería Informática]. La Laguna: Universidad de la Laguna; 2023.
14. Joao-Tchongo EL, López-Chacón A, García-Savón Y, Roche-Madrugal MC. Videos didácticos en el proceso de enseñanza-aprendizaje de Programación y Gestión de Bases de Datos. La Habana: IV Congreso Internacional "Tecnología y Salud"; 2019.
  15. Roche-Madrugal MC, García-Savón Y, Lombillo-Crespo OO. Medios de enseñanza para profundizar los conocimientos del curso de superación Diseño de Bases de Datos. Rev. Cubana. Tecnol. Salud. [Internet]. 2014 [citado 2 Feb 2024]. 5(4):e231. Disponible en: <https://revtecnologia.sld.cu/index.php/tec/article/view/231>
  16. Araujo Inastrilla CR. Aplicación para dispositivos móviles para el cálculo de tasas en salud. Inmedsur [Internet]. 2021 [citado: 2 Feb 2024]; 4(3): e168. Disponible en: <http://www.inmedsur.cfg.sld.cu/index.php/inmedsur/article/view/168>
  17. Farías B, Romero J, Soto A, Reutter J. Graph Analytics en Bases de Datos de Grafos. Chile: Department of Computer Science, School of Engineering. Pontificia Universidad Católica de Chile; 2021.
  18. Gonzalez-Argote D, Gonzalez-Argote J. Generation of graphs from scientific journal metadata with the OAI-PMH system. Seminars in Medical Writing and Education [Internet]. 2023 [citado 2 Feb 2024];2:43. Disponible en: <https://doi.org/10.56294/mw202343>.

## CONTRIBUCIÓN DE AUTORÍA

**Alejandro Araujo Inastrilla:** Conceptualización, Investigación, Software, Redacción-borrador origina, Redacción-revisión y edición.

**Carlos Rafael Araujo Inastrilla:** Metodología, Administración del proyecto, Visualización, Redacción-borrador origina, Redacción-revisión y edición.

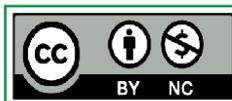
**María del Carmen Roche Madrugal, Anet López Chacón, Delia Gálvez Medina:** Supervisión, Validación, Visualización, Redacción-revisión y edición.

## CONFLITOS DE INTERESES

No se declara.

## FINANCIACIÓN

No se necesitó financiación.



Los artículos de *Revista Cubana de Tecnología de la Salud* se compar-  
ten bajo los términos de la Licencia **Creative Commons Atribución-No  
Comercial 4.0. Internacional**