



Grafos y Python: integración con la Matemática Aplicada a Sistemas de Información en Salud

Autor:

Alejandro Araujo Inastrilla

Matemática aplicada a los Sistemas de Información en Salud



Fomenta 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



TEORÍA DE GRAFOS

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





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



No se cuenta con las herramientas didácticas que permitan la visualización de los grafos



Oportuno disponer de herramientas que coadyuven al desarrollo de habilidades en la utilización de grafos



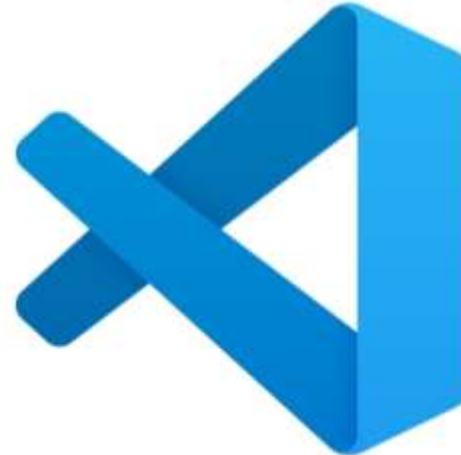
Objetivo

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 desarrolla 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





Python



- Sencillez de la sintaxis de este lenguaje
- El aprendizaje de este resulta potable
- Coadyuva a la expresión en formato de códigos los conceptos y procedimientos involucrados en la visualización
- Otros autores han utilizado este mismo lenguaje para generar soluciones que implican la creación de grafos

RESULTADOS



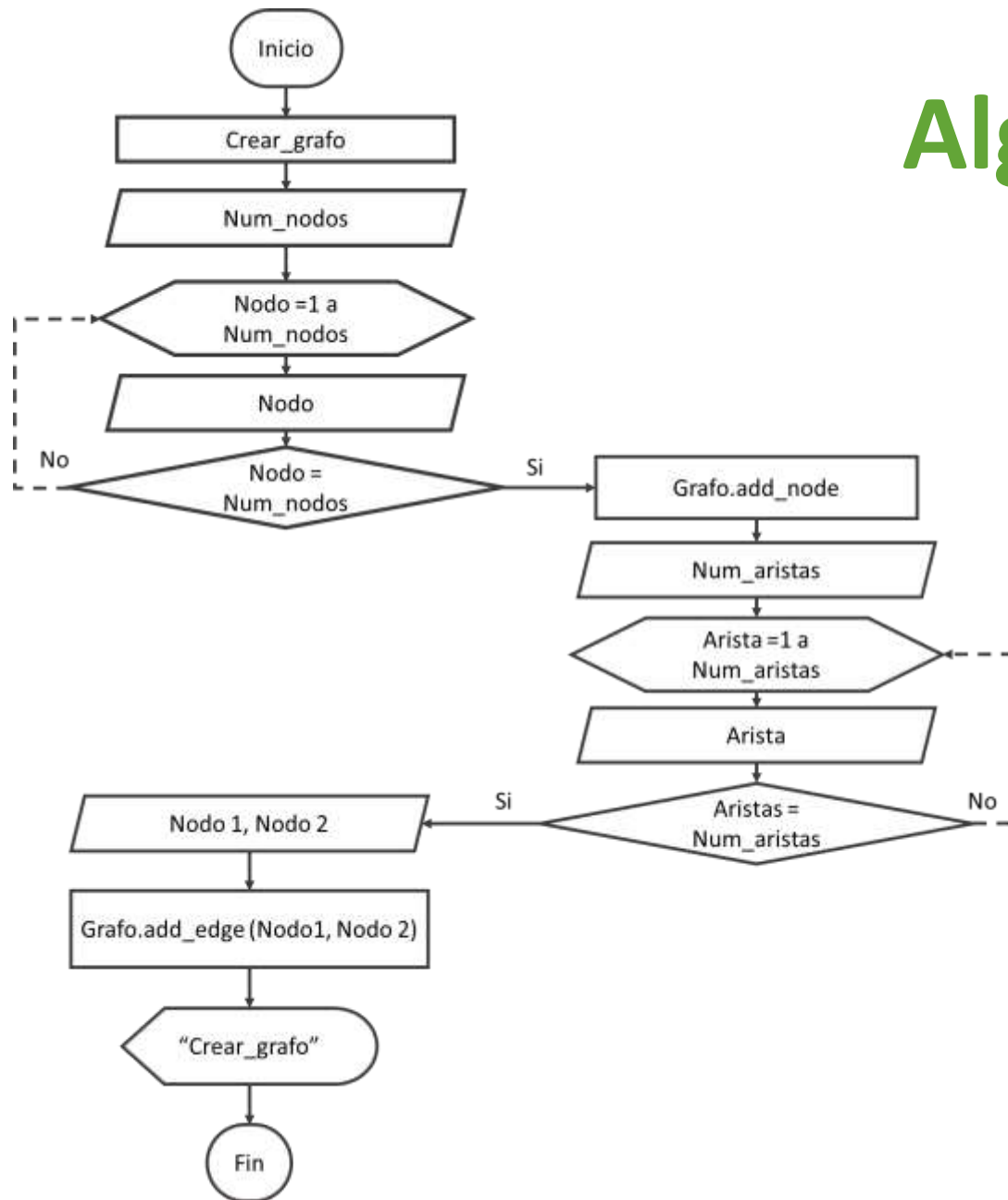
**Introducir
nodos**

**Introducir
relaciones**

**Visualizar
el grafo**



Algoritmo



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

Script en lenguaje Python

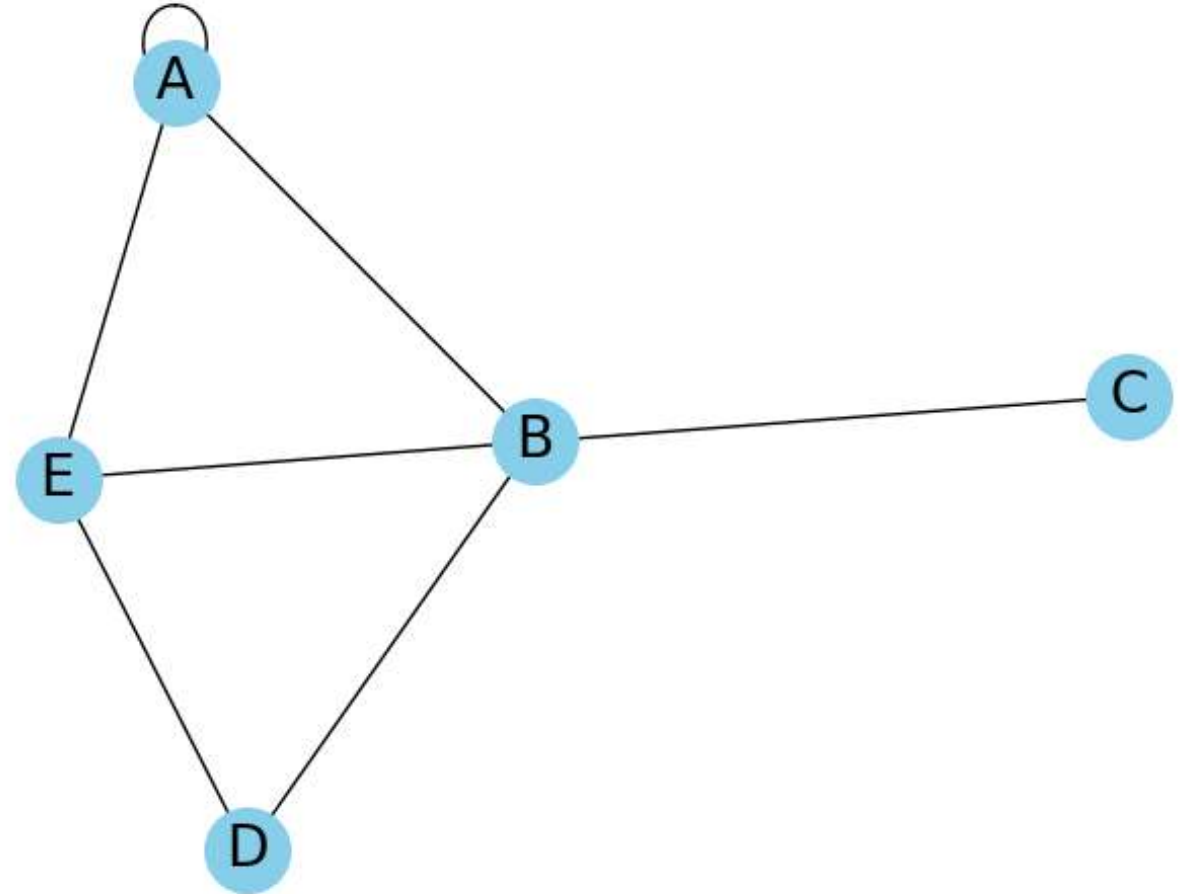
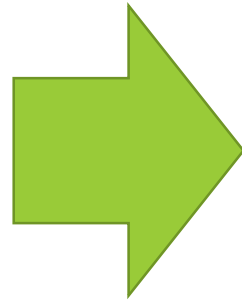


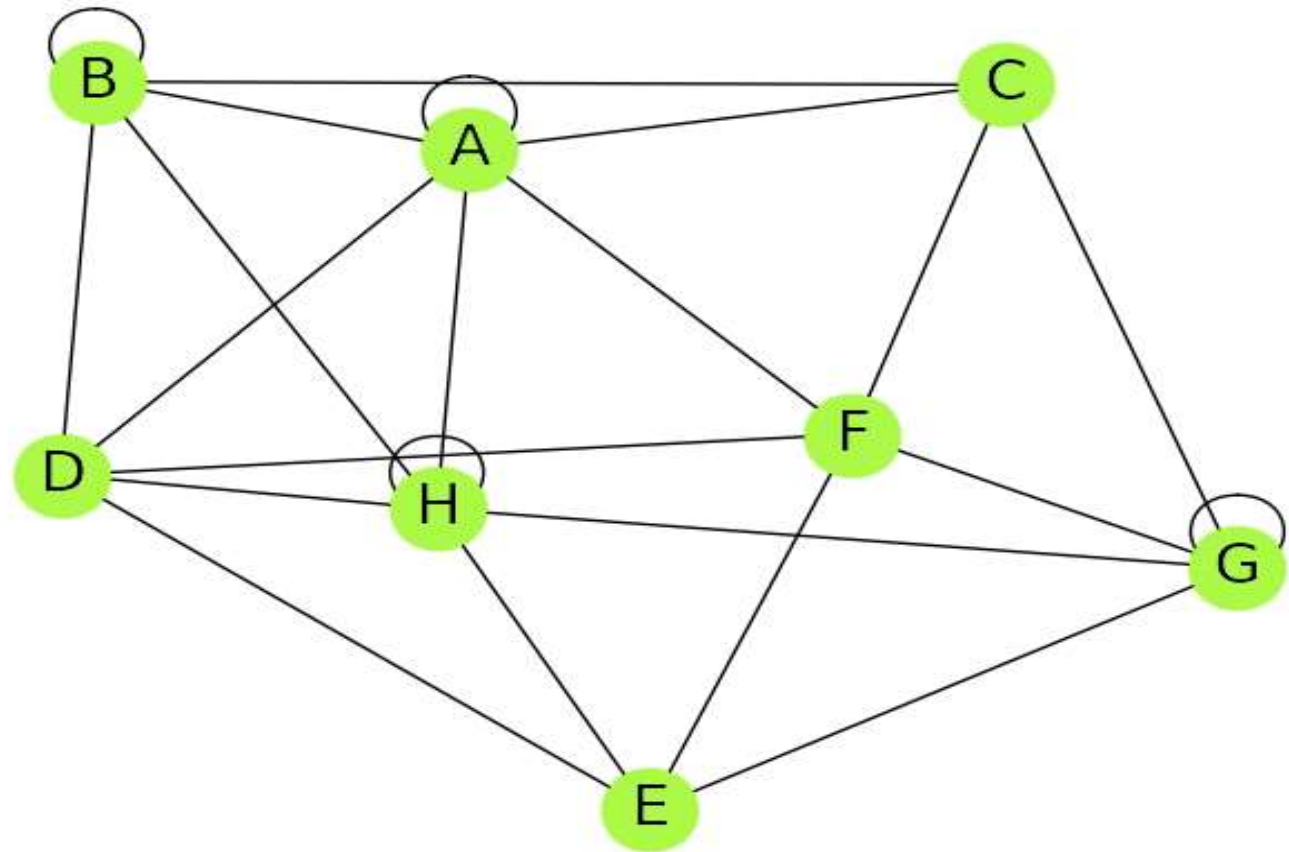
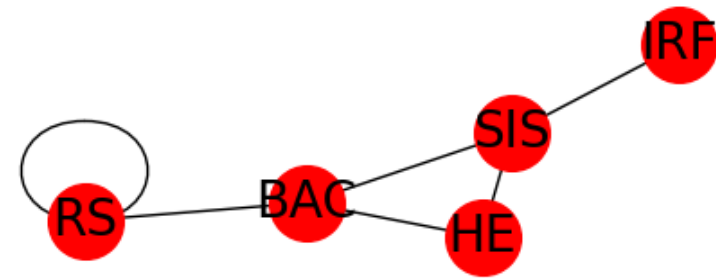
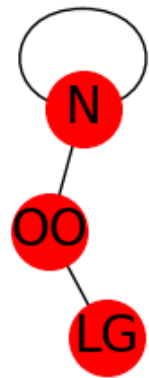
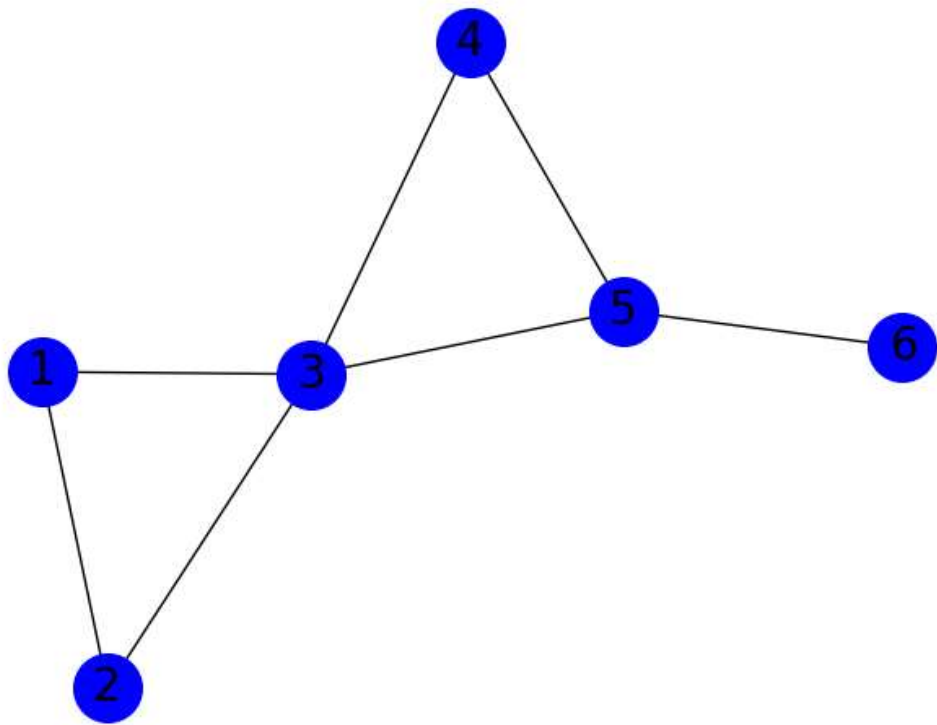
```
1 import networkx as nx
2 import matplotlib.pyplot as plt
3
4 def crear_grafo():
5     # Crear un grafo vacío
6     grafo = nx.Graph()
7
8     # Insertar nodos desde la entrada del usuario
9     num_nodos = int(input("Ingresa el número de nodos: "))
10    for i in range(num_nodos):
11        nodo = 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, node_color='skyblue', font_color='black')
28 plt.title("Grafo de Nodos y Aristas")
29 plt.show()
```

Relación

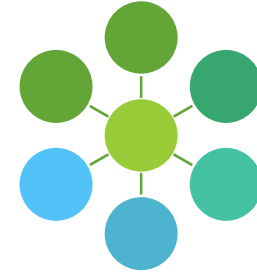
Nodo 1	Nodo 2
A	B
A	A
B	C
B	D
D	E
E	A
E	B

Grafo





CONCLUSIONES



Se enfatiza la importancia de imbricar la teoría de grafos con herramientas de visualización efectivas en la educación de los estudiantes de SIS. El script Python propuesto es un paso hacia el logro de este objetivo, al proporcionar una forma favorable para generar grafos y mejorar el proceso de aprendizaje al permitir a los estudiantes aplicar la teoría de grafos en situaciones prácticas