

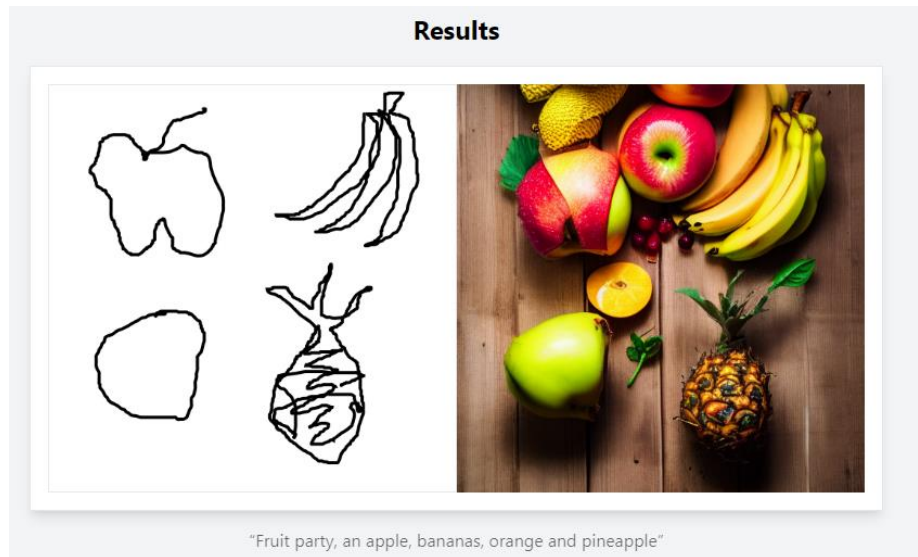
Práctica Estructura de Datos

Resuelto por: Nicolas Acha Palacios

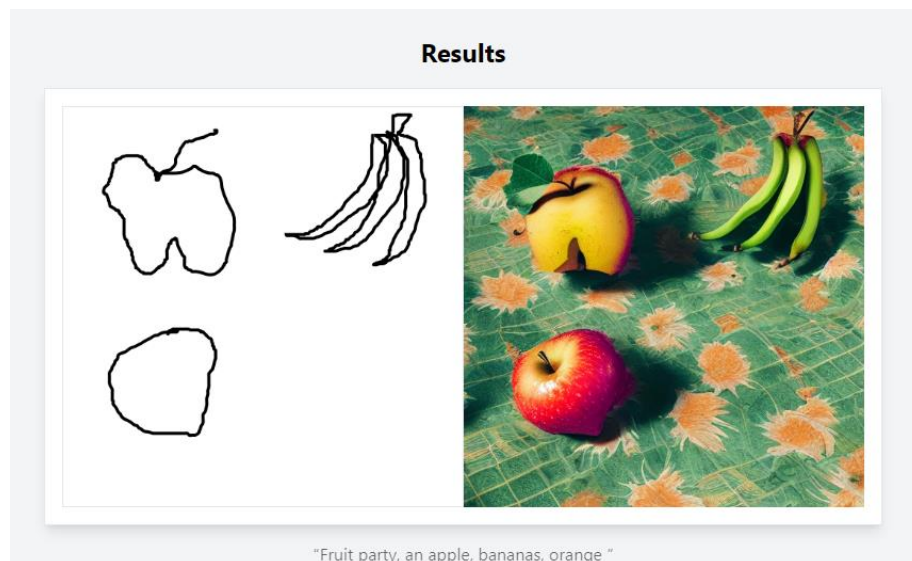
1. Ingrese a Scribblediffusion:

<https://scribblediffusion.com/>

- Dibuje 4 frutas (apoyando el click del mouse una vez por fruta)
- Genere un dibujo describiendo las frutas



- Deshaga el dibujo de una de las frutas
- Vuelva a generar el dibujo describiendo las frutas que quedan



- Capture las fotos y adjúntelas en su respuesta.

Ingrese a Jupyter, genere una Pila. Añada en orden el nombre de las frutas que incluyó en su dibujo. En líneas inferiores ilustre con una función pop el cómo la máquina quita el último elemento de su dibujo.

```

+ Code + Markdown | ▶ Run All ⏮ Restart ⚙ Clear All Outputs | 📄 Variables 📄 Outline ... Python 3.10.11

• ### Genere una Pila. Añada en orden el nombre de las frutas que incluyó en su dibujo.

# Lista
frutas = []

# Apilando frutas
frutas.append('Apple')
frutas.append('Bananas')
frutas.append('Orange')
frutas.append('Pineapple')

# Ver pila de frutas:

print('Fruit party:')
print(frutas)

[3] ✓ 0.0s Python
...
Fruit party:
['Apple', 'Bananas', 'Orange', 'Pineapple']

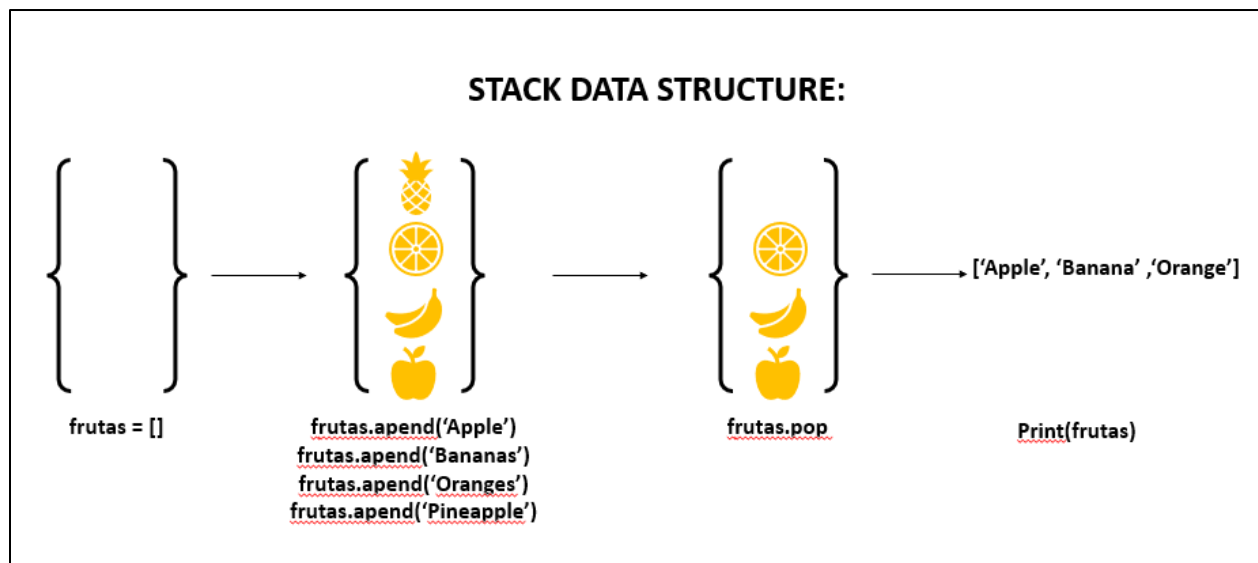
▶ ▾ ### En líneas inferiores ilustre con una función pop el cómo la máquina quita el último elemento de su dibujo.

# pop() function to pop
frutas.pop()

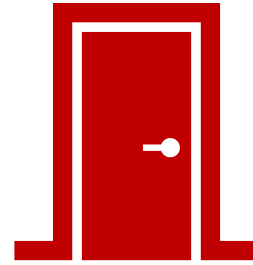
print('\nFrutas despues de funcion .pop():')
print(frutas)

[4] ✓ 0.0s Python
...
Frutas despues de funcion .pop():
['Apple', 'Bananas', 'Orange']
+ Code + Markdown

```



2. Estas personas están esperando una entrevista de trabajo.



La asistente sentó al primero en llegar lado de la puerta del cuarto de entrevistas, para que los primeros en llegar tengan una mayor proximidad.

Cree una cola en Python, muestre cómo la cola “avanza” atendiendo a dos personas y obtenga el nombre del tercero a ser entrevistado.

```
import queue
✓ 0.0s Python

# Queue de trabajo
entrevista = queue.Queue()

#añadimos elementos
entrevista.put("Tamara")
entrevista.put("Andrea")
entrevista.put("Sonia")
entrevista.put("Miguel")
entrevista.put("Jose")

medida=entrevista.qsize()
print(medida)
✓ 0.0s Python

5

#Avance de cola:

#Atienden al primer entrevistado:
entrevista.get()
#Atienden al segundo entrevistado:
entrevista.get()
#Nombre del siguiente entrevistado:
nombre_entrevista = entrevista.get()

print(f"El siguiente en ser entrevistado(a) es {nombre_entrevista}")
✓ 0.0s Python

El siguiente en ser entrevistado(a) es Sonia
```

3. Completar el árbol de ejemplo de Python del material de curso, con los nietos faltantes.

Abuelo: Martin, Hijos: Andrés (nietos: Mateo), Christian (nietos: José, Melany) y Estefany (nietos: Andrea)

```
#3. Completar el árbol de ejemplo de Python del material de curso, con los nietos faltantes.
#Abuelo: Martin, Hijos: Andrés (nietos: Mateo), Christian (nietos: José, Melany) y Estefany (nietos: Andrea)
```

```
class Nodo:
    def __init__(self, valor, nombre):
        self.valor = valor
        self.hijos = []
        self.nombre = nombre

    def agregar_hijo(self, hijo):
        self.hijos.append(hijo)

def imprimir_arbol(nodo, nivel=0):
    print(" " * nivel + "|_" + nodo.nombre)
    for hijo in nodo.hijos:
        imprimir_arbol(hijo, nivel + 1)

# Agregar raíz, hijos y nietos:
# creamos los nodos del árbol con sus respectivos nombres
raiz = Nodo(1, "Martin")
hijo1 = Nodo(2, "Andrés")
hijo2 = Nodo(3, "Christian")
hijo3 = Nodo(4, "Estefany")
nieto1 = Nodo(5, "Mateo")
nieto2 = Nodo(6, "Jose")
nieto3 = Nodo(7, "Melanie")
nieto4 = Nodo(8, "Andrea")
```

```
# establecemos las relaciones padre-hijo
raiz.agregar_hijo(hijo1)
raiz.agregar_hijo(hijo2)
raiz.agregar_hijo(hijo3)
hijo1.agregar_hijo(nieto1)
hijo2.agregar_hijo(nieto2)
hijo2.agregar_hijo(nieto3)
hijo3.agregar_hijo(nieto4)
```

```
# Imprimir el árbol
imprimir_arbol(raiz)
```

✓ 0.0s

Python

```
|_Martin
|_Andrés
|_Mateo
|_Christian
|_Jose
|_Melanie
|_Estefany
|_Andrea
```

4. Genera un grafo tomando en cuenta las siguientes relaciones:

- José es amigo de Tamara y Tonia
- Tamara es amiga de Andrea

```
#4. Genera un grafo tomando en cuenta las siguientes relaciones:
# * José es amigo de Tamara y Tonia
# * Tamara es amiga de Andrea

import networkx as nx
import pandas as pd
import warnings
import matplotlib.pyplot as plt
from pyvis.network import Network

# Create an empty graph
G = nx.Graph()

# Add multiple nodes
G.add_nodes_from(["Jose", "Tamara", "Tonia", "Andrea"])

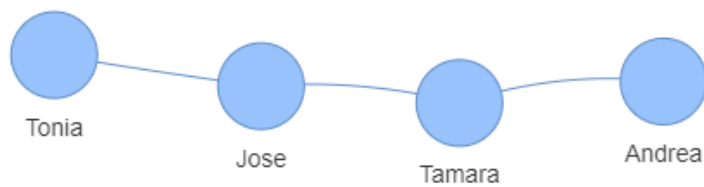
# Add edges to connect nodes
G.add_edge("Jose", "Tamara")
G.add_edge("Jose", "Tonia")
G.add_edge("Tamara", "Andrea") # New edge

# Create a pyvis network instance
nt = Network(notebook=True)

# Add nodes and edges to the network
for node in G.nodes:
    nt.add_node(node)
for edge in G.edges:
    nt.add_edge(edge[0], edge[1])

# Show the network
nt.show("amigos.html")
```

✓ 02s Python



5. Suponga que es periodista y desea invitar a alguien que haya vivido la guerra del agua y la hiperinflación de los años 80 para conocer su opinión en un programa de televisión. Indagando, de diversas fuentes se recopilan las siguientes listas de nombres.

Personas que vivieron la guerra del agua:

Mariana Mariscal
Carla Martínez
Ursula Ibañez
Ana Rosales
Andrés Cáceres

Personas que vivieron la hiperinflación:

Martin Arandia
Mariana Mariscal
Sergio Bonavide
Andrés Ibañez
Diego Almagro

¿A quiénes invitará a su programa? Debe usar Python para su respuesta.

```
# Create an empty graph
G = nx.Graph()

# Add multiple nodes
G.add_nodes_from(["Guerra del Agua", "Hiperinflacion", "Mariana Mariscal", "Carla Martinez", "Ursula Ibanez", "Ana Rosales", "Andres Caceres",
                  "Martin Arandia", "Sergio Bonavide", "Andres Ibanez", "Diego Almagro"])

# Add edges to connect nodes
G.add_edge("Guerra del Agua", "Mariana Mariscal")
G.add_edge("Guerra del Agua", "Carla Martinez")
G.add_edge("Guerra del Agua", "Ursula Ibanez")
G.add_edge("Guerra del Agua", "Ana Rosales")
G.add_edge("Guerra del Agua", "Andres Caceres")
G.add_edge("Hiperinflacion", "Martin Arandia")
G.add_edge("Hiperinflacion", "Mariana Mariscal")
G.add_edge("Hiperinflacion", "Sergio Bonavide")
G.add_edge("Hiperinflacion", "Andres Ibanez")
G.add_edge("Hiperinflacion", "Diego Almagro")

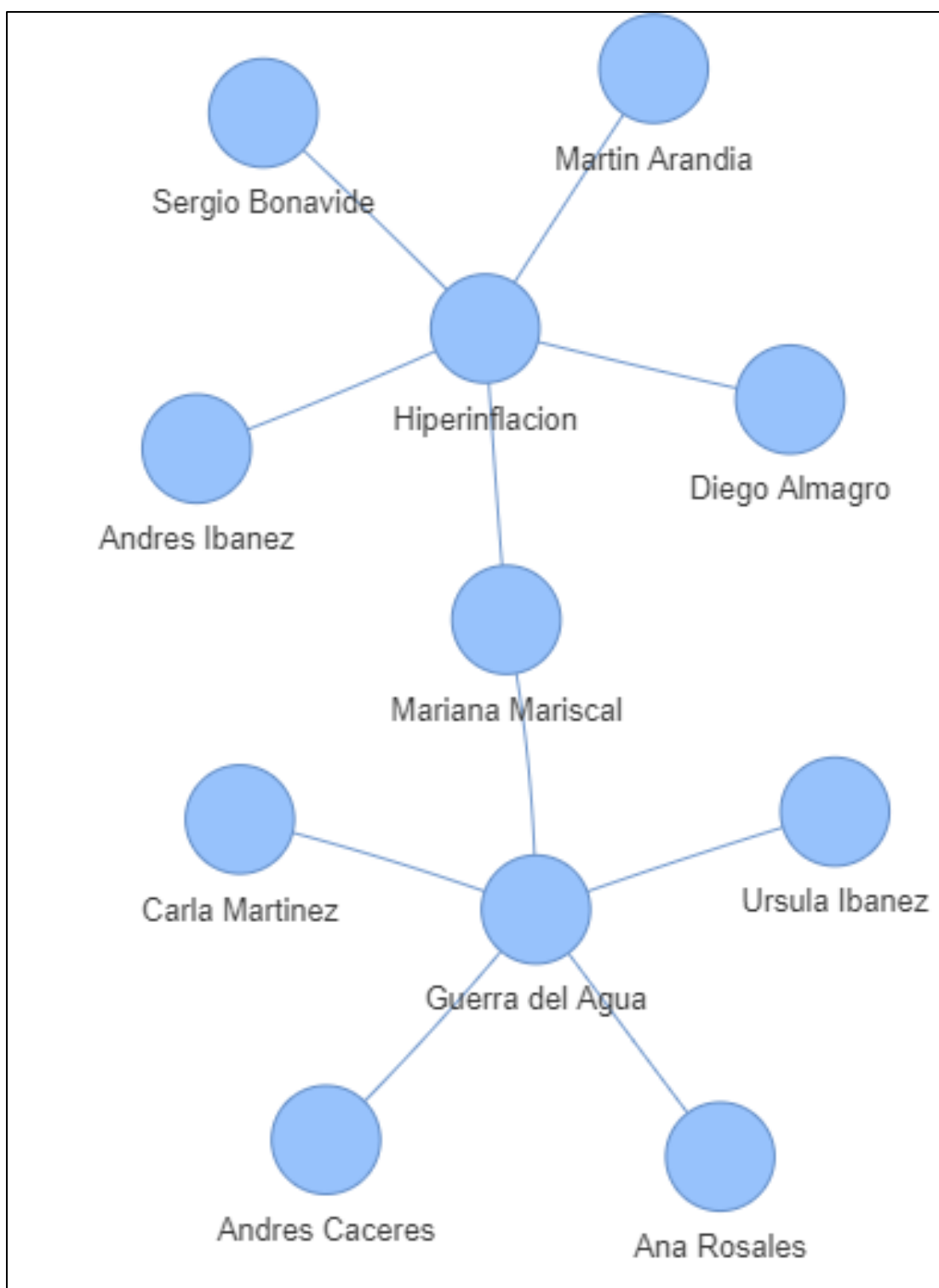
# Create a pyvis network instance
nt = Network(notebook=True)

# Add nodes and edges to the network
for node in G.nodes:
    nt.add_node(node)
for edge in G.edges:
    nt.add_edge(edge[0], edge[1])

# Show the network
nt.show("entrevista.html")
```

✓ 0.0s

Python



Invitaría a Mariana Mariscal

6. Replica el ejemplo del mapeo para 6 ciudades en Bolivia y en qué departamento están ubicadas.

```
# 6. Replica el ejemplo del mapeo para 6 ciudades en Bolivia y en qué departamento están ubicadas.

Bolivia = {
    "Sucre": "Chuquisaca",
    "N. Senora de LPz": "La Paz",
    "El Alto": "La Paz",
    "Santa Cruz de la Sierra": "Santa Cruz",
    "Montero": "Santa Cruz",
    "La Guardia": "Santa Cruz",
    "Tarija": "Tarija"
}
print(Bolivia)
```

✓ 0.0s Python

```
{'Sucre': 'Chuquisaca', 'N. Senora de LPz': 'La Paz', 'El Alto': 'La Paz', 'Santa Cruz de la Sierra': 'Santa Cruz', 'Montero': 'Santa Cruz', 'La Guardia': 'Santa Cruz', 'Tarija': 'Tarija'}
```