Prueba de escritorio del Algoritmo de Floyd's Tortoise

Viviana Marcela Garcia Valderrama
Braian Felipe Ramírez Ortiz
Brenda Lorena Vargas Parra

I. Introducción

El algoritmo de Floyd's Tortoise, comúnmente denominado el algoritmo del puntero lento y rápido, desempeña un papel esencial en la detección de ciclos dentro de estructuras de datos, y su relevancia se extiende a diversos campos, incluyendo la programación y la informática teórica. En el transcurso de este informe, llevaremos a cabo una exhaustiva prueba de escritorio de este algoritmo, basada en un ejercicio práctico que previamente se resolvió en una sesión de clase. Para una comprensión más profunda de su funcionamiento, hemos empleado la herramienta PyTutor, la cual nos permite analizar de manera detallada el comportamiento de los punteros a lo largo del código. A través de esta exploración, se busca arrojar luz sobre la efectividad y eficacia del algoritmo de Floyd's Tortoise en la detección de ciclos en estructuras de datos, contribuyendo así a nuestro conocimiento sobre su aplicación práctica y su valor en la solución de problemas computacionales.

En este informe, describiremos paso a paso el ejercicio práctico en el que se aplicó el algoritmo de Floyd's Tortoise, detallando cada movimiento de los punteros y su interacción con la estructura de datos. Además, presentaremos resultados y observaciones significativas derivadas de la prueba de escritorio, lo que permitirá comprender a fondo la efectividad de este algoritmo en la detección de ciclos. A lo largo de este proceso, destacaremos la utilidad y relevancia de Floyd's Tortoise en la resolución de problemas prácticos que involucran estructuras de datos cíclicas, subrayando su importancia en el ámbito de la programación y la informática teórica.

II. Desarrollo

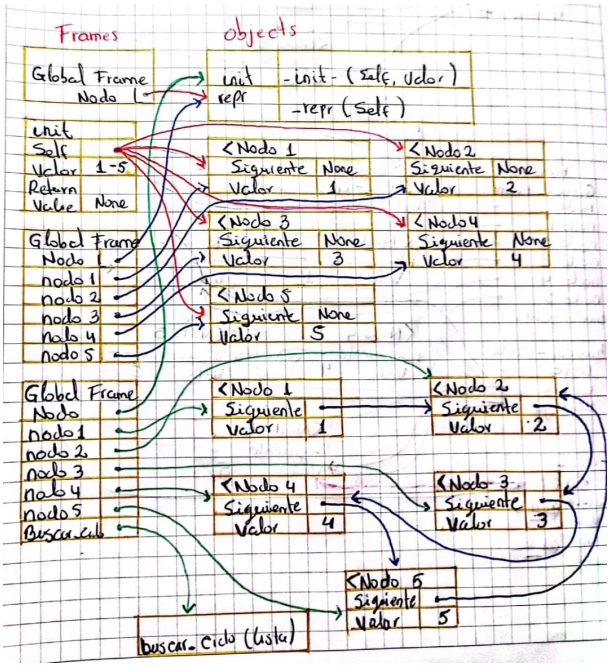
A continuación, se describe el ejercicio práctico y los pasos seguidos para validar el funcionamiento del algoritmo de Floyd's Tortoise.

A. Ejercicio Práctico

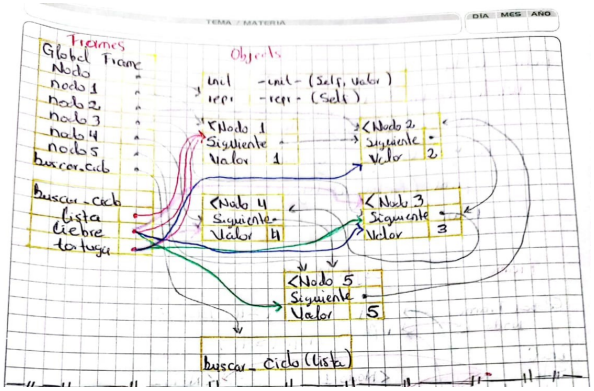
El ejercicio práctico consiste en un grafo enlazado en el que se debe determinar si existe un ciclo y, en caso afirmativo, identificar el punto de inicio del ciclo. El algoritmo de Floyd's Tortoise se utiliza para abordar este problema.

A continuacion se adjunta todo el ejercicio desarrollado a mano.

1) Rojo: Este color se empleó para representar la creacion de los nodos del 1 al 5 como se puede observar.



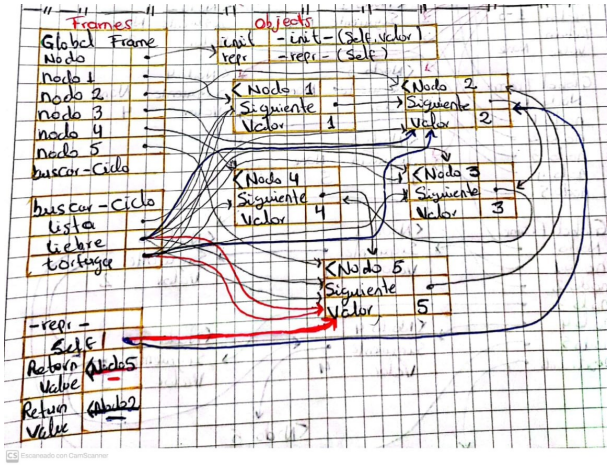
2) Azul: Con este color se le dió un valor a cada nodo del 1 al 5.
3) Verde: Este color se emplea para representar los punteros que permiten la conexion entre los nodos.



1) Rojo: este color permite inicializar los punteros de liebre y tortuga en la raiz del grafo.
2) Azul: este color nos permite identificar el avance de la liebre y la tortuga, es decir nos permite identificar

cuando la liebre avanza 2 posiciones y la tortuga solo una. En este caso la liebre se encuentra en el nodo numero 3 mientras que la tortuga apenas va en el 2.

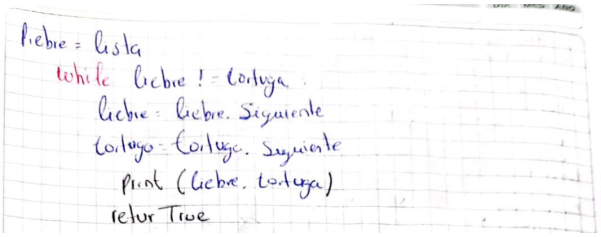
3) Verde: en este paso la liebre se encuentra al final del arbol (nodo 5) mientras que la tortuga se encuetra en el nodo 3.
4) Lila: en este color, la liebre se encuetra en el nodo numero 3 ya que avanzó otros 2 espacios, lo cual lo llevó al nodo numero 3, mientras que la tortuga se encuentra en el nodo numero 4.

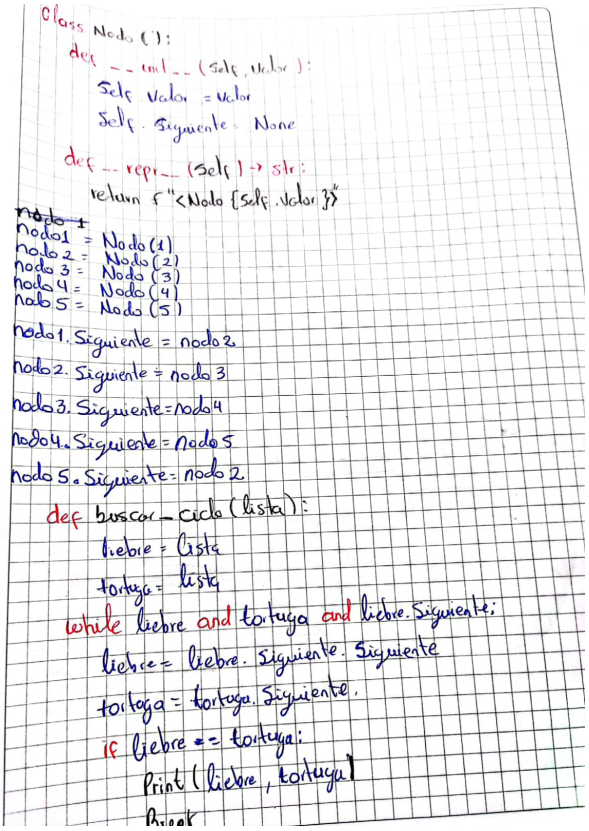


1) Azul: Determina que el nodo inicial del ciclo es el numero 2, es decir: la tortuga y la liebre se encuentran en el nodo 2 determinando asi el inicio del ciclo dentro del grafo.
2) Rojo: aca la tortuga y la liebre se encuentran en el nodo numero 5 determinando que ese es el final del ciclo que se encuentra en el grafo.

B. Imagenes del codigo hecho a mano

Este codigo se hizo con el propisito de realizar la prueba de escritorio del algoritmo de Floyd's Tortoise





C. Pasos del Algoritmo

A continuacion se explicara paso por paso el comportamiento del algoritmo de Floyd's Tortoise:

1) Inicialización de dos punteros: tortuga y liebre. Ambos comienzan en el nodo inicial de la lista enlazada.
2) Los punteros avanzan a diferentes velocidades. La tortuga se mueve un paso a la vez, mientras que la liebre se mueve dos pasos a la vez.
3) Se repiten los pasos 2 hasta que los punteros se encuentren en un mismo nodo o uno de ellos llegue al final de la lista.
4) Una vez que los punteros se encuentran, se reinicia la tortuga en el nodo inicial y se avanza ambas a la misma velocidad (un paso a la vez) hasta que se vuelvan a encontrar en el punto de inicio del ciclo (si existe).

III. Conclusiones

En la prueba de escritorio del algoritmo de Floyd's Tortoise realizada en este ejercicio, se demostró su eficacia en la

detección de ciclos en grafos ciclicos. El algoritmo es capaz de identificar el punto de inicio y final del ciclo, si este existe en el grafo. Su simplicidad y eficiencia lo convierten en una herramienta valiosa en la resolución de problemas relacionados con estructuras de datos que involucran ciclos.

IV. Referencias

[1] Floyd, R. W. (1967). "Tortoise and Hare". Communications of the ACM, 22(11), 643–644.