

UADE



Programacion II

Cuatrimestre 1 de 2025

TRABAJO PRÁCTICO

Curso: Jueves, Mañana.

1166210

Calveira Gonzalo.

Fecha de presentación: 26/06/2025

Observaciones:

.....

.....

.....

.....

Optimización de Rutas Urbanas Usando A*

Problemática:

En aplicaciones de viajes como Uber y DIDI, la misma, tiene el conflicto de elegir el camino más eficiente para cierto destino. El desafío no es simplemente elegir el más corto, sino el más rápido o conveniente pero esta dado por factores como el tránsito, velocidad permitida, cortes, intersecciones y semáforos.

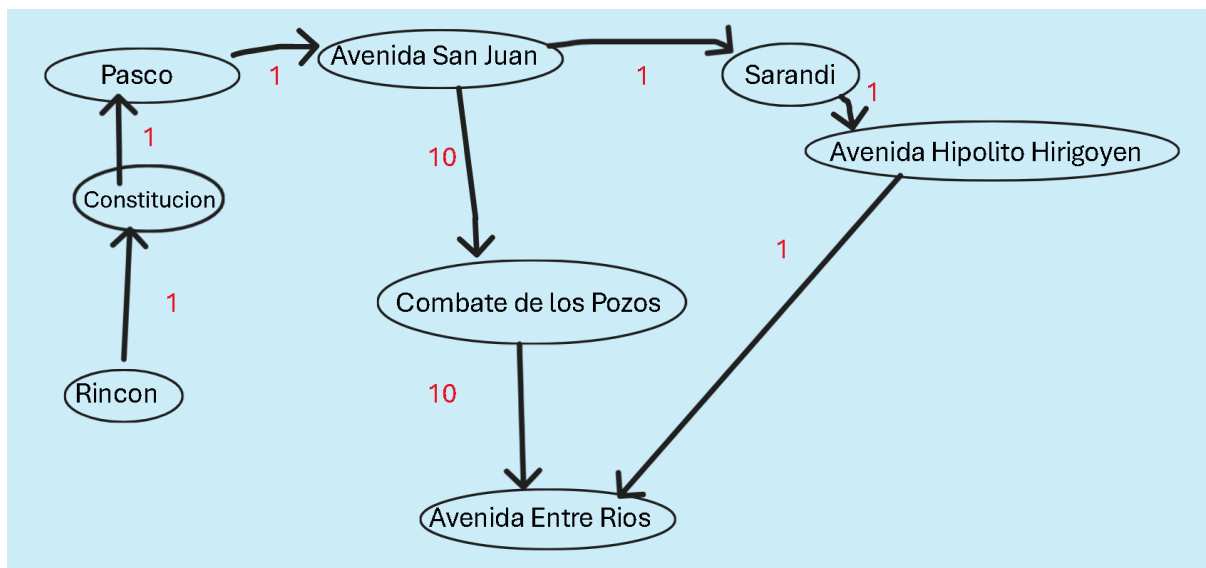
Por ejemplo, si queremos ir desde “Rincon” hasta “Avenida Entre Rios” la aplicación puede elegir la ruta que es más corta, pero tiene más transito o la ruta que es más larga, pero con transito fluido. La aplicación elegirá la última ruta mencionada, aunque tenga más distancia.

El objetivo del proyecto es dar la solución para que la aplicación elegia el camino más eficiente.

Modelado del programa:

Empezamos creando un mapa urbano como un grafo no dirigido donde cada nodo representa una calle y la arista representa un posible trayecto entre las calles con un peso asociado (este peso es dependiendo los factores mencionados anteriormente).

La estructura del grafo es de la siguiente manera:



Los nombres de las calles esta dado por una clase llamada “Calle” que implementa una interfaz “ICalle”.

Para crear los nodos, se crea una clase llamada “Nodo” que implementa una interfaz “INodo”. Las relaciones se establecen en la clase “Grafo”, que implementa una interfaz “IGrafo”, con el método “agregarNodo(T Nodo)”, que recibe la calle y lo transforma en nodo, y el método “agregarArista(T origen,T destino, T peso)” que crea la relación. Ambas clases trabajan con la estructura HashMap.

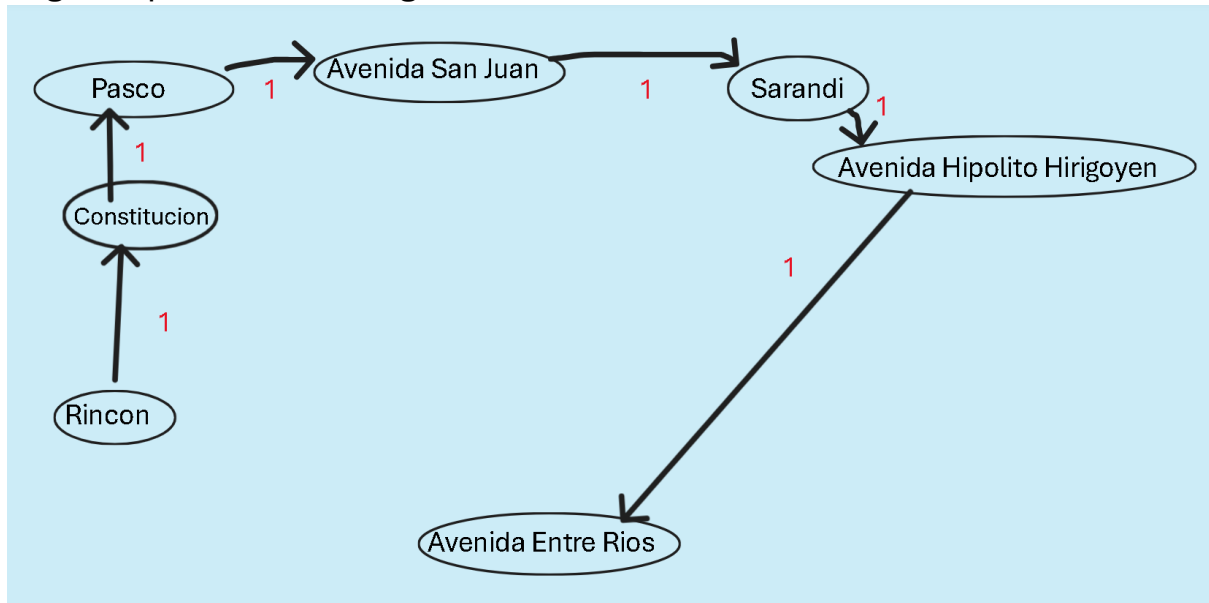
El cálculo del costo desde “Rincon” hasta “Avenida Entre Rios” estará dado por el costo real desde el origen hasta el nodo actual más una estimación del costo del nodo actual hasta el destino (heurística).

En este caso utilizamos una heurística personalizada debido a que no sabemos en tiempo real como están las calles, es decir, siempre vamos a tener la información desactualizada, por eso implementamos que la misma devuelva 0.

Resultado y Conclusión:

Al ejecutar el programa nos devuelve con el camino de “Rincon”, “Constitucion”, ”Pasco”, ” Avenida San Juan”, “Sarandi” y "Avenida Hipolito Hirigoyen”.

El grafo quedaría de la siguiente manera:



Esto indica que, aunque la distancia podría ser mayor, el algoritmo detecta que ese camino es el más eficiente.

La conclusión es que el algoritmo AStar es eficiente para encontrar el camino mas optimo pero es lento para realizarlo con muchos caminos posibles.