Alumno: Carlos Andrade

Docente: Ing. Diego Quisi.

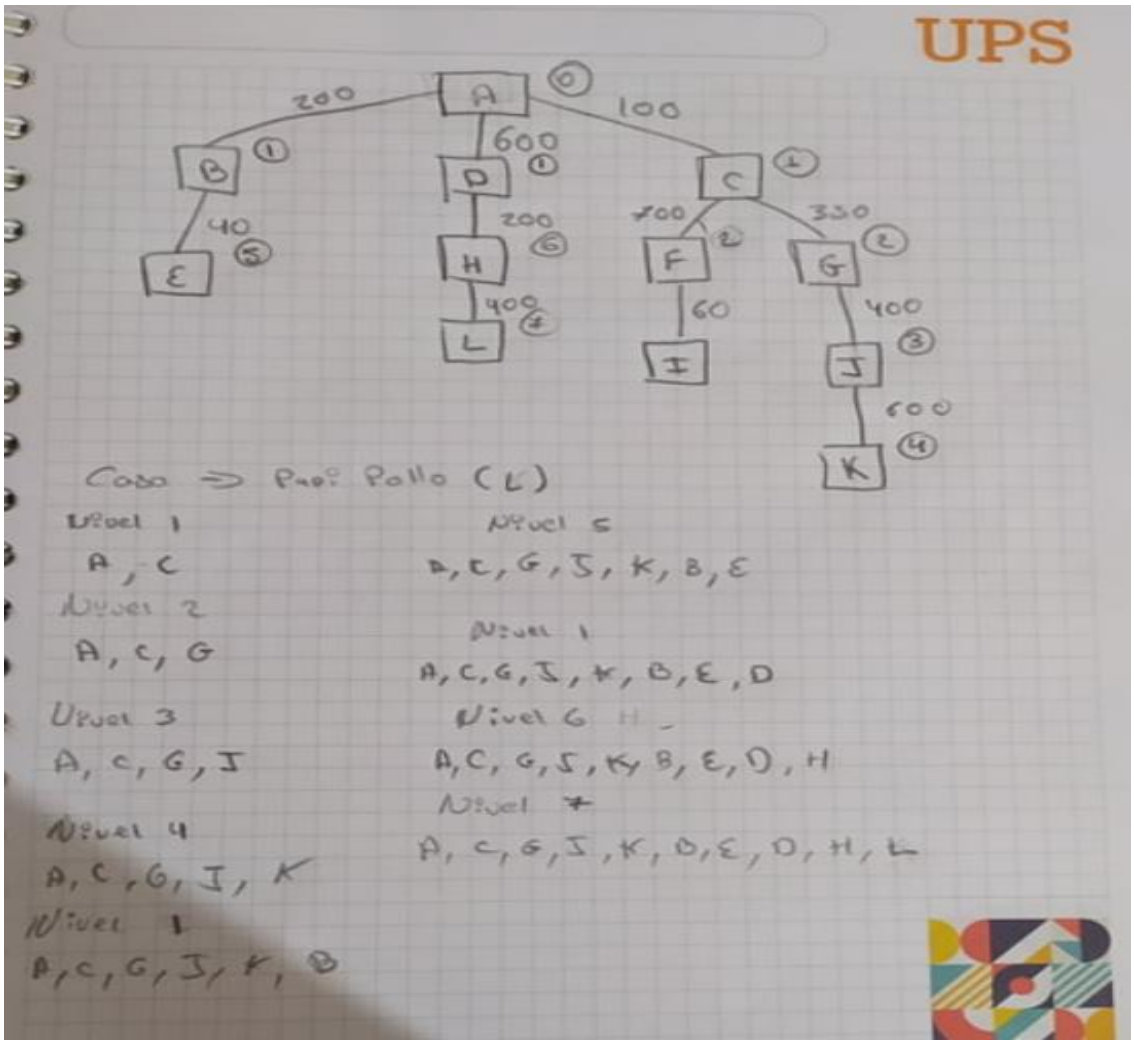
Materia: IA

Ciclo: 9no

Fecha: 20/05/2020

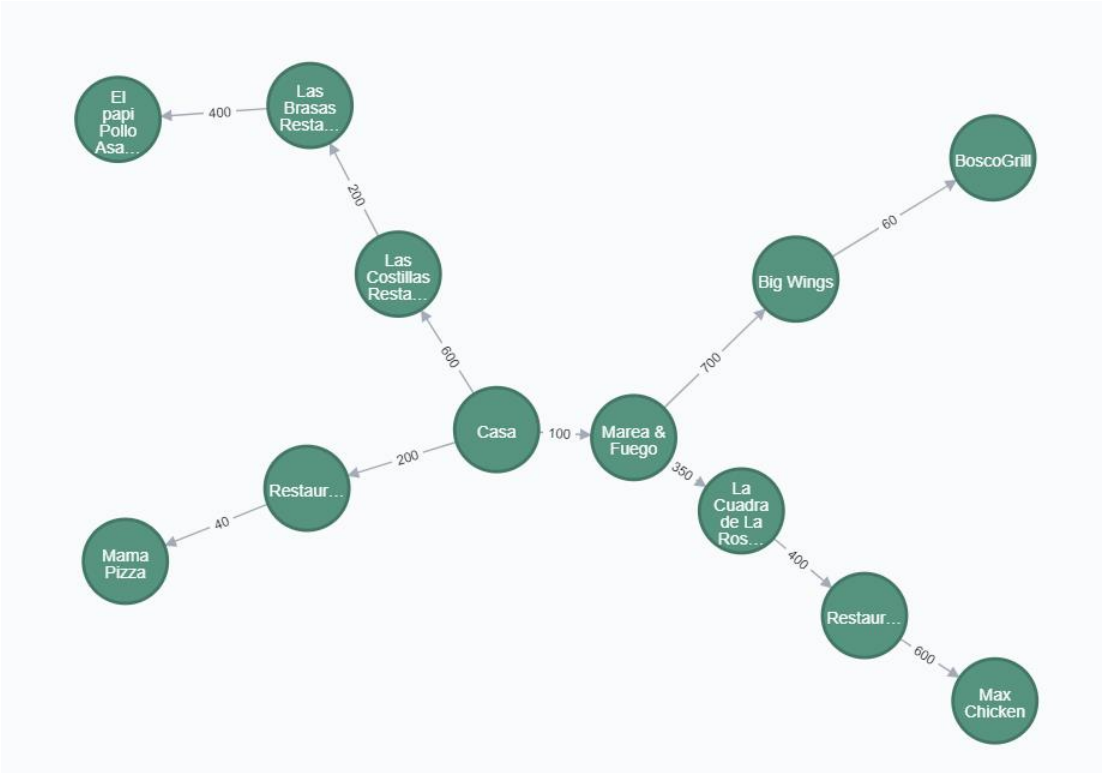
ID	NODO	COSTO
A	Casa	0
B	Restaurante Rancho Dorado 3	200
C	Marea & Fuego	100
D	Las Costillas Restaurante	600
E	Mama Pizza	40
F	Big Wings	700
G	La Cuadra de La Rosita	350
H	Las Brasas Restaurante	60
I	BoscoGrill	400
J	Restaurante La Carpa	600
K	Max Chicken	200
L	El papi Pollo Asadero Restaurant	400

Algoritmo A* a mano



Algoritmo HC En Neo4j

```
CREATE
(A:Restaurante{name:'Casa'}),
(B:Restaurante{name:'Restaurante Rancho Dorado 3'}),
(C:Restaurante{name:'Marea & Fuego'}),
(D:Restaurante{name:'Las Costillas Restaurante'}),
(E:Restaurante{name:'Mama Pizza'}),
(F:Restaurante{name:'Big Wings'}),
(G:Restaurante{name:'La Cuadra de La Rosita'}),
(H:Restaurante{name:'Las Brasas Restaurante'}),
(I:Restaurante{name:'BoscoGrill'}),
(J:Restaurante{name:'Restaurante La Carpa'}),
(K:Restaurante{name:'Max Chicken'}),
(L:Restaurante{name:'El papi Pollo Asadero Restaurant'}),
(A)-[:RUTA { cost: 200}]->(B),
(A)-[:RUTA { cost: 100}]->(C),
(A)-[:RUTA { cost: 600}]->(D),
(B)-[:RUTA { cost: 40}]->(E),
(C)-[:RUTA { cost: 700}]->(F),
(C)-[:RUTA { cost: 350}]->(G),
(F)-[:RUTA { cost: 60}]->(I),
(G)-[:RUTA { cost: 400}]->(J),
(J)-[:RUTA { cost: 600}]->(K),
(D)-[:RUTA { cost: 200}]->(H),
(H)-[:RUTA { cost: 400}]->(L)
```



```
MATCH (start:Restaurante {name: "Casa"}), (end:Restaurante {name: "El papi Pollo Asadero Restaurant"})
CALL gds.alpha.shortestPath.stream({
  nodeProjection: 'Restaurante',
  relationshipProjection: {
  CAMINO: {
  type: 'RUTA',
  properties: 'cost',
  orientation:'UNDIRECTED'
  }
  },
  startNode: start,
  endNode: end,
  relationshipWeightProperty: 'cost'
  })
  YIELD nodeId, cost
  RETURN gds.util.asNode(nodeId).name AS NOMBRE, cost
```

Conclusión:

Mediante el uso de cos podemos tener una mayor precisión acerca del mejor camino ayudando a encontrar el camino más corto de sin gastar mucha memoria al abrir todos los nodos, podemos probar rutas cortas que no llevan al destino deseado.