

UNIVERSIDAD POLITECNICA SALESIANA

Nombre: Luis Orellana P.

Metodos de Busqueda Sin Razonamiento

Busqueda por Amplitud



Ejercicio No 1: Búsqueda por anchura

Diseñe un grafo similar al que se ha presentado en este ejercicio partiendo de las siguientes coordenadas de latitud y longitud: **-2.8801604,-79.0071712**. Para ello deberá realizar las siguientes tareas:

- Emplear la herramienta Google Maps (R) con las coordenadas antes indicadas ([Link](#)).
- Definir 11 puntos de interés (El Vecino, Bellavista, Loja Argelia, Misicata, etc.) y armar el grafo.
- Especificar como punto de partida al sector "San Sebastián" y como objetivo "Totoracocha".
- Establecer los arcos o caminos en 1 sola dirección, por ejemplo, del nodo "Bellavista" al nodo "Loja Argelia".
- Realizar el proceso de búsqueda de forma similar a cómo se a explicado en este apartado, almacenando para ello los datos de la lista **Visitados** y de la **Cola**.

El trabajo deberá desarrollarse de forma manual en el cuaderno.



Ejercicio No 2: Búsqueda por costo uniforme

Diseñe un grafo similar al que se ha presentado en este ejercicio partiendo de las siguientes coordenadas de latitud y longitud: **-2.8801604,-79.0071712**. Para ello deberá realizar las siguientes tareas:

- Emplear la herramienta Google Maps (R) con las coordenadas antes indicadas ([Link](#)).
- Definir 11 puntos de interés (El Vecino, Bellavista, Loja Argelia, Misicata, etc.) y armar el grafo.
- Especificar como punto de partida al sector "San Sebastián" y como objetivo "Totoracocha".
- Establecer los arcos o caminos en 1 sola dirección, por ejemplo, del nodo "Bellavista" al nodo "Loja Argelia".
- Calcular la distancia que existe entre los puntos de interés. Para ello puede usar la herramienta de medida (click con el botón derecho del ratón y seleccionar la opción "Medir").
- Realizar el proceso de búsqueda de forma similar a cómo se a explicado en este apartado, almacenando para ello los datos de la lista **Visitados** y de la **Cola**.

El trabajo deberá desarrollarse de forma manual en el cuade

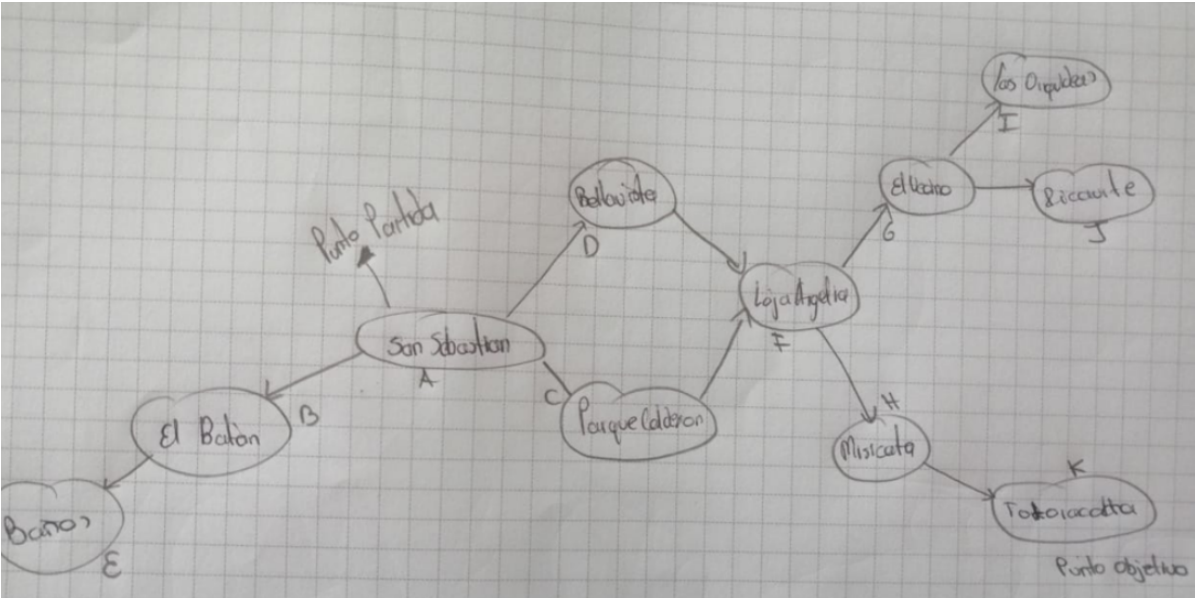
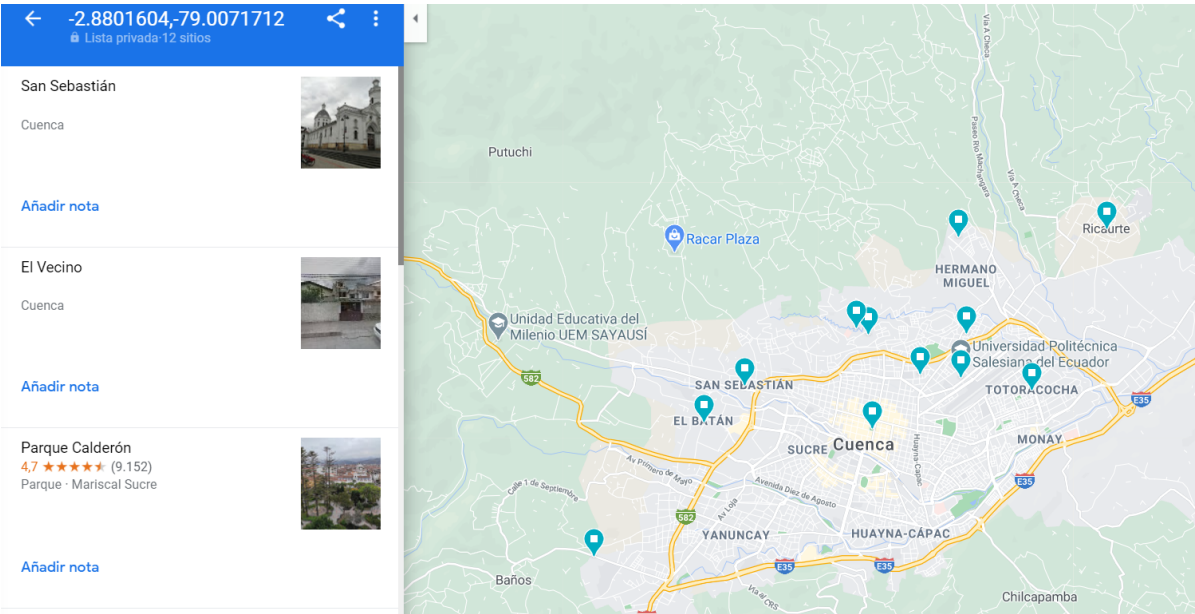


Ejercicio No 2: Búsqueda por profundidad

Diseñe un grafo similar al que se ha presentado en este ejercicio partiendo de las siguientes coordenadas de latitud y longitud: **-2.8801604,-79.0071712**. Para ello deberá realizar las siguientes tareas:

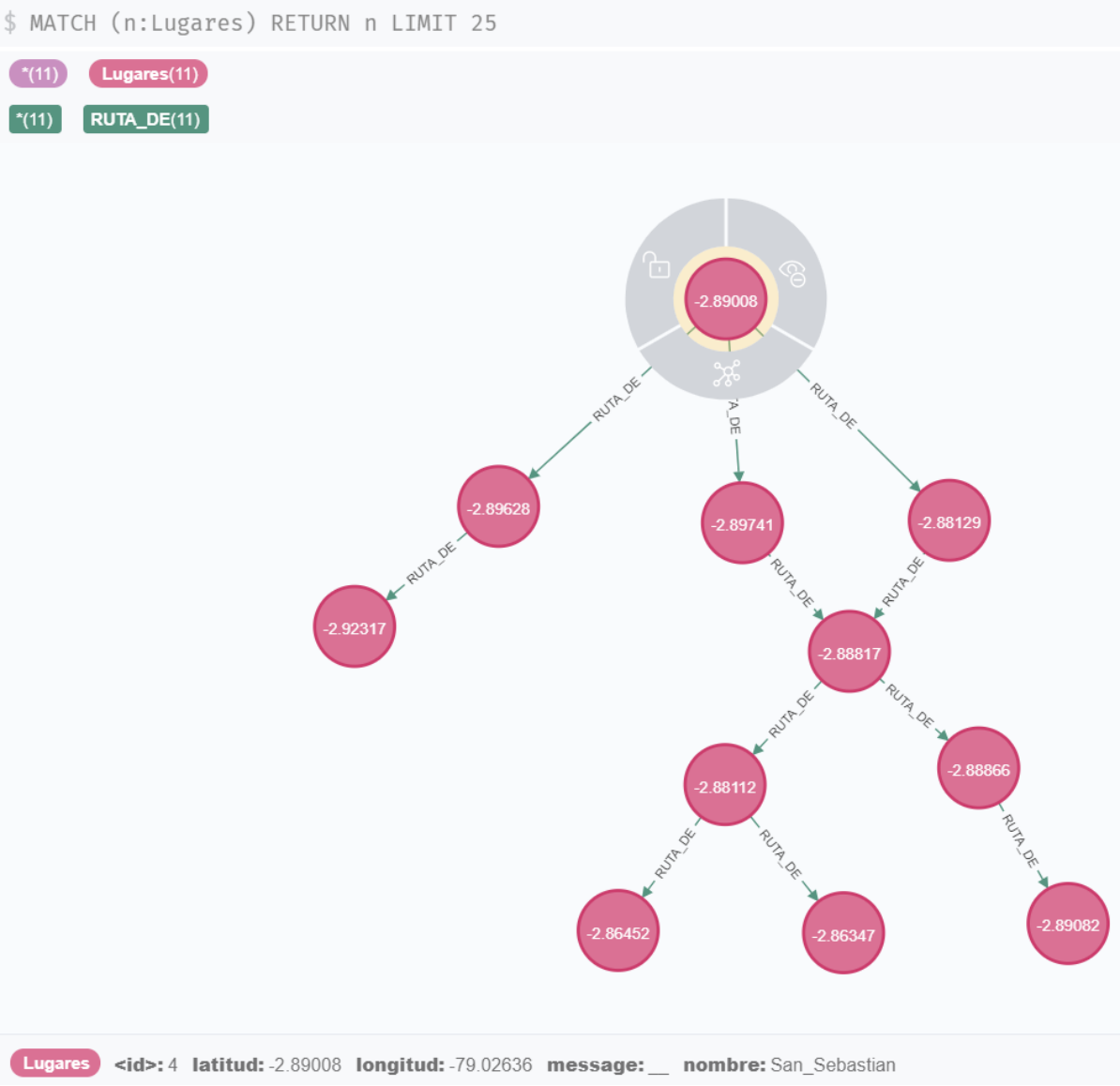
- Emplear la herramienta Google Maps (R) con las coordenadas antes indicadas ([Link](#)).
- Definir 11 puntos de interés (El Vecino, Bellavista, Loja Argelia, Misicata, etc.) y armar el grafo.
- Especificar como punto de partida al sector "San Sebastián" y como objetivo "Totoracocha".
- Establecer los arcos o caminos en 1 sola dirección, por ejemplo, del nodo "Bellavista" al nodo "Loja Argelia".
- Calcular la distancia que existe entre los puntos de interés. Para ello puede usar la herramienta de medida (click con el botón derecho del ratón y seleccionar la opción "Medir").
- Realizar el proceso de búsqueda de forma similar a cómo se a explicado en este apartado, almacenando para ello los datos de la lista **Visitados** y de la **Cola**.

El trabajo deberá desarrollarse de forma manual en el cuaderno.



Origen	Destino	H(h)	G(n)
A	B	7,03 Km	2,5 Km
A	C	2,57 Km	2,9 Km
A	D	2,56 Km	2,9 Km
B	E	4,68 Km	7,9 Km
C	F	1,38 Km	1,9 Km
D	F	1,26 Km	2,4 Km
F	G	1,18 Km	1,5 Km
F	H	0,76985 Km	1,9 Km
G	I	185 Km	2,4 Km
G	J	3,22 Km	4,7 Km
H	K	1,39 Km	2,8 Km

Arbol de nodos generados



Algoritmos

Busqueda por Anchura

```
neo4j$ CALL gds.graph.create('myGraph', 'Lugares', 'RUTA_DE', {
  relationshipProperties: 'costo' })
```

```
neo4j$ CALL gds.graph.create('myGraph', 'Lugares', 'RUTA...
```

table

1

A

ext

ode

nodeProjection	relationshipProjection	graphName	nodeCount	relati
<pre>{ "Lugares": { "properties": { }, "label": "Lugares" } }</pre>	<pre>{ "RUTA_DE": { "orientation": "NATURAL", "aggregation": "DEFAULT", "type": "RUTA_DE", "properties": { "costo": { "property": "costo", "aggregation": "DEFAULT", "defaultValue":</pre>	"myGraph"	11	11

```

1 MATCH (San_Sebastian:Lugares{nombre:'San_Sebastian'}),
  (Totoracocha:Lugares{nombre:'Totoracocha'})
2 WITH id(San_Sebastian) AS startNode, [id(Totoracocha)] AS
  targetNodes
3 CALL gds.alpha.bfs.stream('myGraph', {startNode: startNode,
  targetNodes: targetNodes})
4 YIELD path
5 UNWIND [ n in nodes(path) | n.nombre ] AS tags
6 RETURN tags

```

tags	
1	"San_Sebastian"
2	"Bellavista"
3	"Parque_Calderon"
4	"El_Batan"
5	"Loja_Argelia"
6	"Baños"
7	"El_Vecino"

started streaming 11 records in less than 1 ms and completed after 2 ms

Busqueda coste

```

MATCH (source:Lugares {nombre: 'San_Sebastian'}), (target:Lugares
{nombre: 'Totoracocha'})
CALL gds.beta.shortestPath.yens.stream('myGraphs', {
  sourceNode: id(source),
  targetNode: id(target),
  k: 3,
  relationshipWeightProperty: 'costo'
})
YIELD index, sourceNode, targetNode, totalCost, nodeIds, costs
RETURN
  index,
  gds.util.asNode(sourceNode).nombre AS sourceNodeName,

```

1	"San_Sebastian"	"Totoracocha"	6.10985	["San_Sebastian", "Parque_Calderon", "Loja_Argelia", "Misicata", "Totoracocha"]	[0.0, 2.57, 3.9499999999999997, 4.71985, 6.10985]
---	-----------------	---------------	---------	--	---

Siendo la ruta:

San_Sebastian, Parque Calderon, Loja Argelia, Misicata, Totoracocha

Busqueda por Profundidad

```
MATCH (San_Sebastian:Lugares{nombre:'San_Sebastian'}),
(Totoracocha:Lugares{nombre:'Totoracocha'})
WITH id(San_Sebastian) AS startNode, [id(Totoracocha)] AS
targetNodes
CALL gds.alpha.dfs.stream('myGraph', {startNode: startNode,
targetNodes: targetNodes})
YIELD path
UNWIND [ n in nodes(path) | n.nombre ] AS tags
RETURN tags
```

tags	
1	"San_Sebastian"
2	"El_Batan"
3	"Baños"
4	"Parque_Calderon"
5	"Loja_Argelia"
6	"Misicata"
7	"Totoracocha"

ted streaming 7 records after 1 ms and completed after 4 ms.

In []: