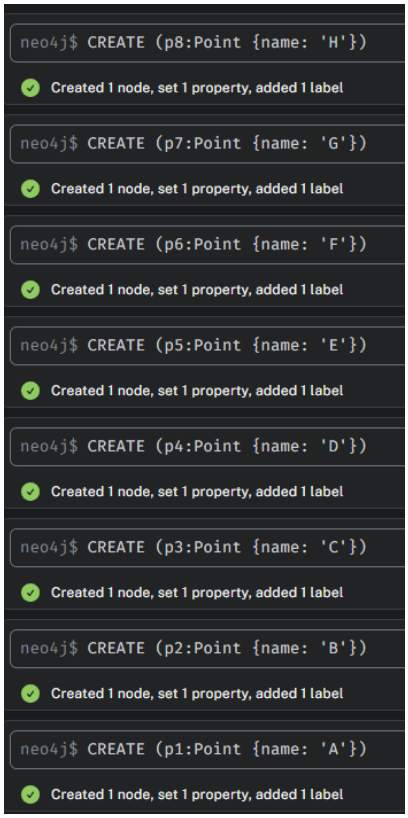
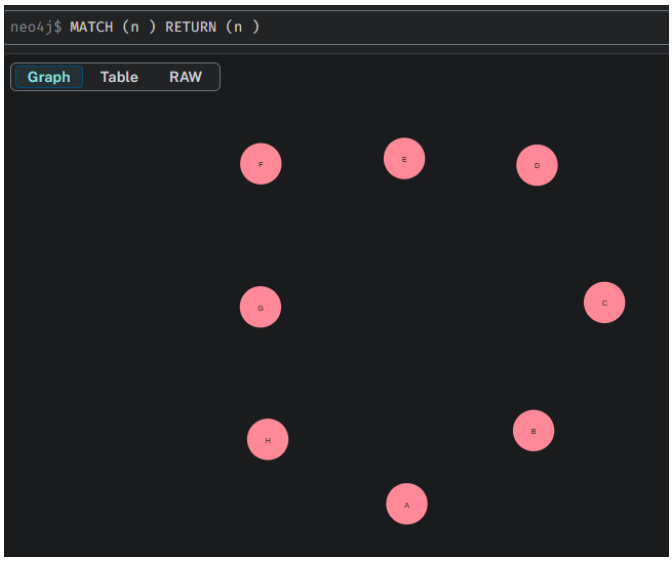
SOLUCIÓN TAREA 1-4

1. Creamos los nodos





2. Creamos las conexiones entre los nodos

```
neo4j$ MATCH (x:Point), (y:Point) WHERE x.name = 'H' AND y.name = 'D' CREATE (x )-[:CONNECTED {distance: 14}]->(y )
```

Created 1 relationship, set 1 property

03N90: Cartesian product

```
neo4j$ MATCH (x:Point), (y:Point) WHERE x.name = 'H' AND y.name = 'G' CREATE (x )-[:CONNECTED {distance: 3}]->(y )
```

Created 1 relationship, set 1 property

03N90: Cartesian product

```
neo4j$ MATCH (x:Point), (y:Point) WHERE x.name = 'H' AND y.name = 'B' CREATE (x )-[:CONNECTED {distance: 6}]->(y )
```

Created 1 relationship, set 1 property

03N90: Cartesian product

```
neo4j$ MATCH (x:Point), (y:Point) WHERE x.name = 'H' AND y.name = 'F' CREATE (x )-[:CONNECTED {distance: 9}]->(y )
```

Created 1 relationship, set 1 property

03N90: Cartesian product

```
neo4j$ MATCH (x:Point), (y:Point) WHERE x.name = 'H' AND y.name = 'A' CREATE (x )-[:CONNECTED {distance: 10}]->(y )
```

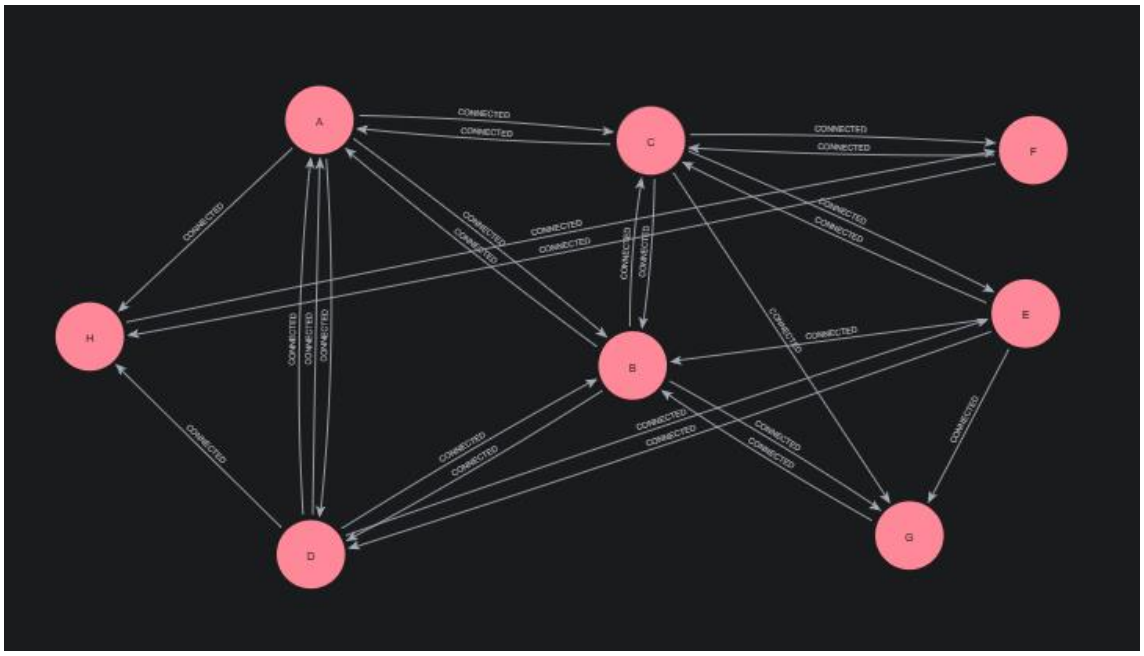
Created 1 relationship, set 1 property

03N90: Cartesian product

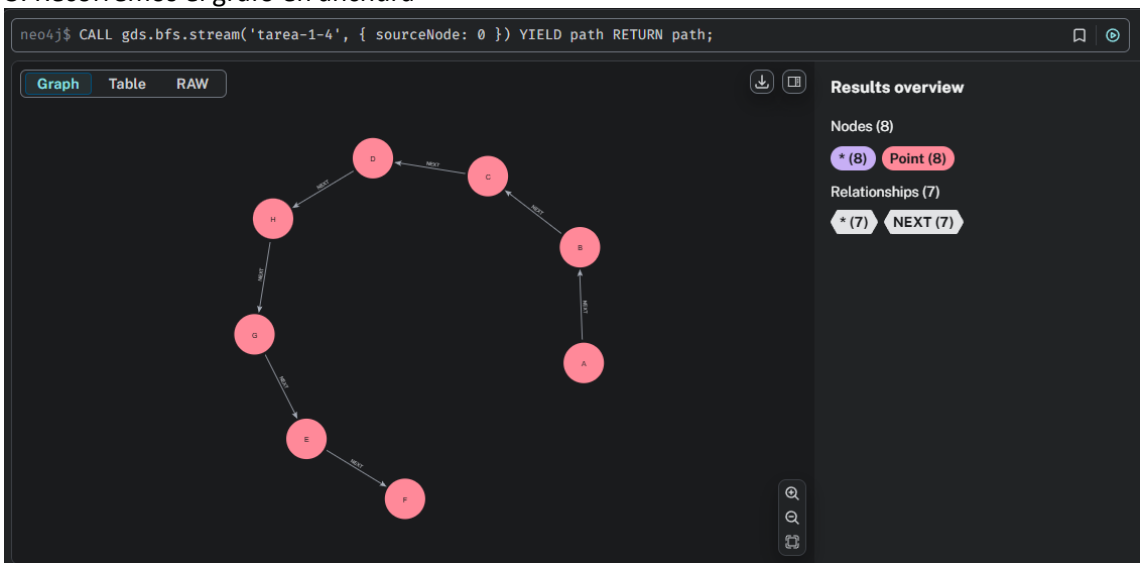
```
neo4j$ MATCH (x:Point), (y:Point) WHERE x.name = 'G' AND y.name = 'E' CREATE (x )-[:CONNECTED {distance: 15}]->(y )
```

Created 1 relationship, set 1 property

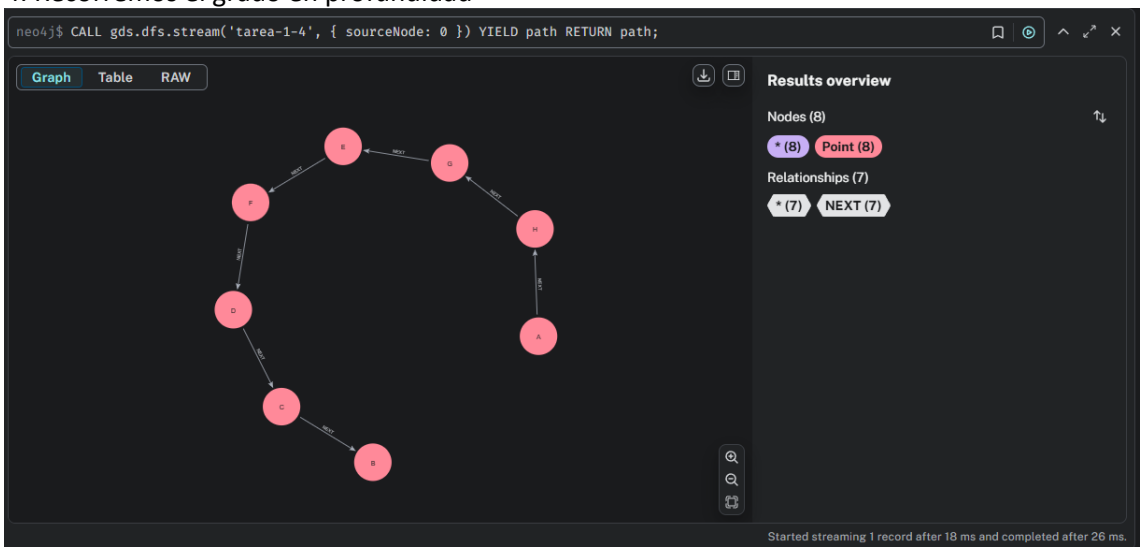
03N90: Cartesian product



3. Recorremos el grafo en anchura



4. Recorremos el grado en profundidad



5. Hayamos el camino mínimo entre H y G