

TSP – Traveling Salesman Problem

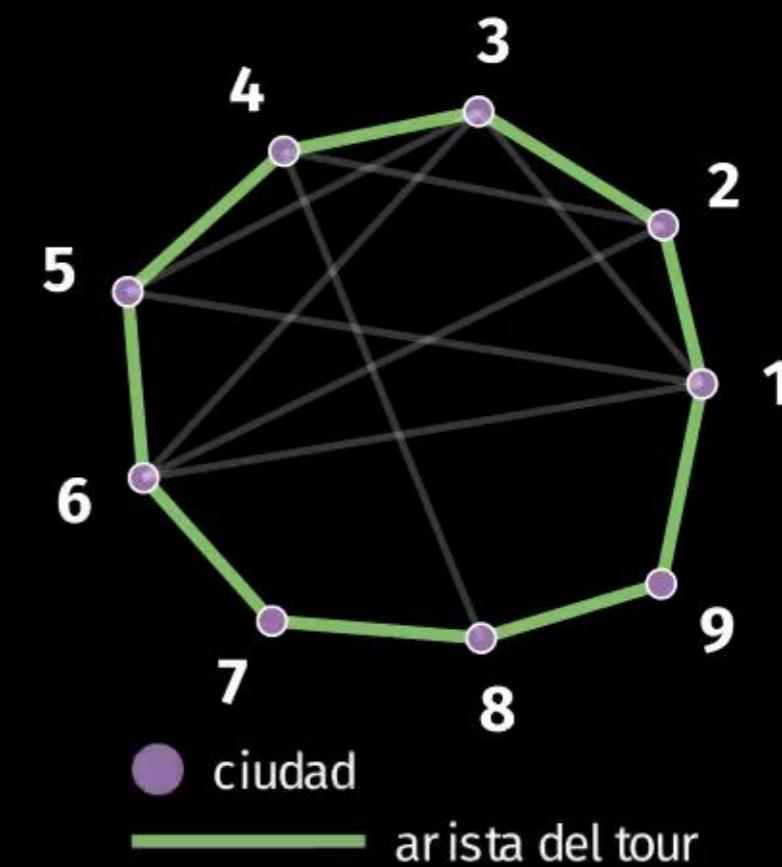
Problema:

Encontrar el tour de mínima longitud que visita cada ciudad una vez y regresa al inicio (grafo completo con distancias).

Intuición:

Un ciclo por todos los vértices; el costo = suma de distancias de sus aristas.

TSP – Traveling Salesman Problem



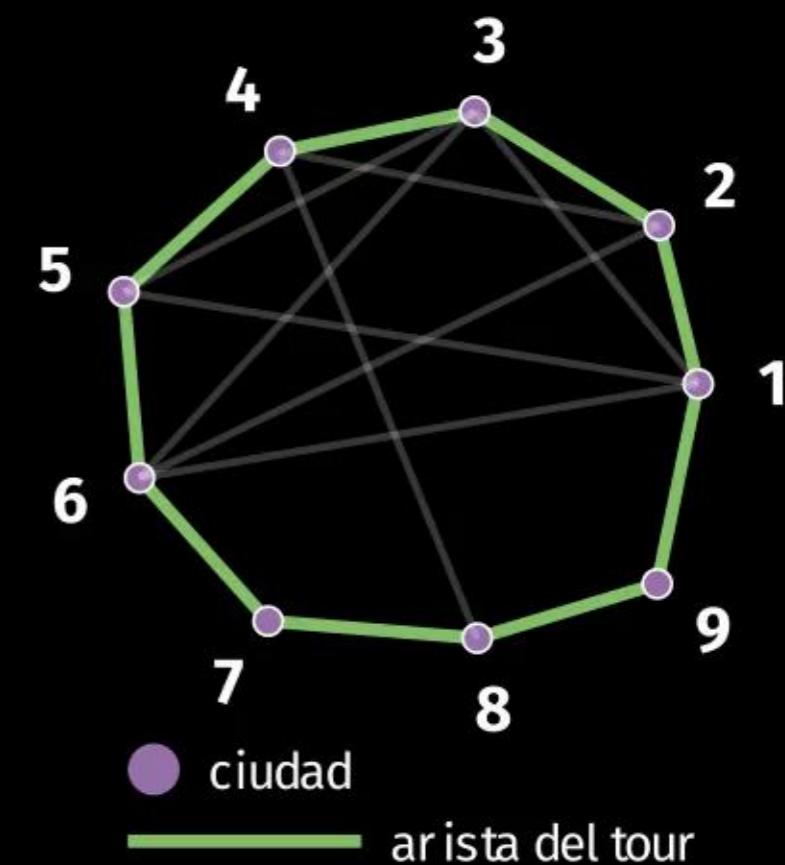
Problema:

Encontrar el tour de mínima longitud que visita cada ciudad una vez y regresa al inicio (grafo completo con distancias).

Intuición:

Un ciclo por todos los vértices; el costo = suma de distancias de sus aristas.

TSP – Traveling Salesman Problem



Problema:

Encontrar el tour de mínima longitud que visita cada ciudad una vez y regresa al inicio (grafo completo con distancias).

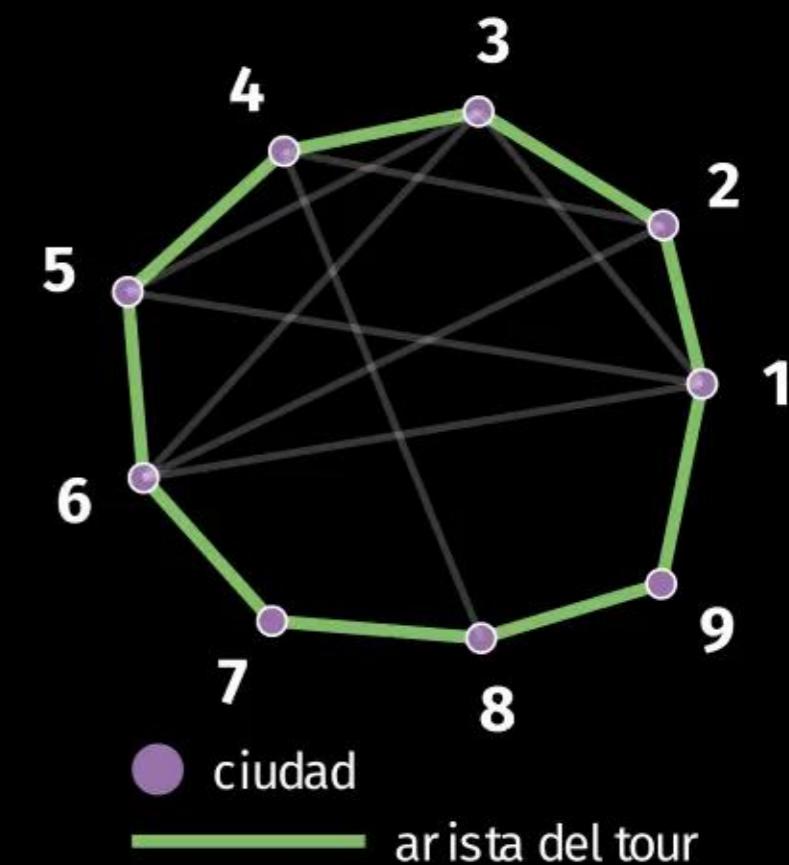
Intuición:

Un ciclo por todos los vértices; el costo = suma de distancias de sus aristas.

Formulación clásica (ILP):

$$\min \sum_{(i,j)} c_{ij} x_{ij}$$

TSP – Traveling Salesman Problem



Problema:

Encontrar el tour de mínima longitud que visita cada ciudad una vez y regresa al inicio (grafo completo con distancias).

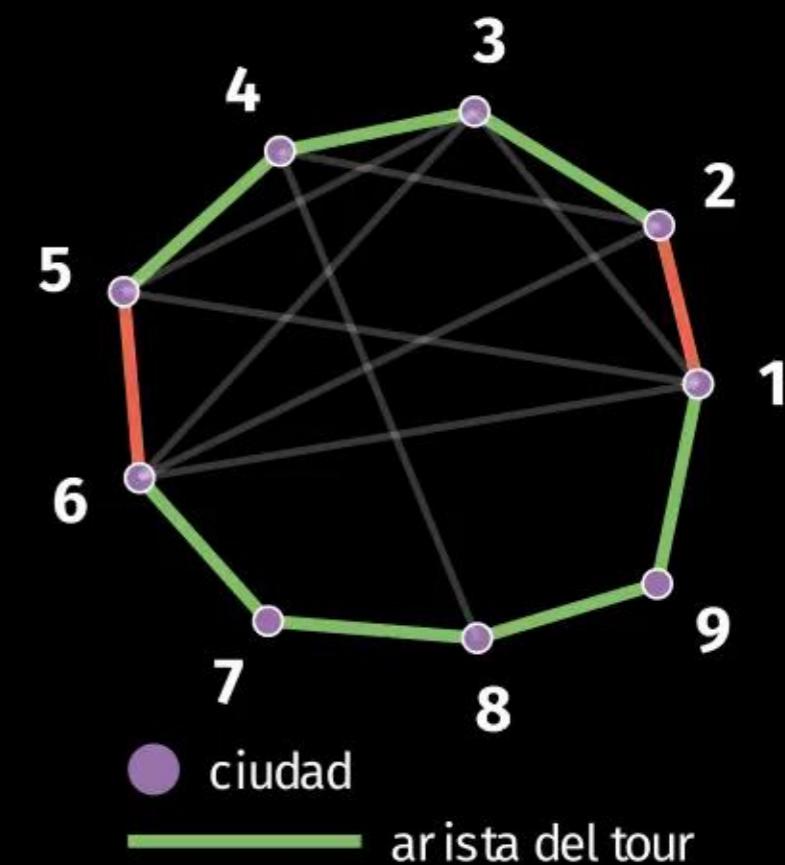
Intuición:

Un ciclo por todos los vértices; el costo = suma de distancias de sus aristas.

Formulación clásica (ILP):

$$\min \sum_{(i,j)} c_{ij} x_{ij}$$

TSP – Traveling Salesman Problem



Problema:

Encontrar el tour de mínima longitud que visita cada ciudad una vez y regresa al inicio (grafo completo con distancias).

Intuición:

Un ciclo por todos los vértices; el costo = suma de distancias de sus aristas.

Formulación clásica (ILP):

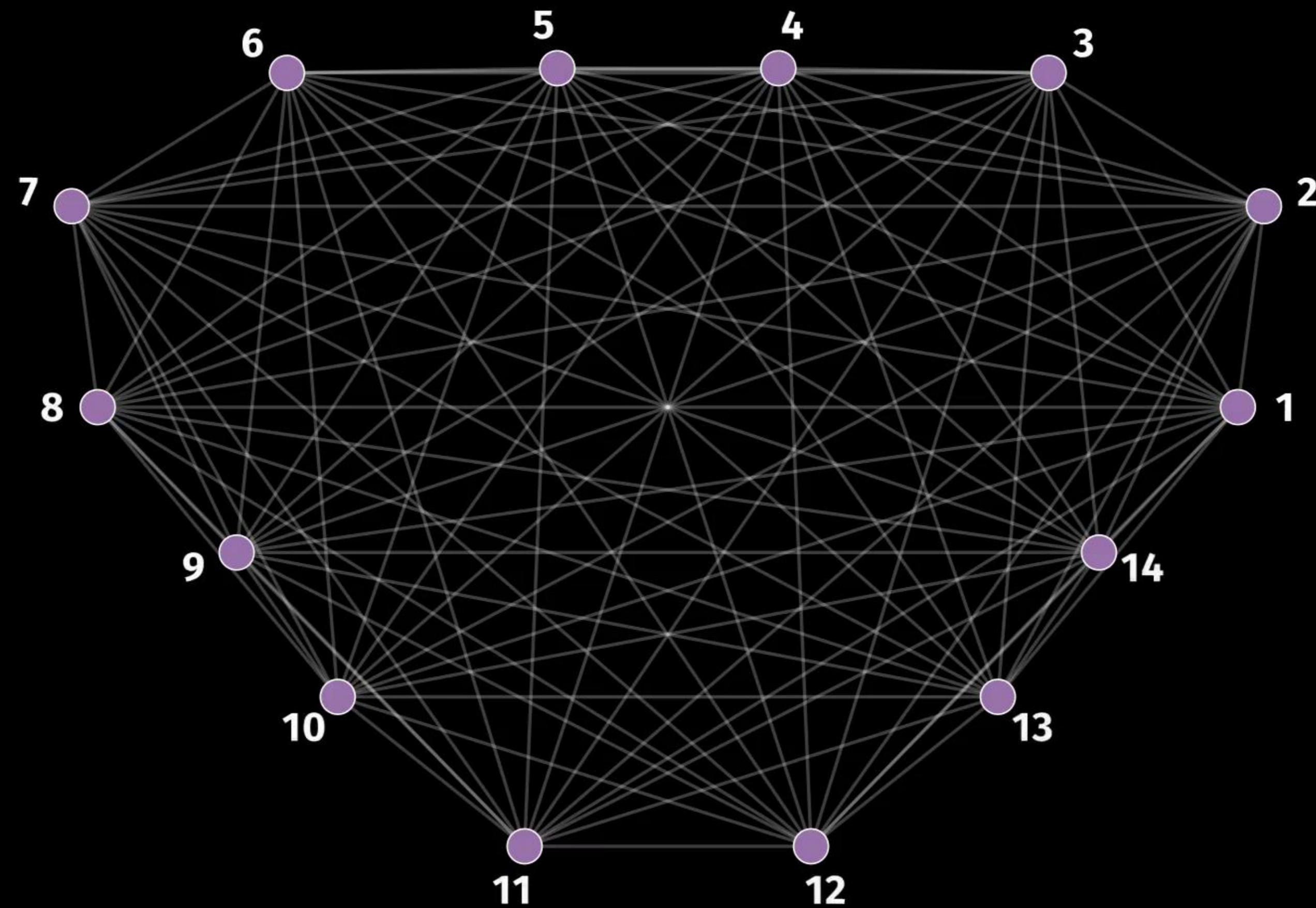
$$\min \sum_{(i,j)} c_{ij} x_{ij}$$

Construcción Golosa: TSP

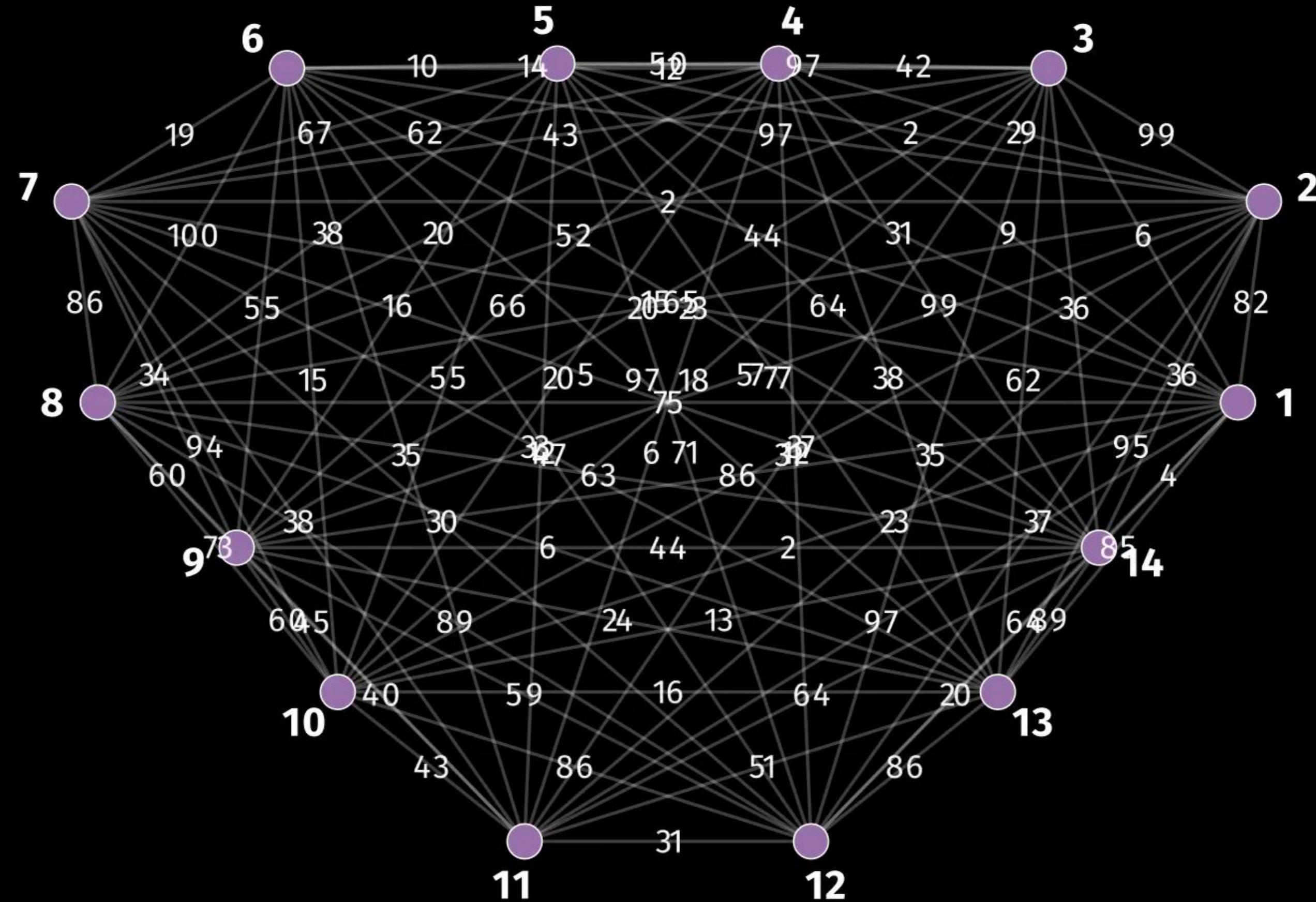
- Golosa (en arcos): $c(s, (jk)) = d_{jk}$
- Vecino más cercano: $c(s, e) = d_{ie}$
- Máximo arrepentimiento: $c(s, e) = \min_{j,k \in R} (d_{je} + d_{ek}) - \min_{j \in R} (d_{ie} - d_{ej})$
- Menor/Mayor inserción: $c(s, e) = \min_{j \in S} (d_{je} + d_{e,j+1} - d_{j,j+1})$

GRASP

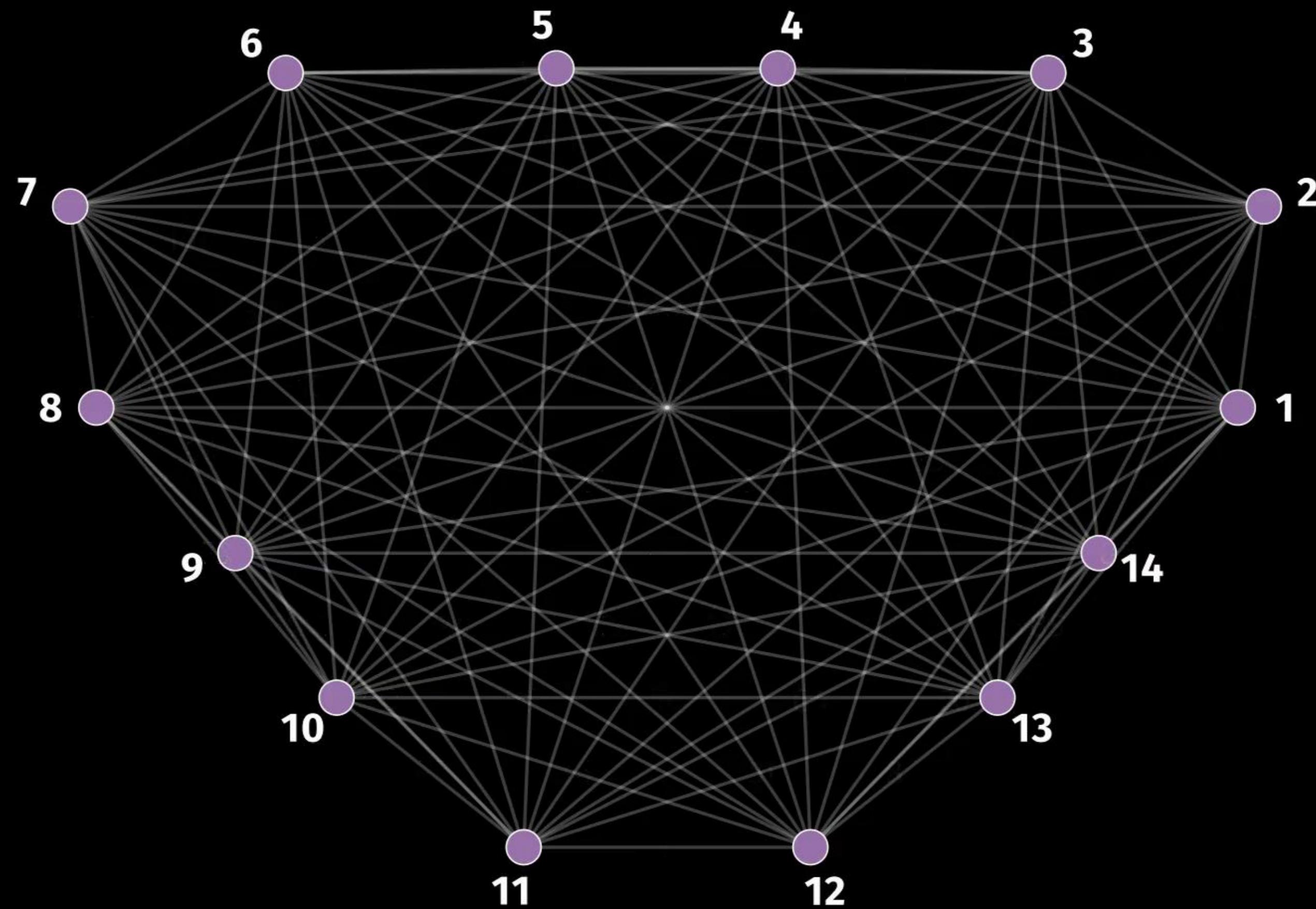
GRASP

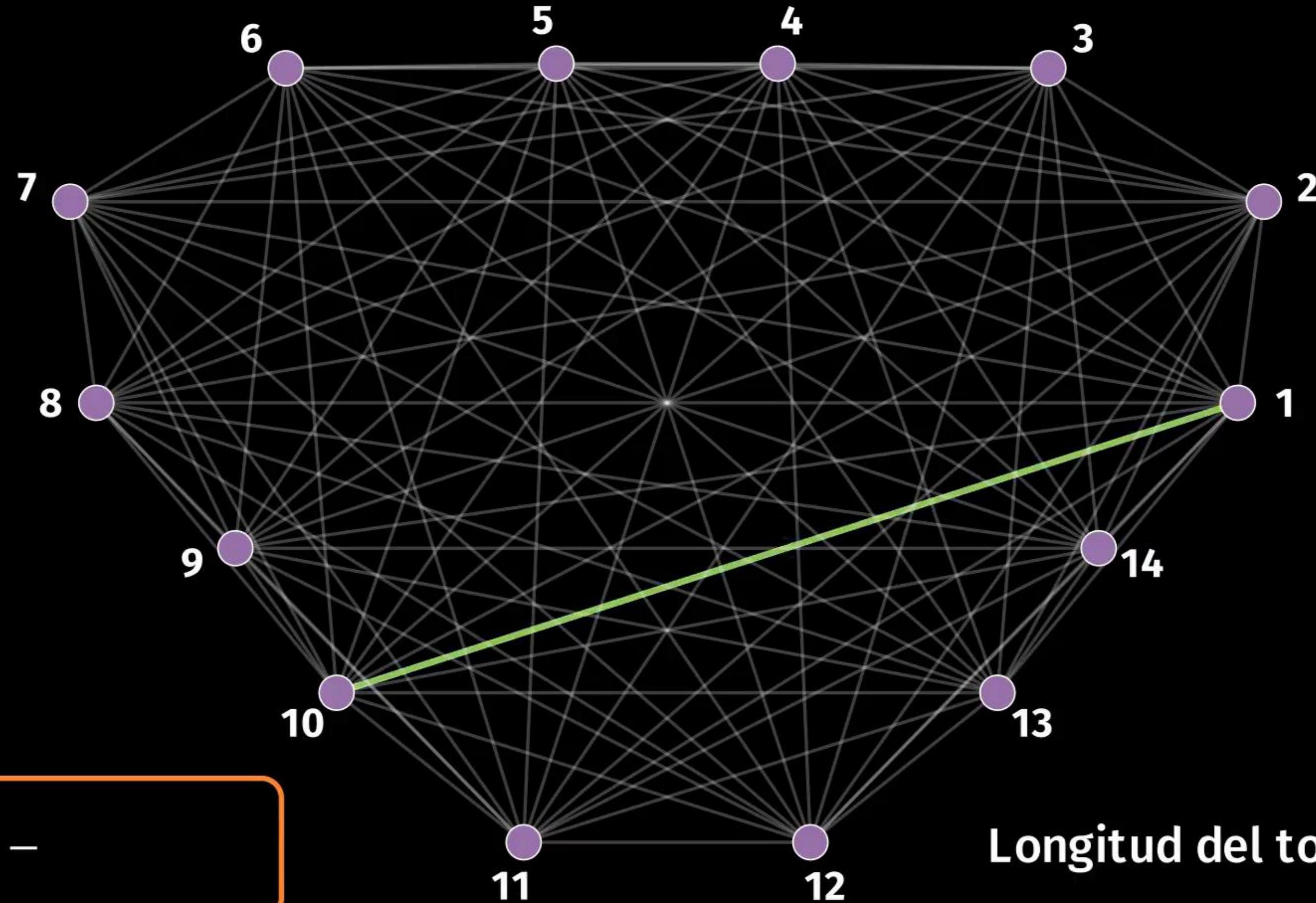


GRASP



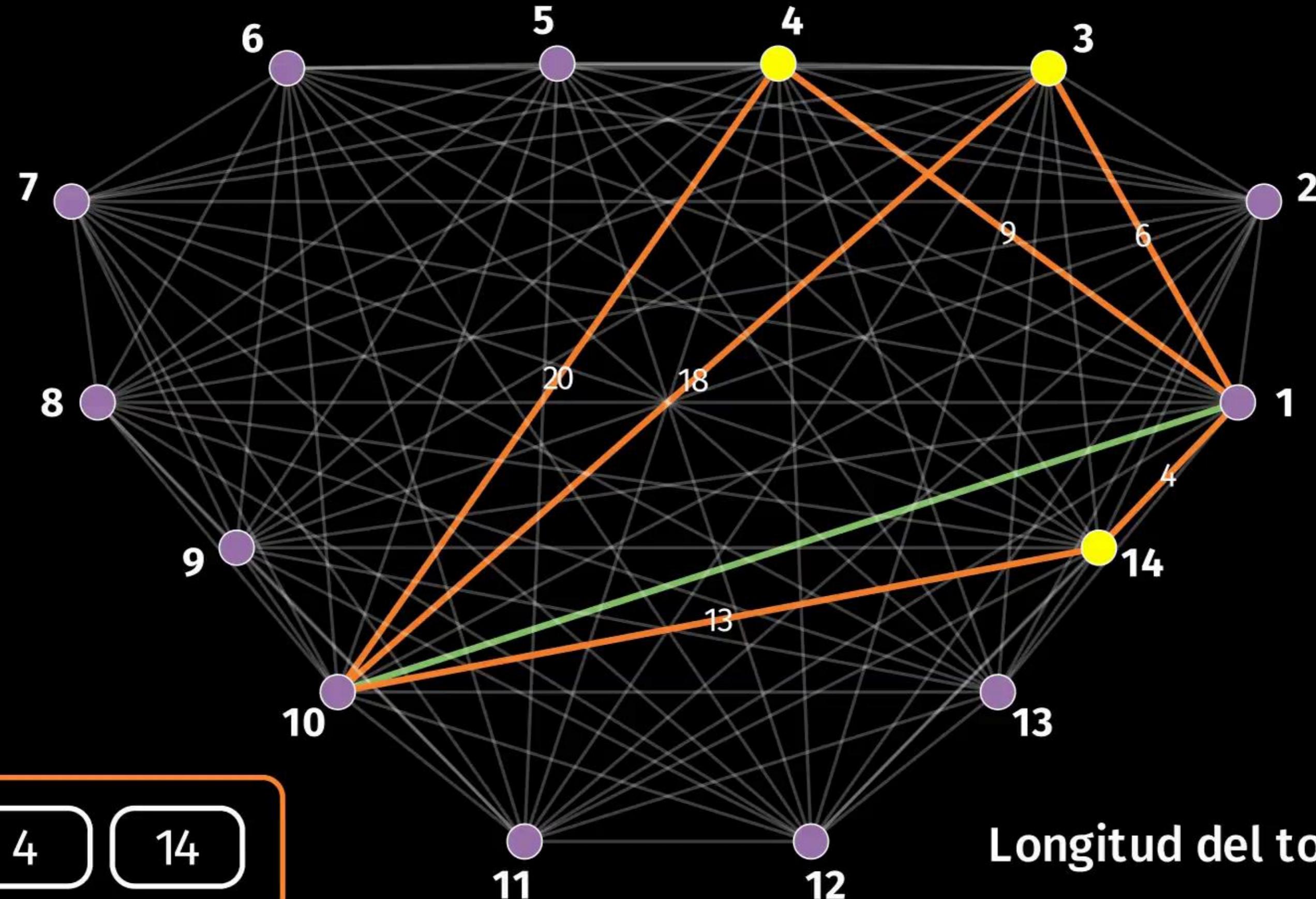
GRASP

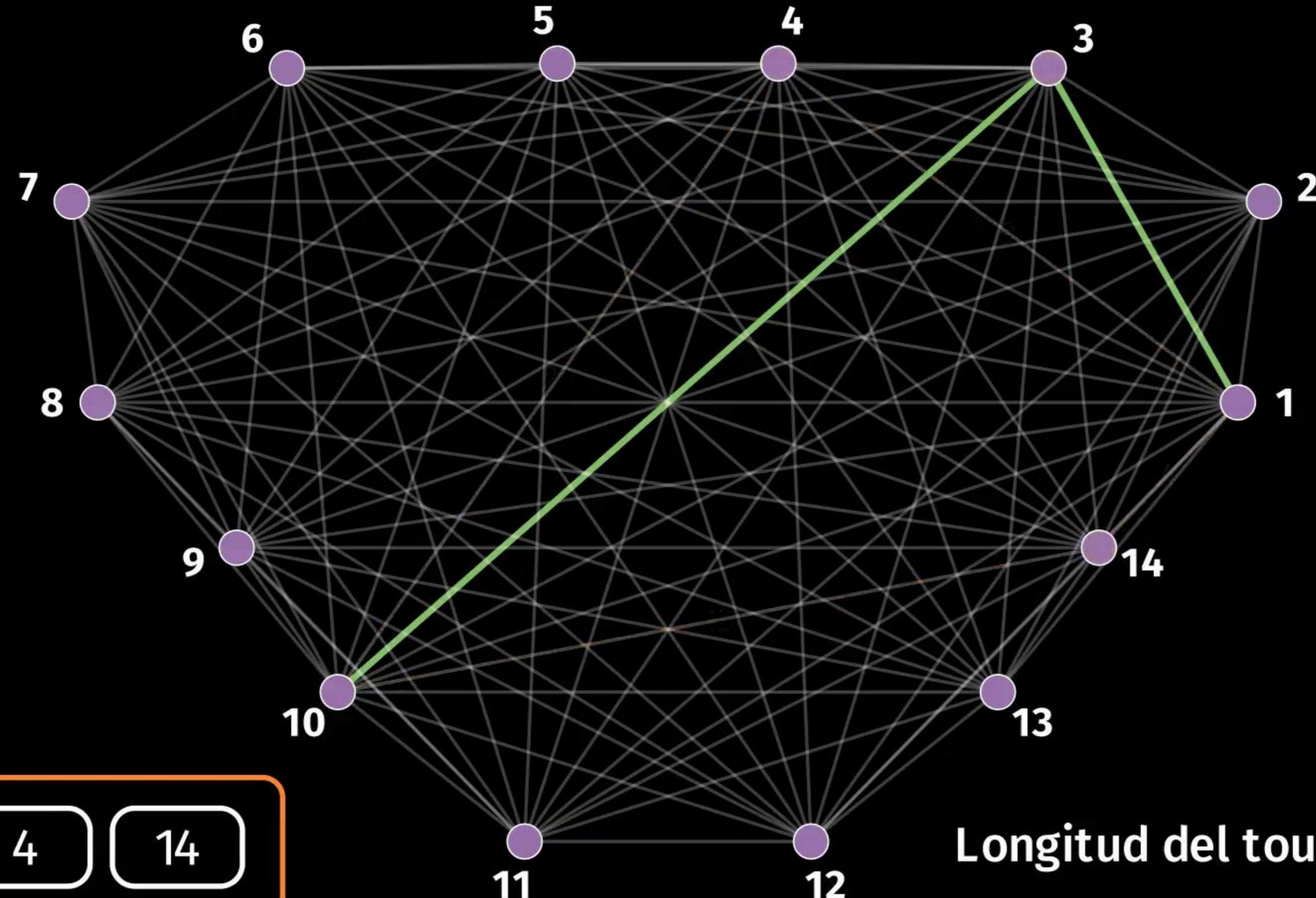




-

Longitud del tour: 2.0

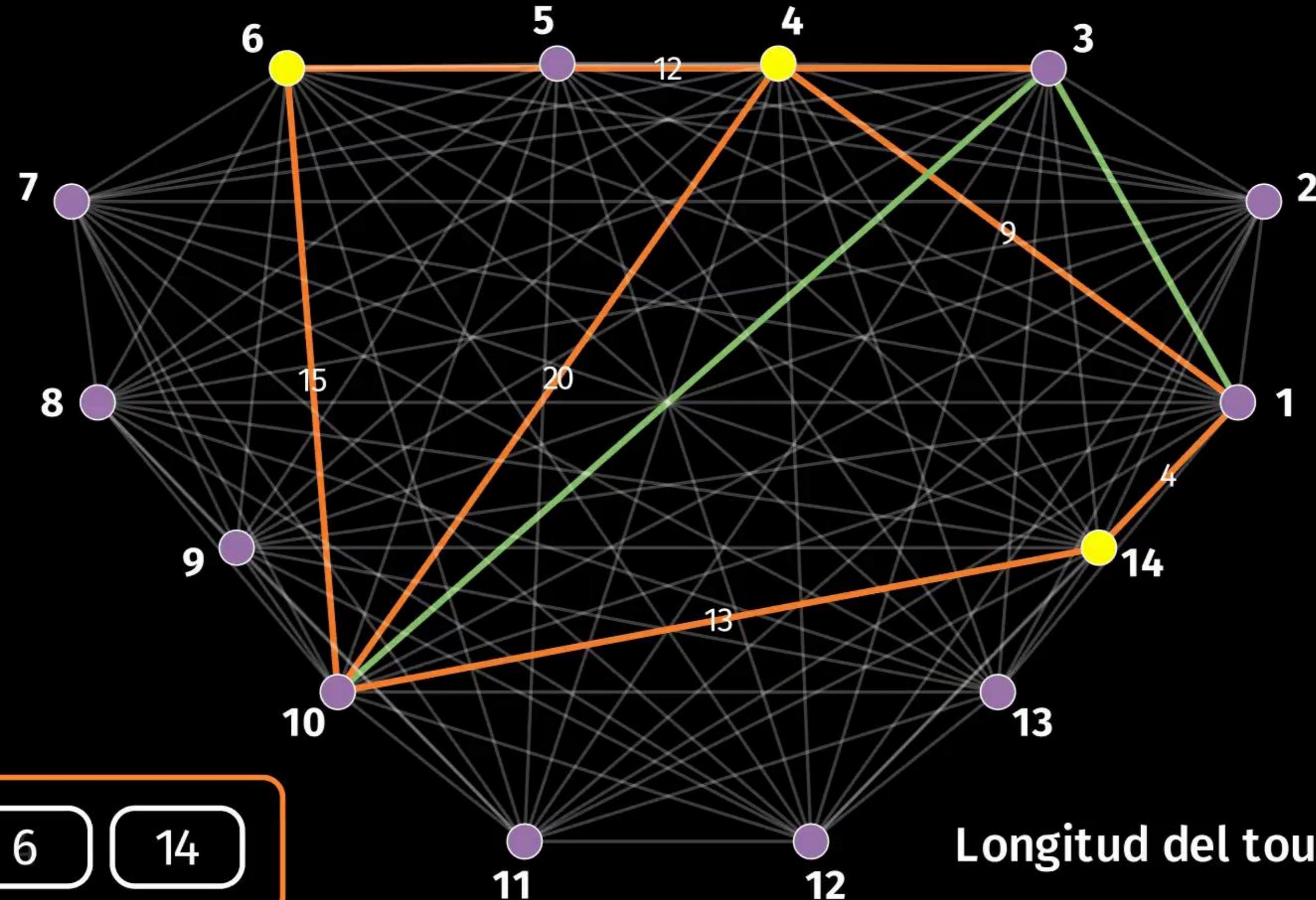




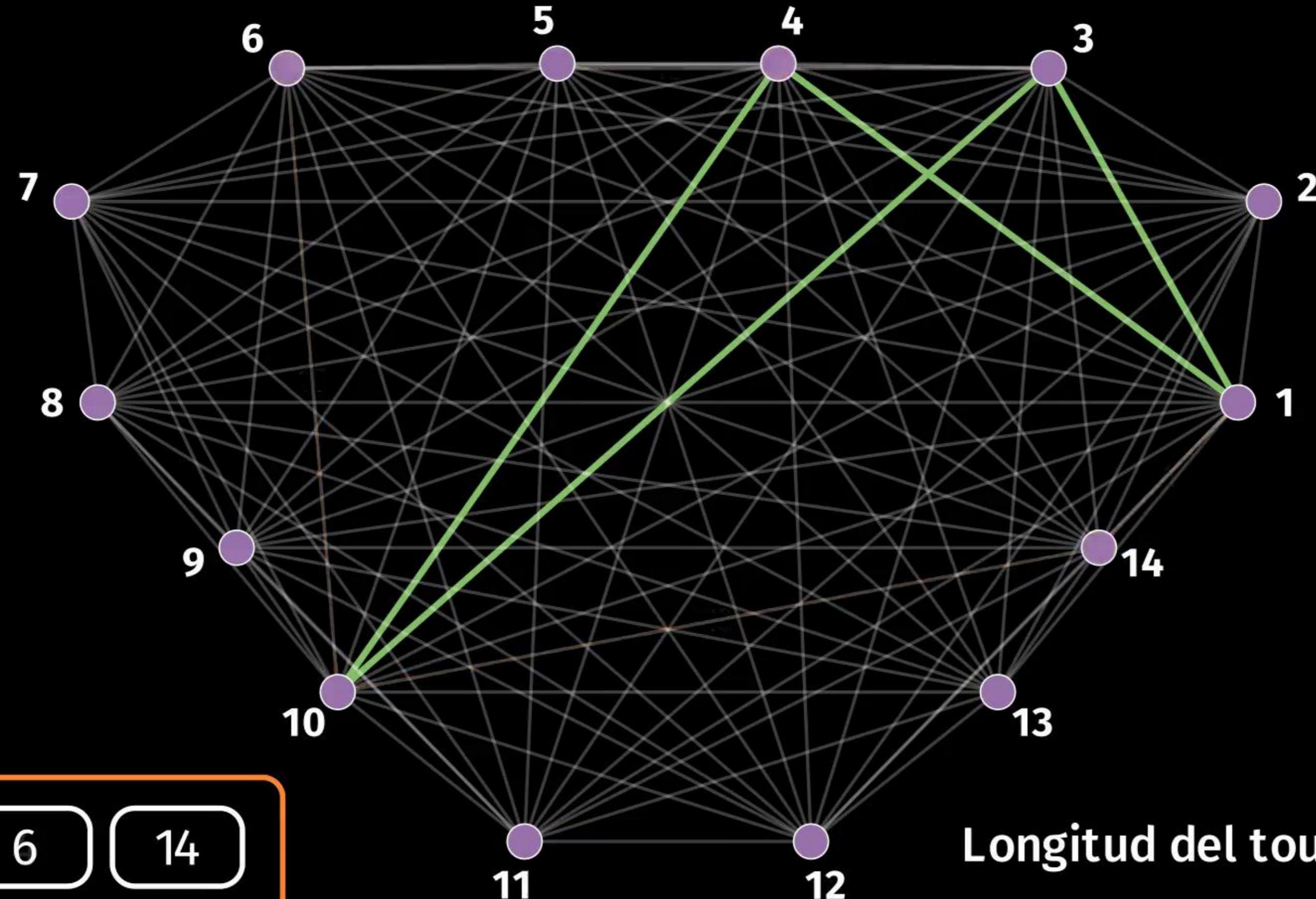
3

4

14



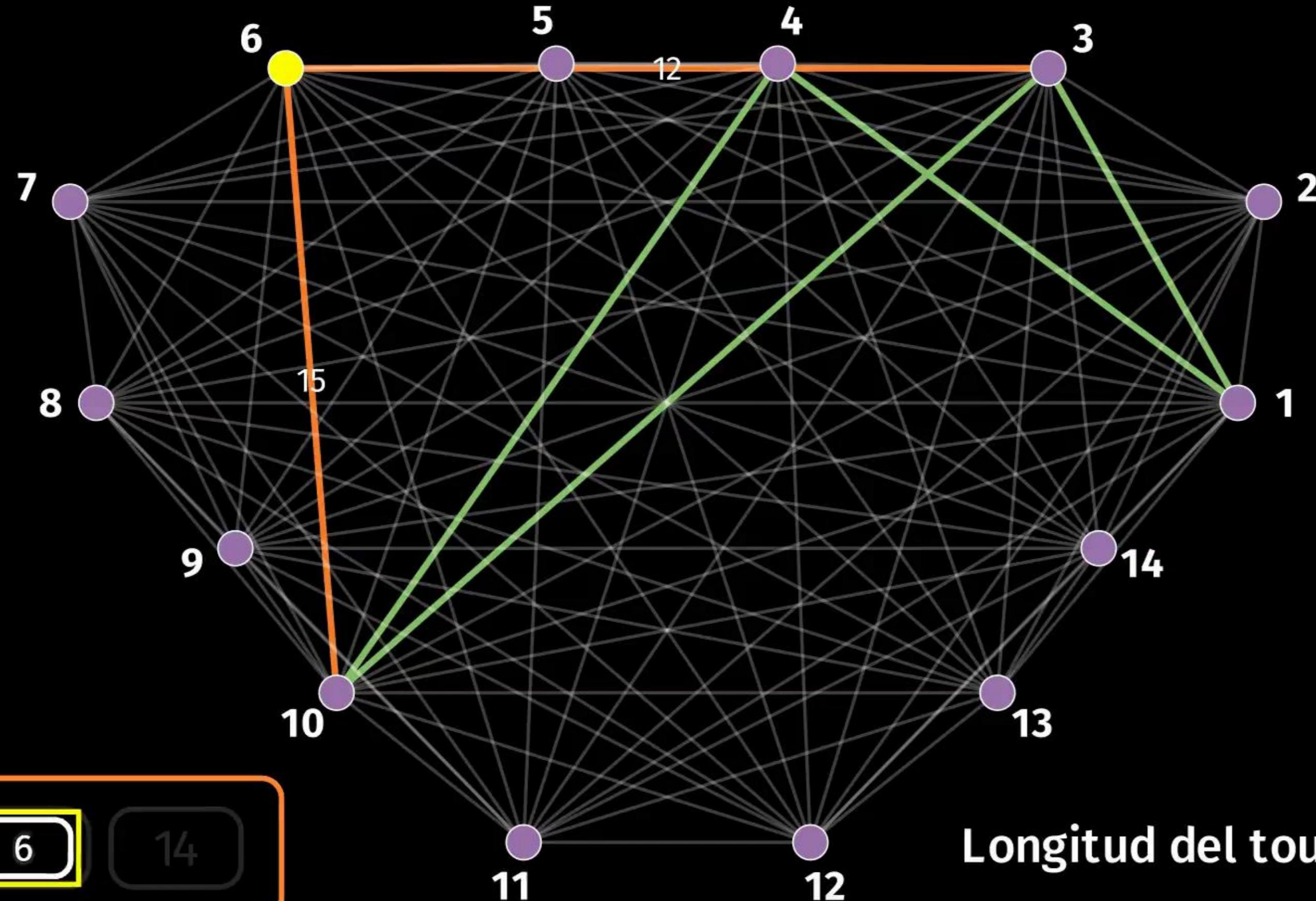
Longitud del tour: **24.0**



4

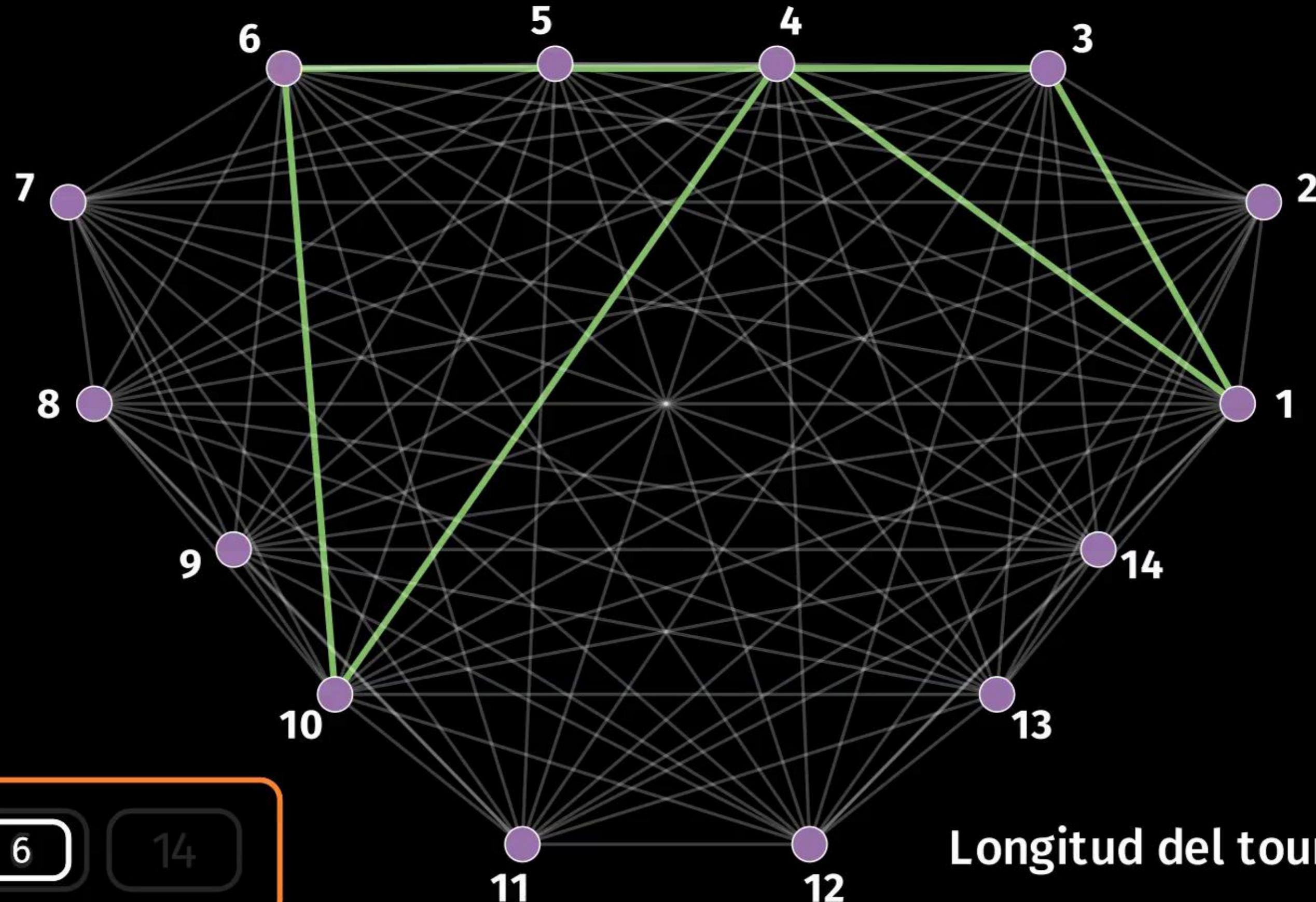
6

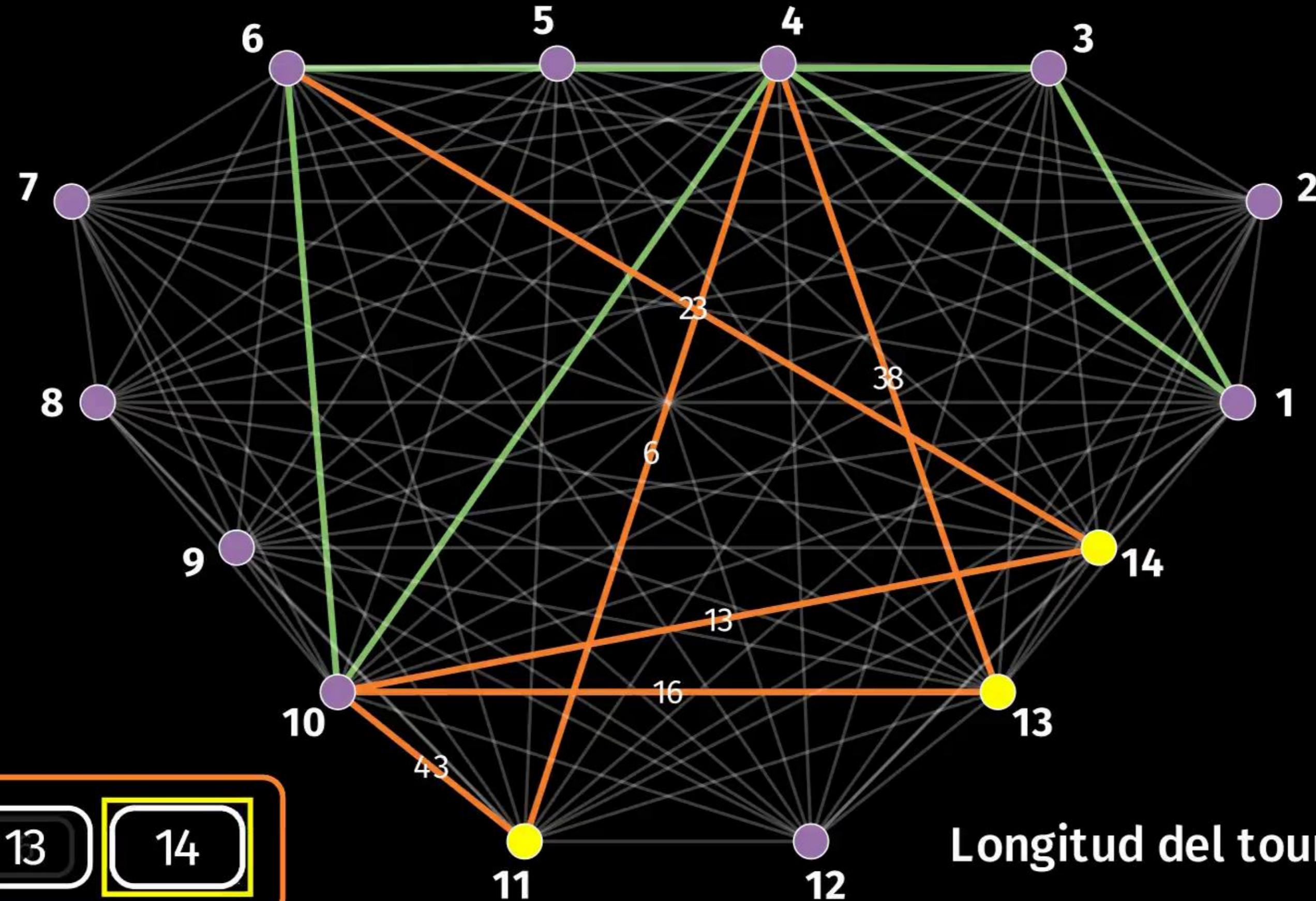
14



4 6 14

Longitud del tour: 51.0

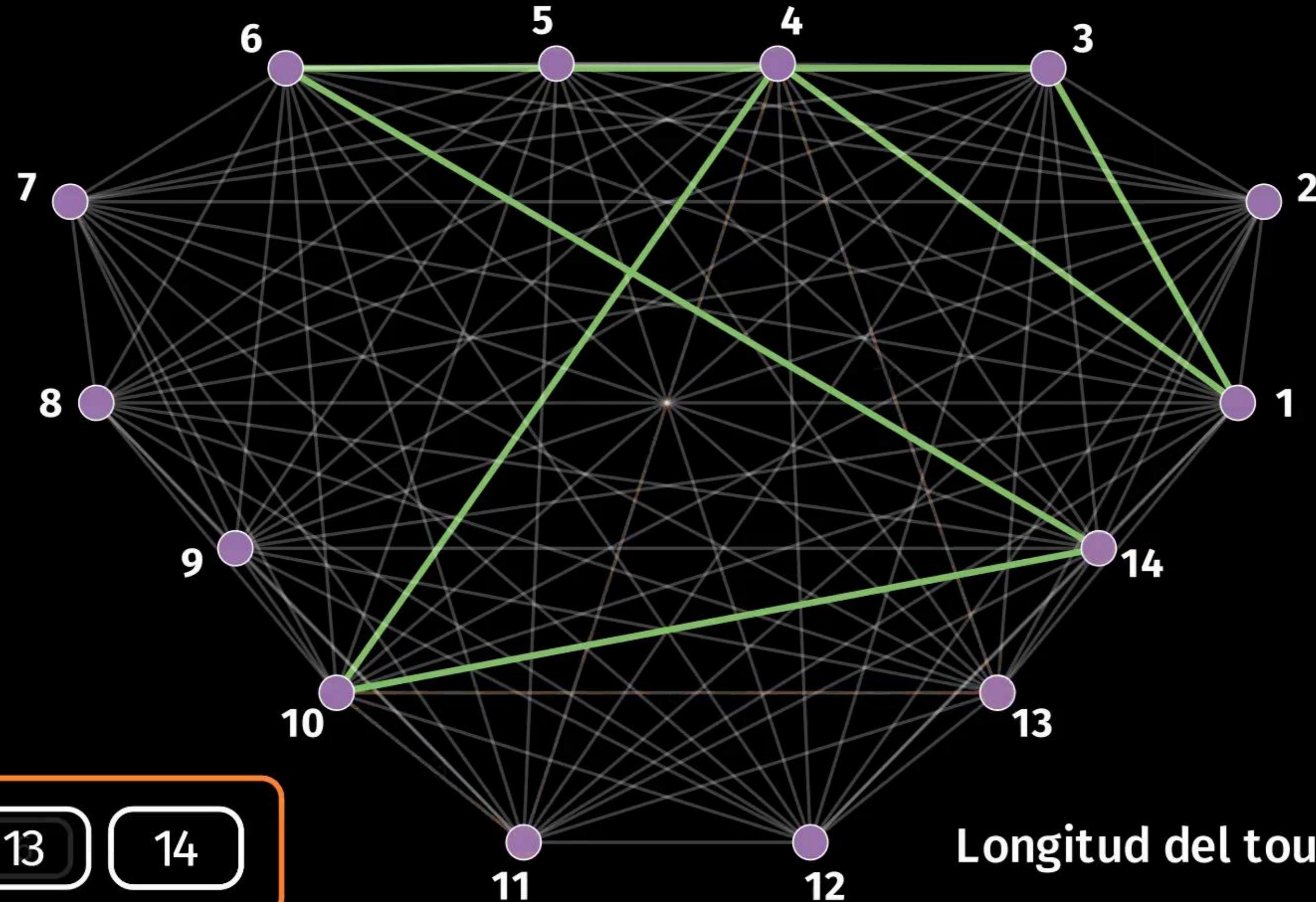




11

13

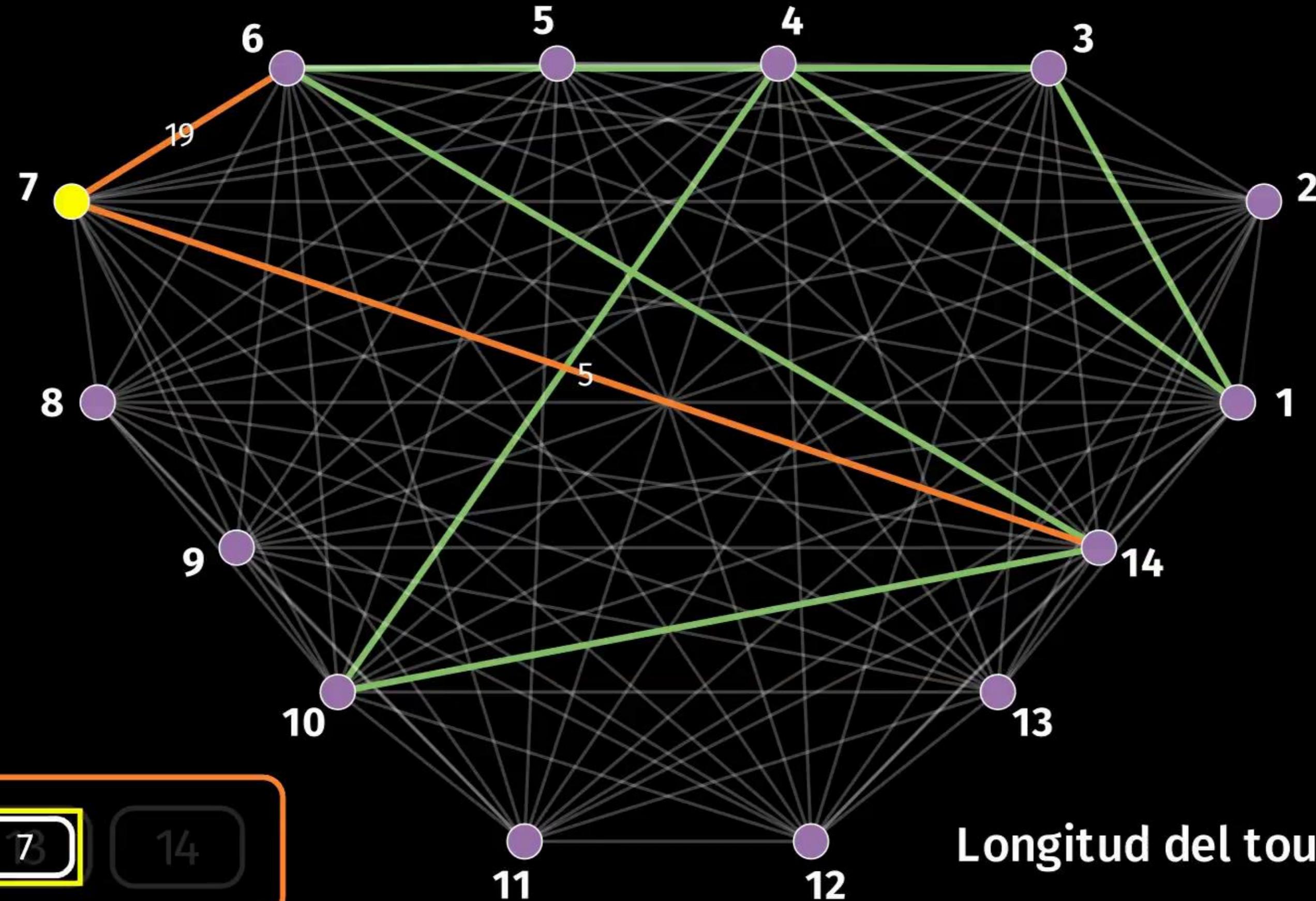
14



11

13

14

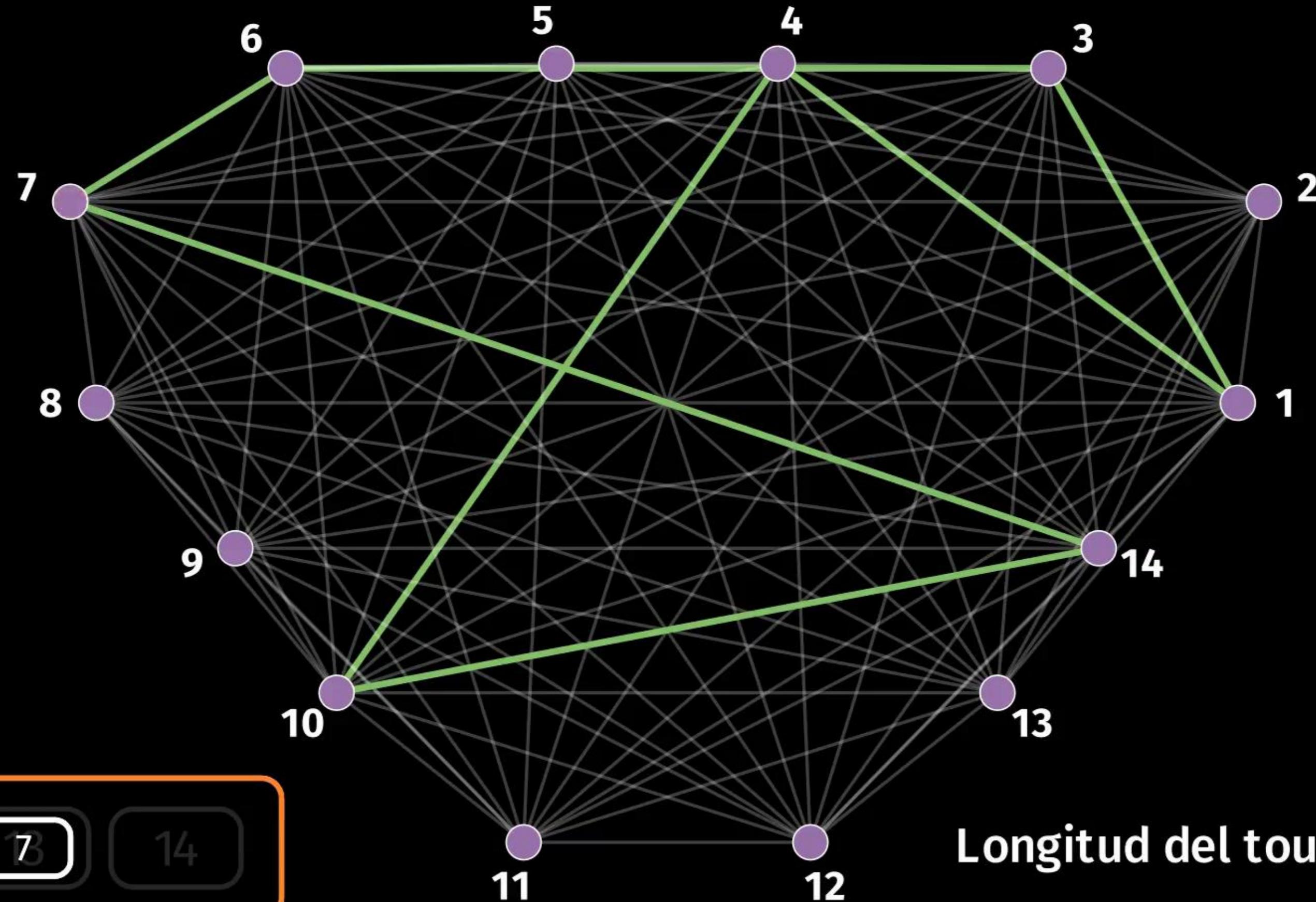


Longitud del tour: **81.0**

11

13

14

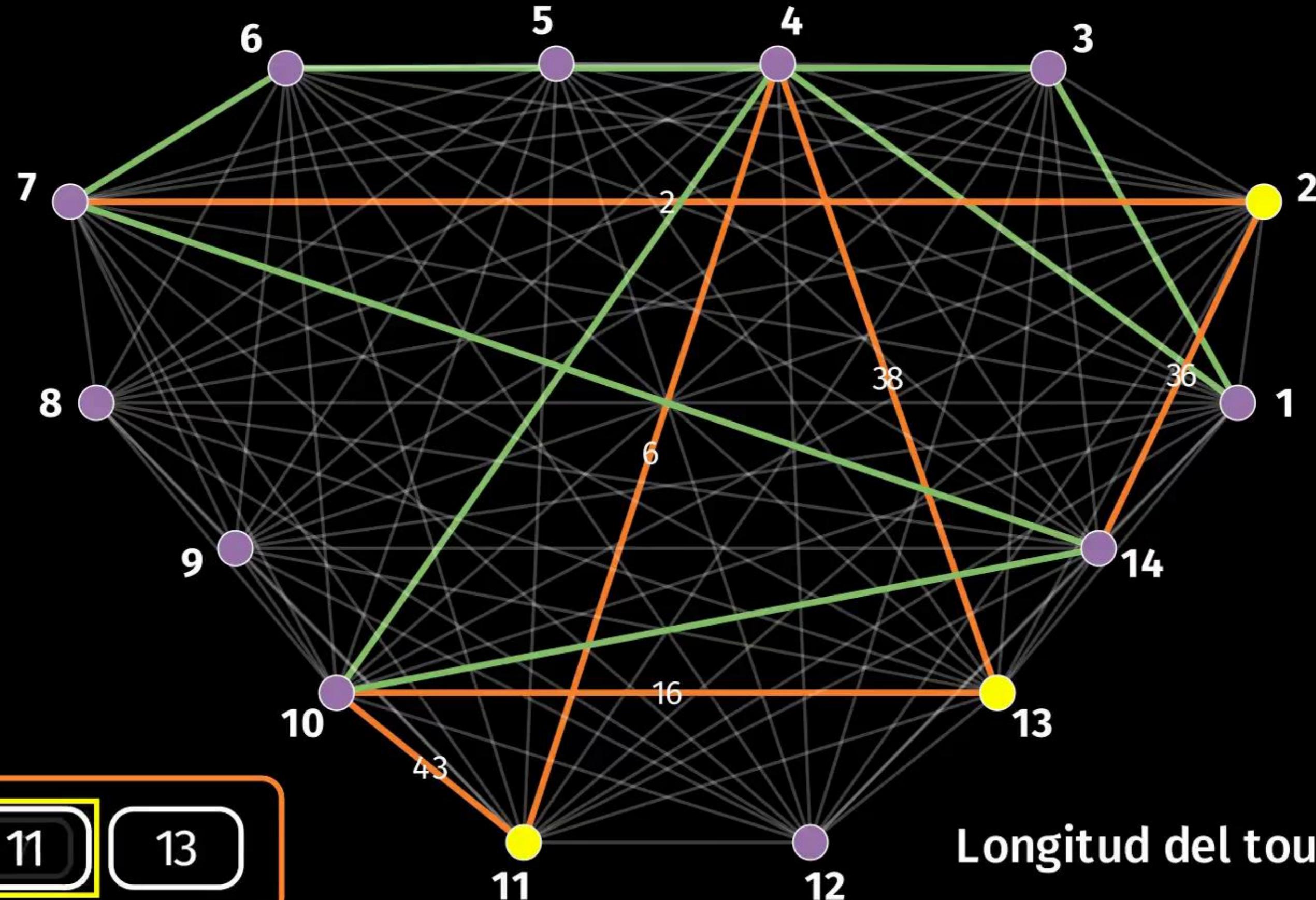


11

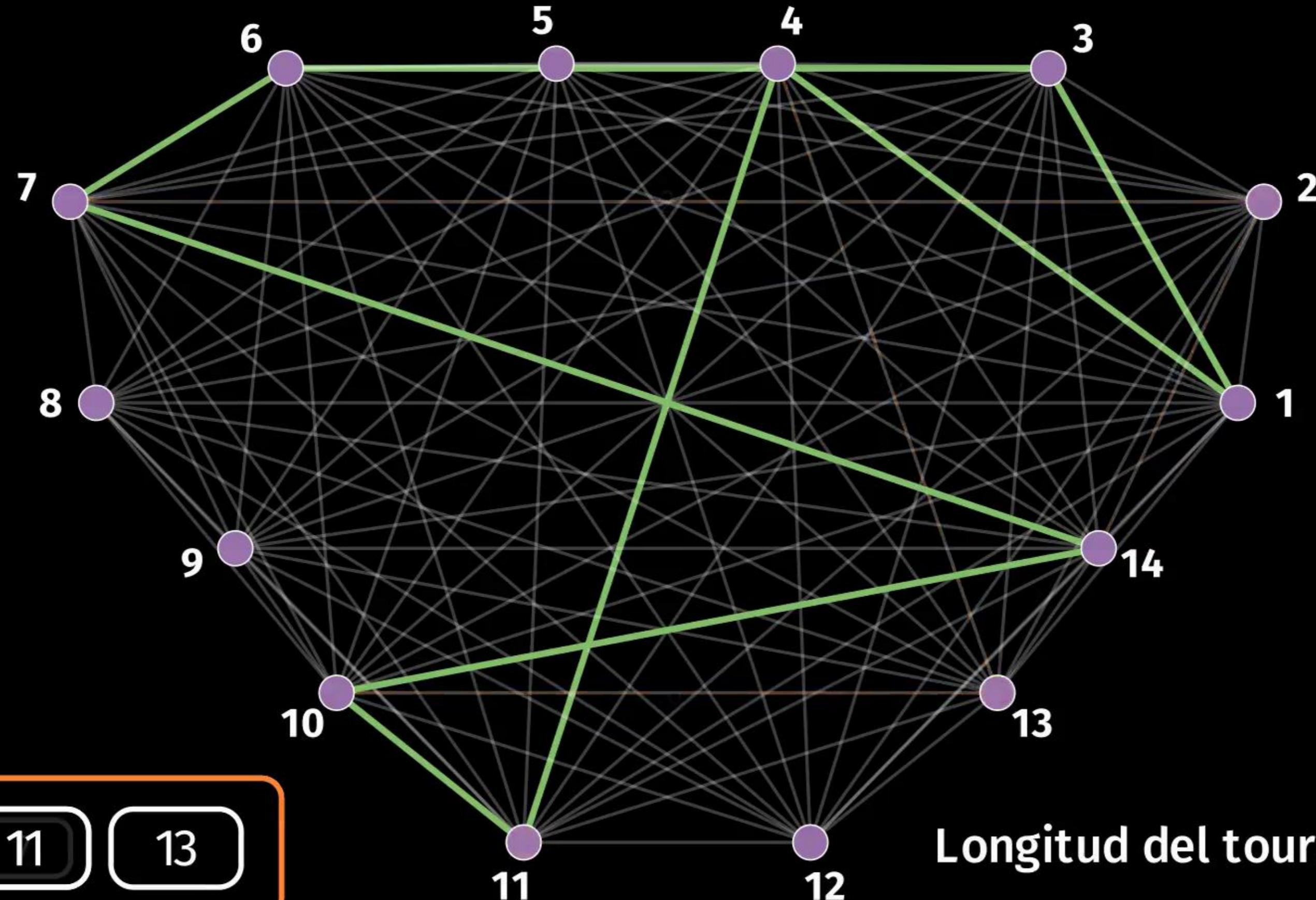
13

14

GRASP



2 11 13

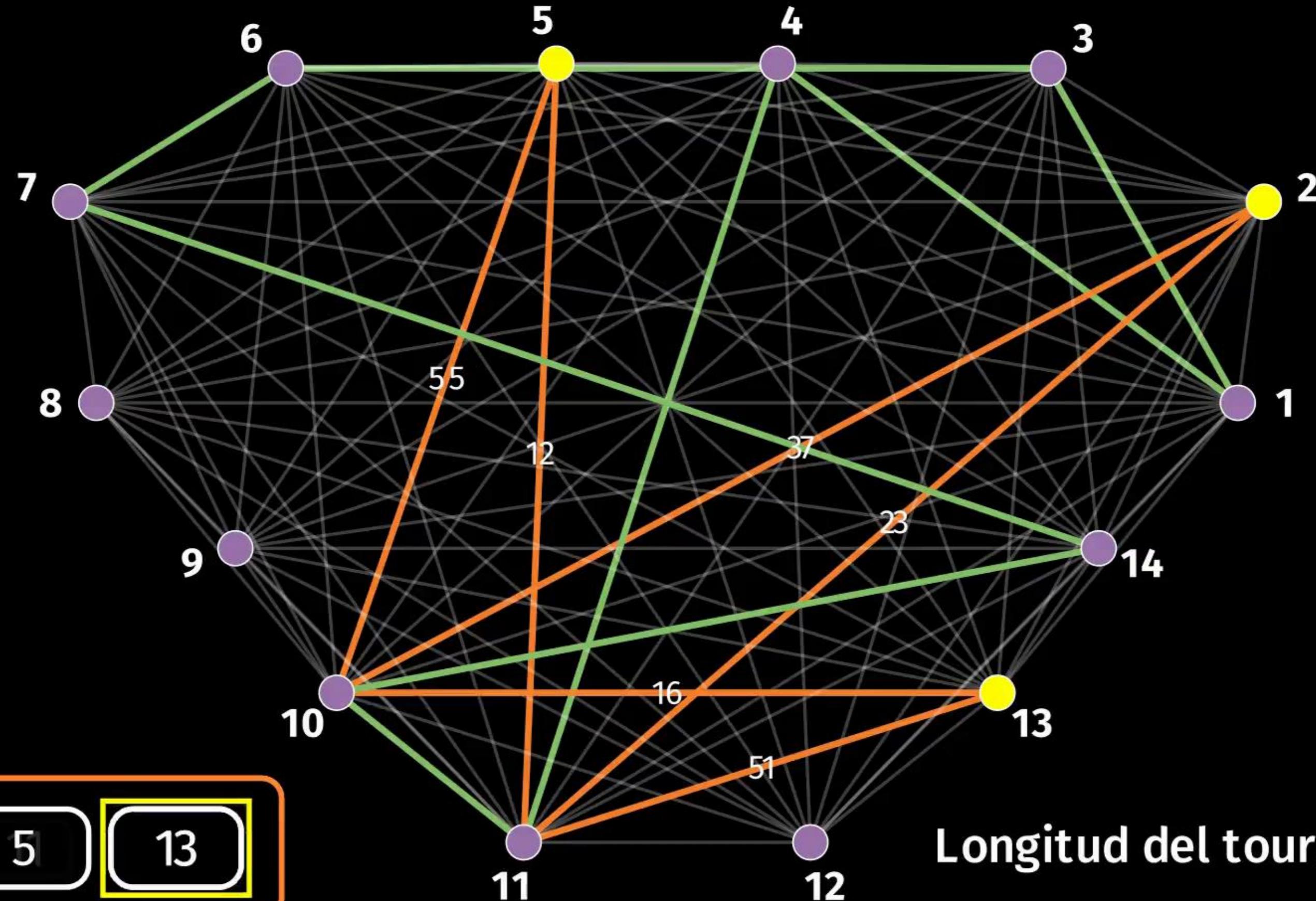


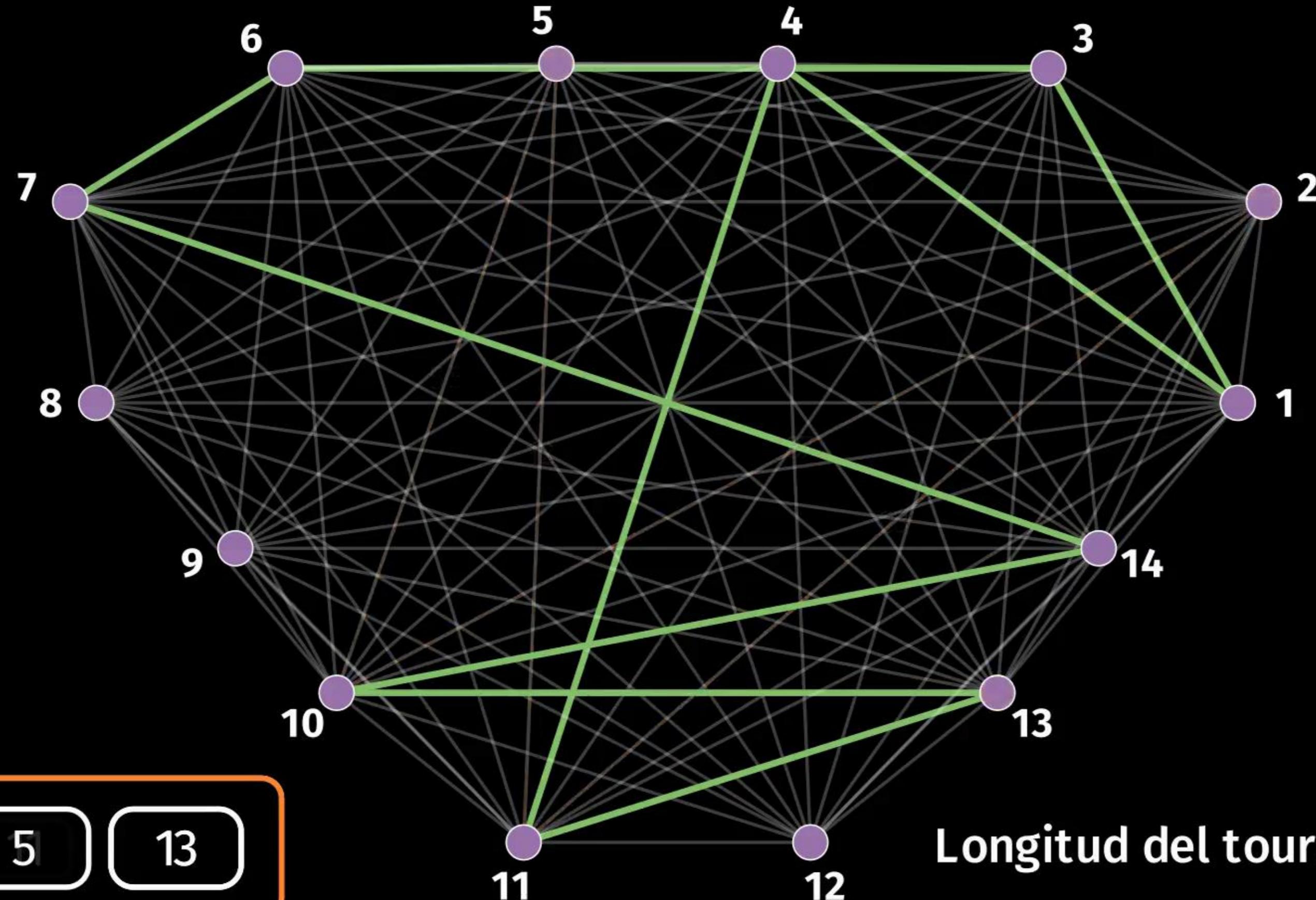
2

11

13

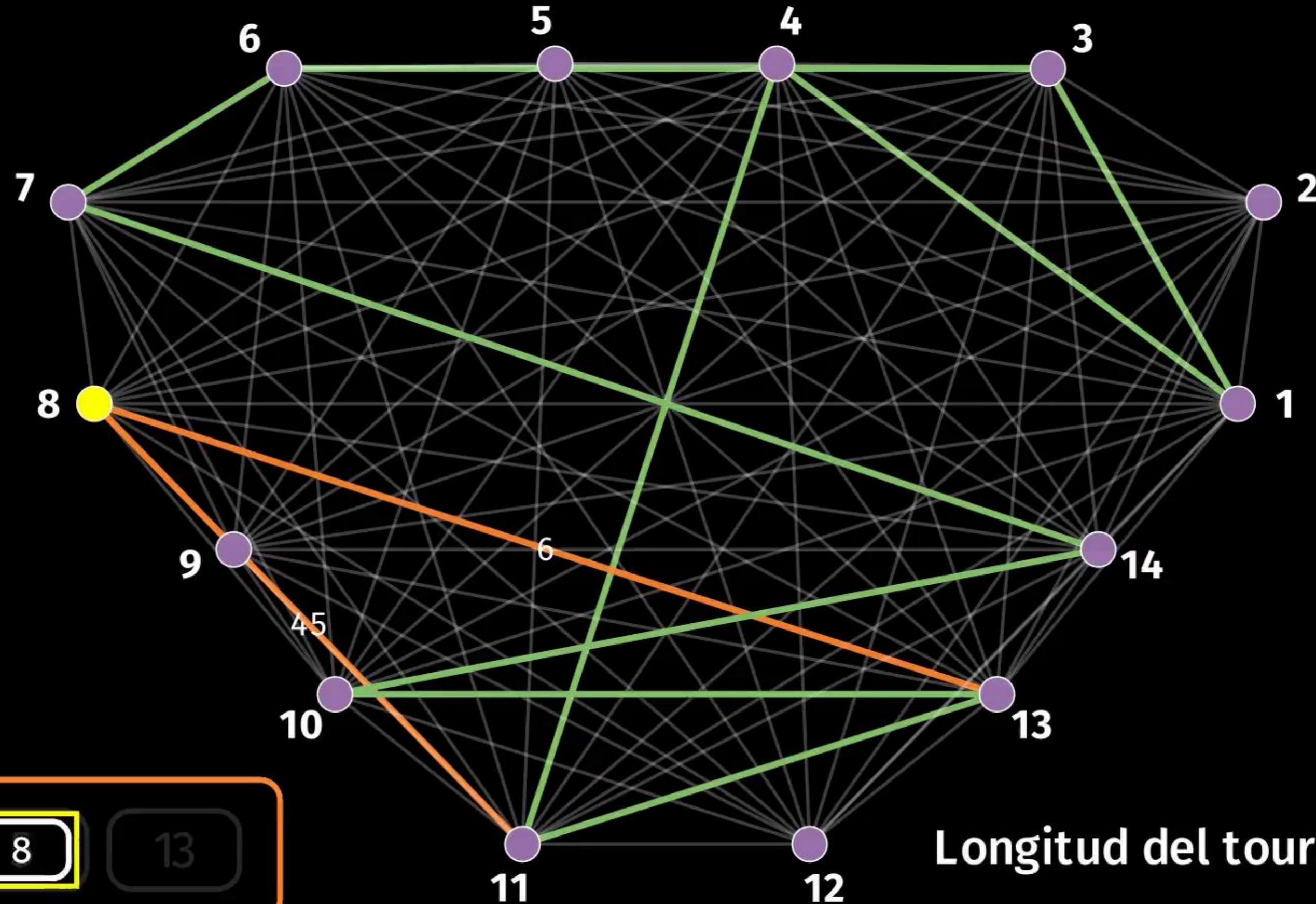
Longitud del tour: 111.0

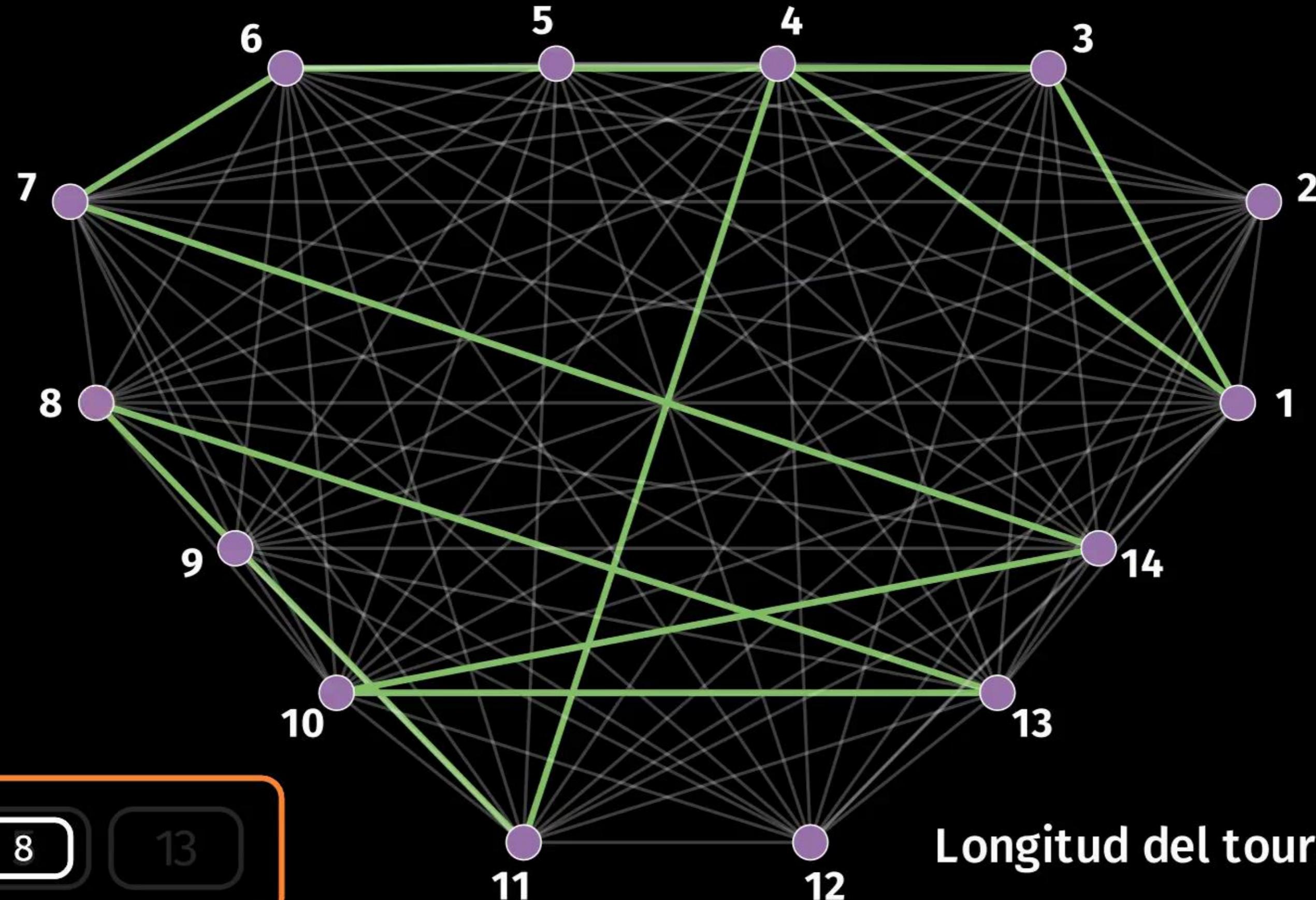




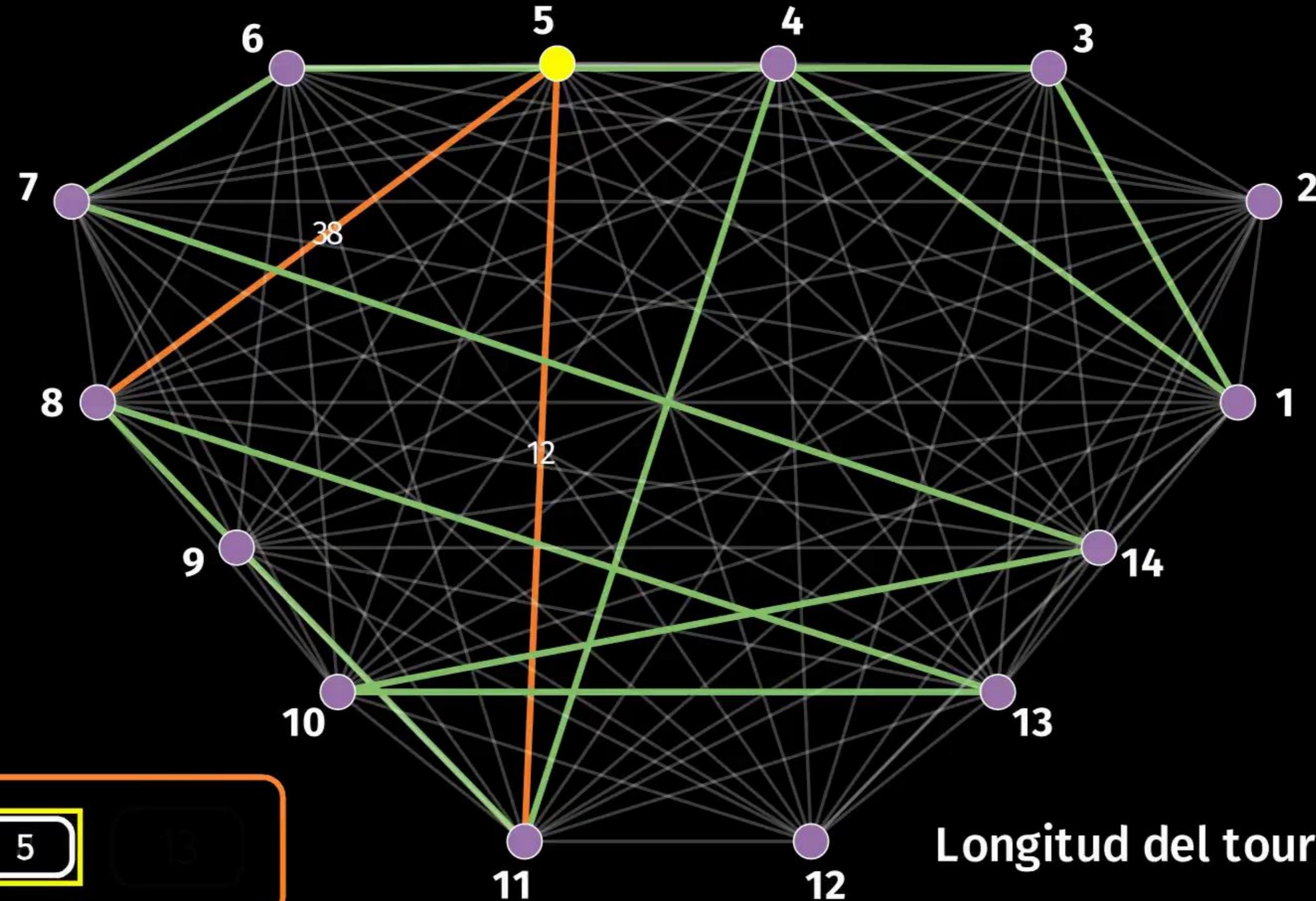
2 5 13

GRASP



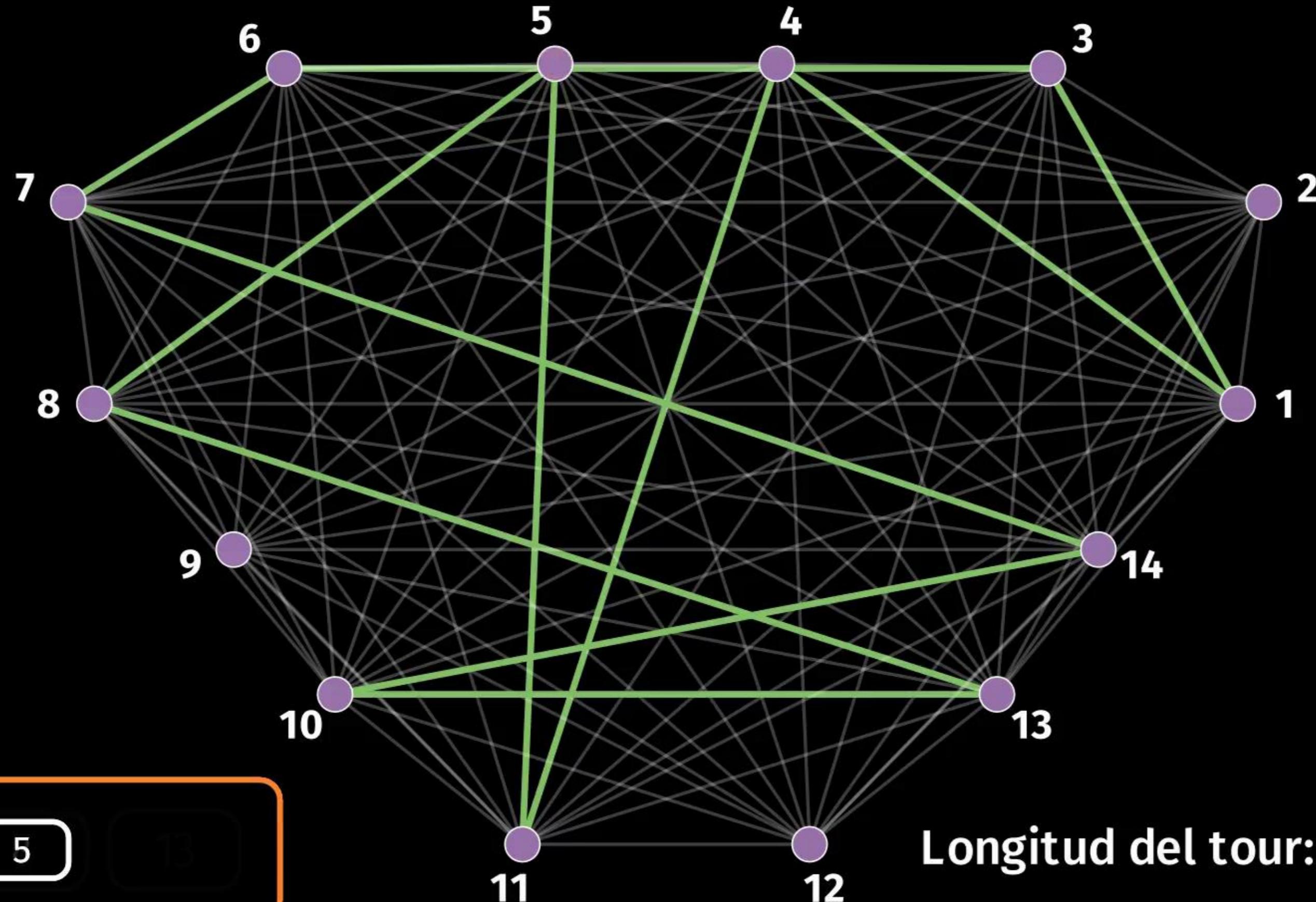


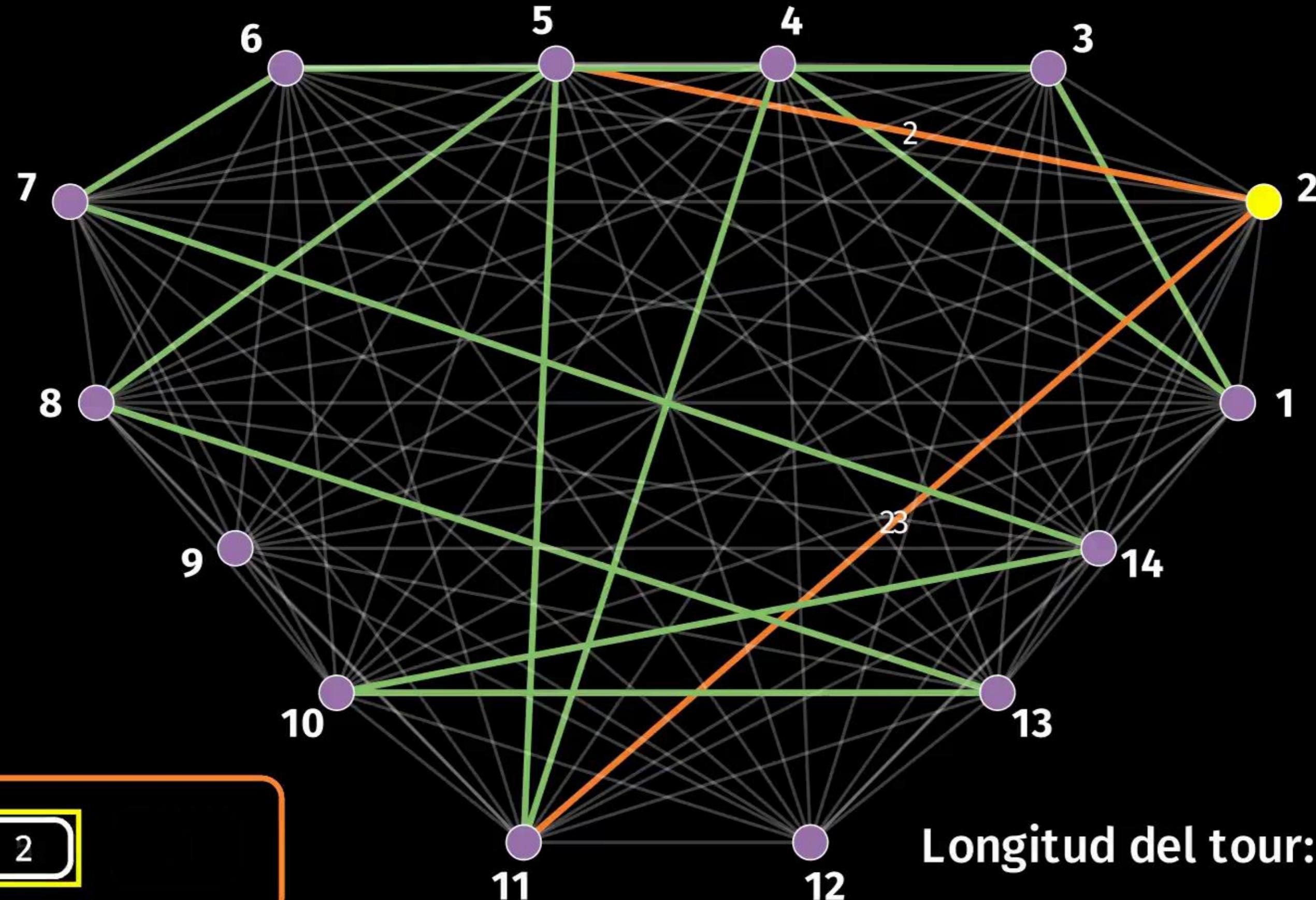
Longitud del tour: **135.0**

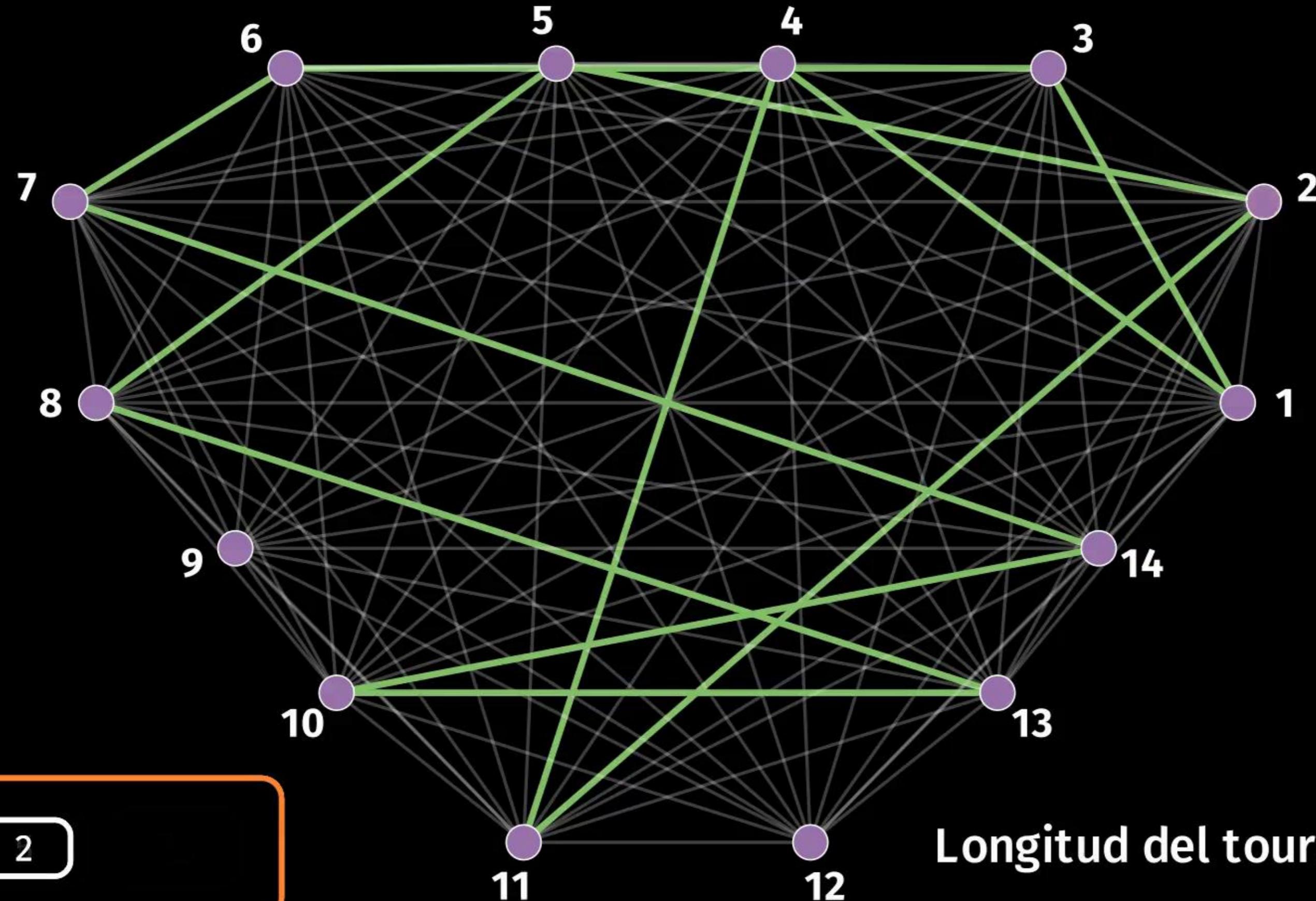


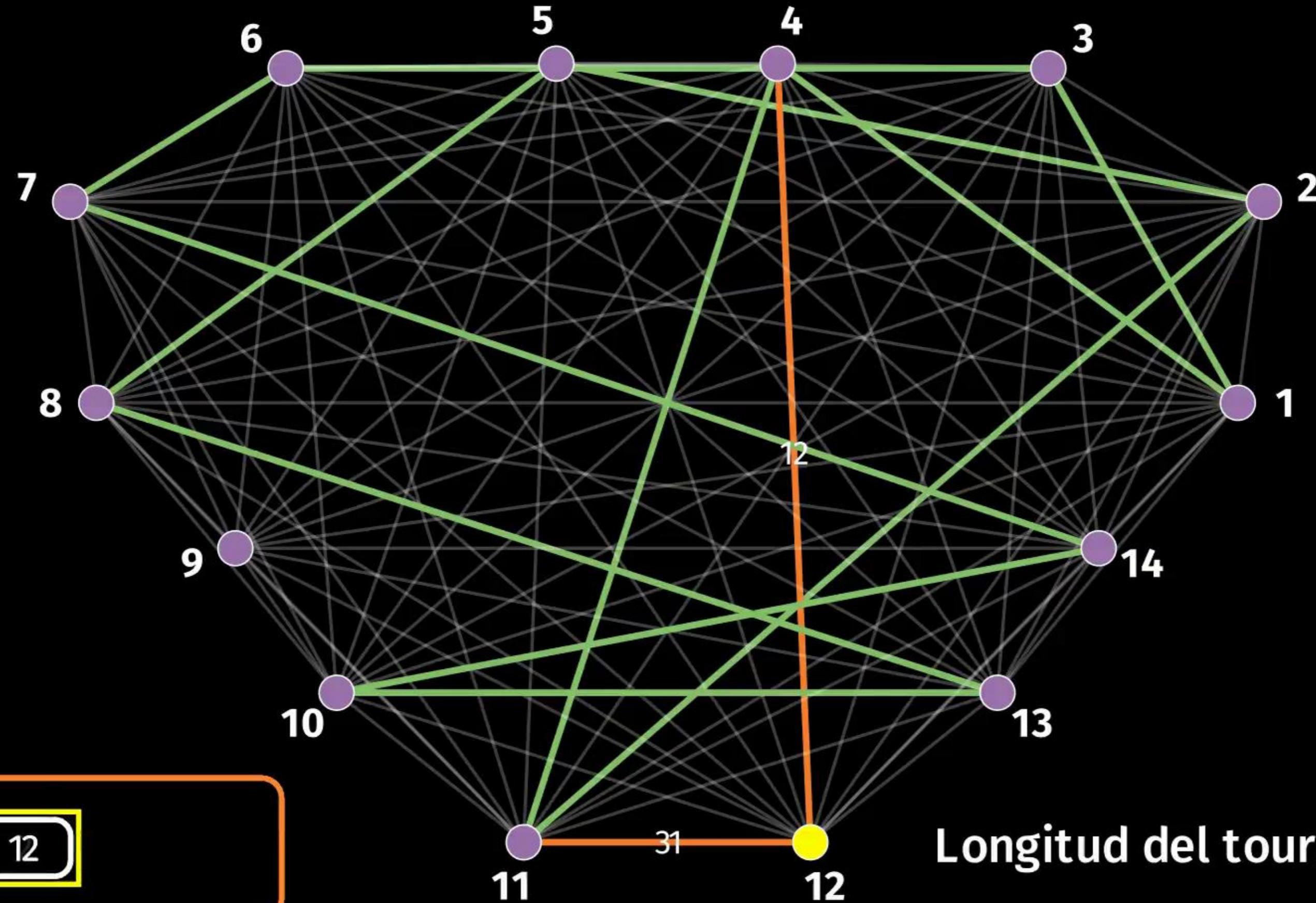
5

Longitud del tour: 135.0



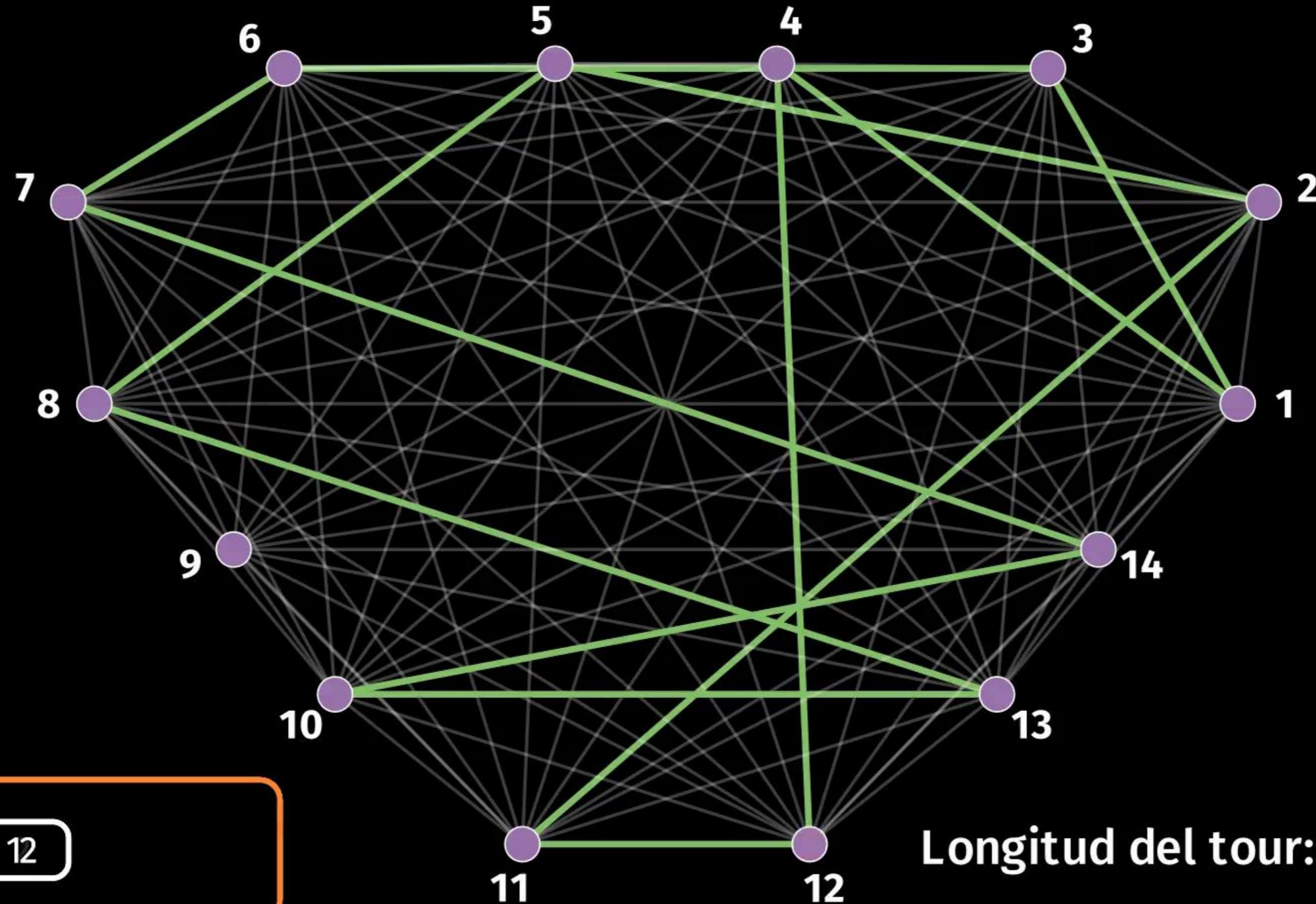






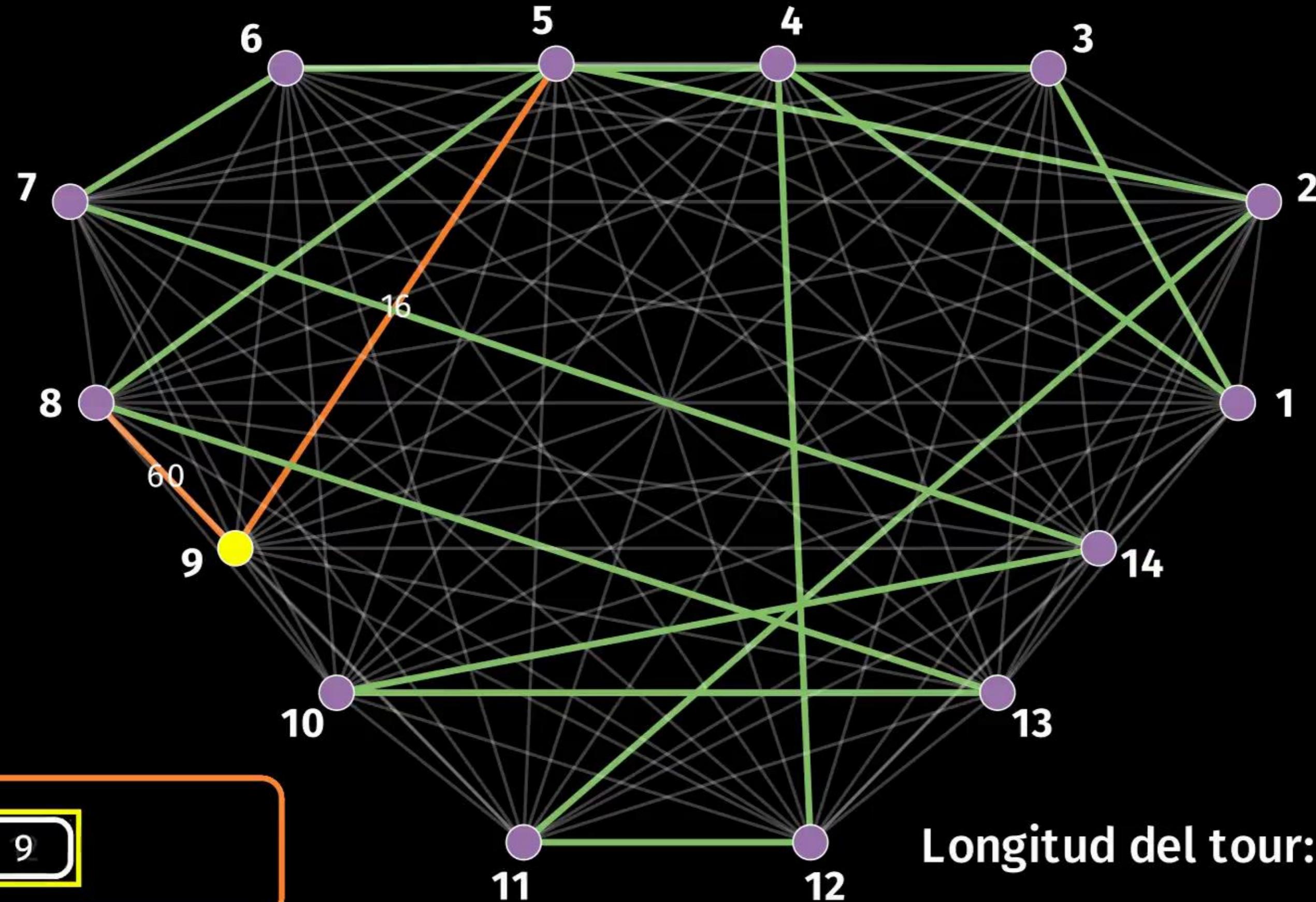
12

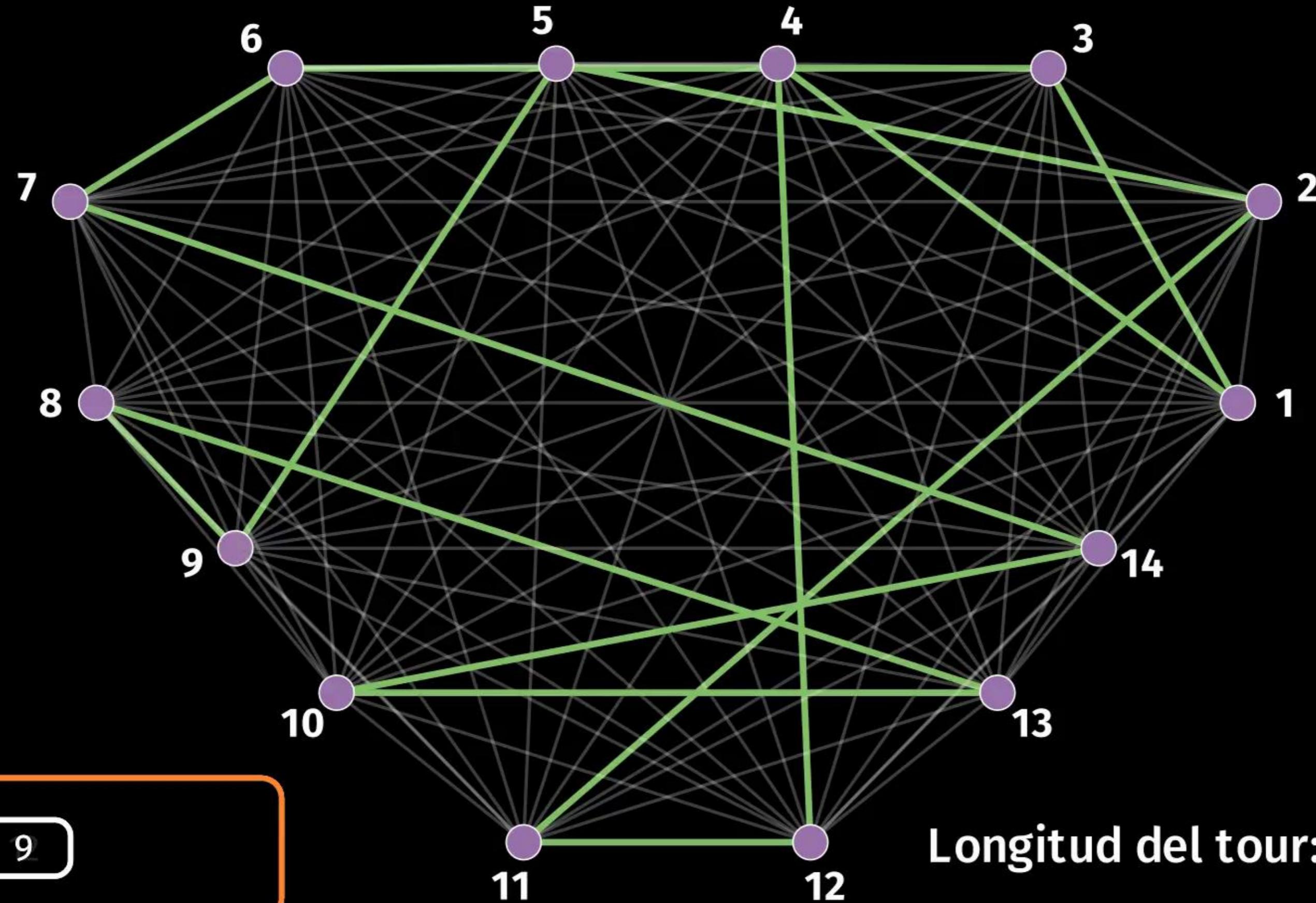
Longitud del tour: 153.0

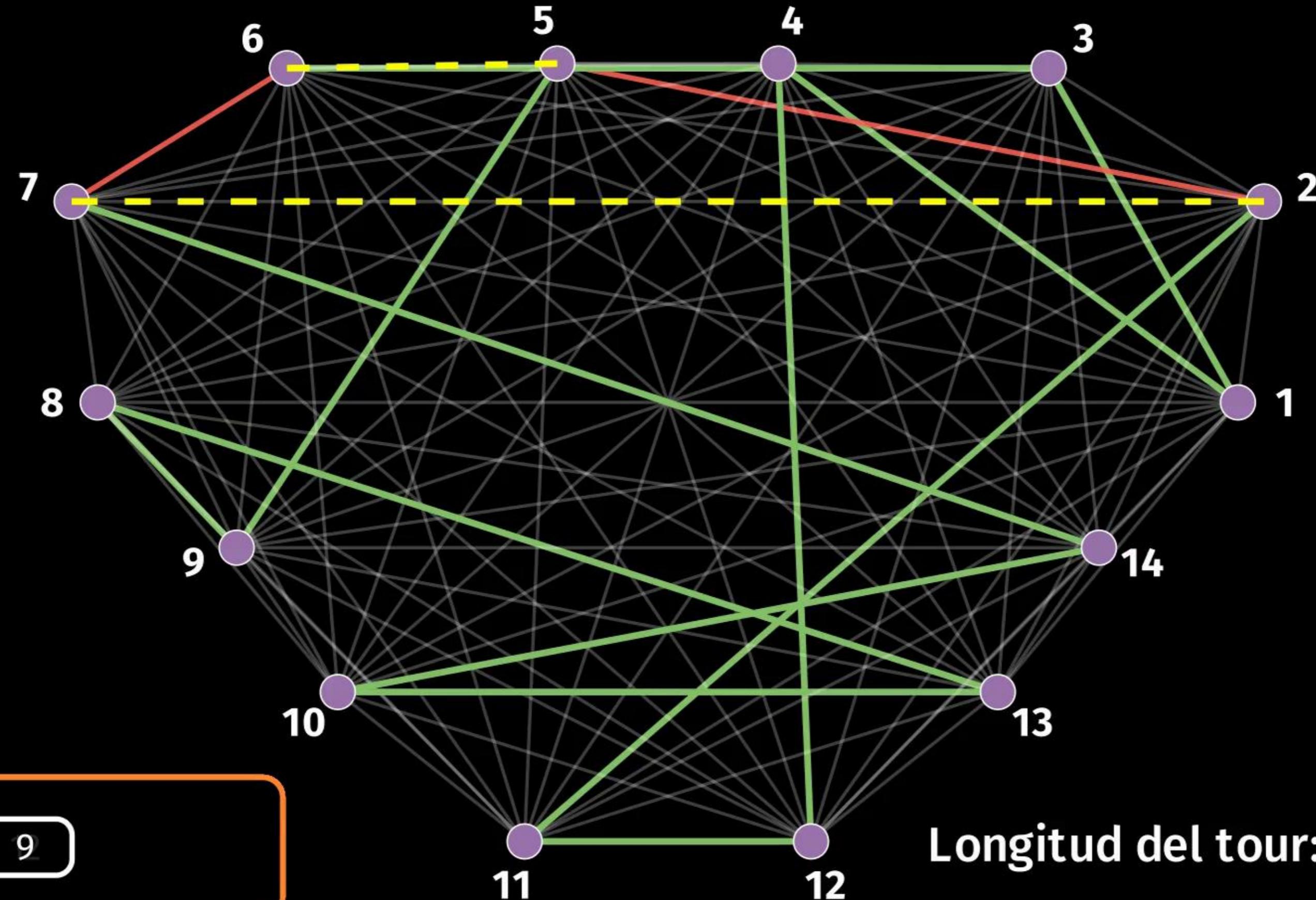


12

Longitud del tour: 190.0

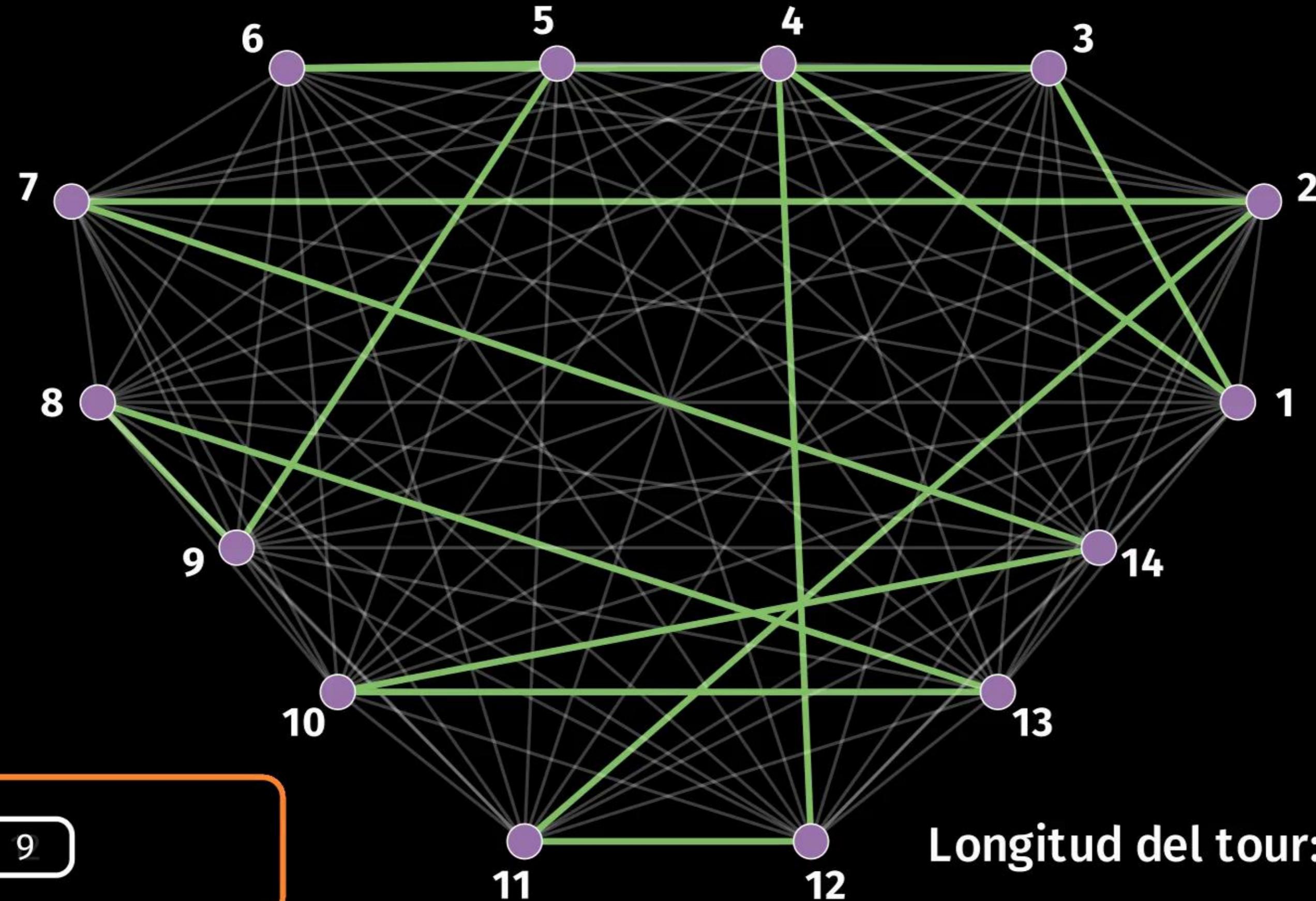






9

Longitud del tour: 230.0



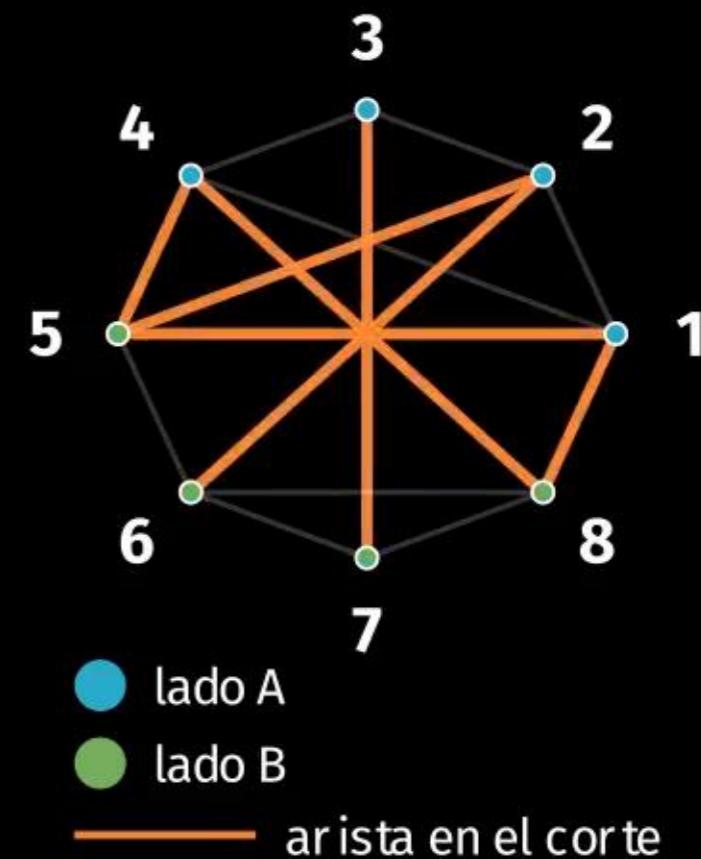
Max-Cut

Problema:

Partitionar los vértices en dos grupos para maximizar la suma de pesos de las aristas que cruzan entre grupos.

Intuición:

Pinta cada vértice AZUL/VERDE. Las aristas entre colores distintos se iluminan NARANJO y suman al objetivo.

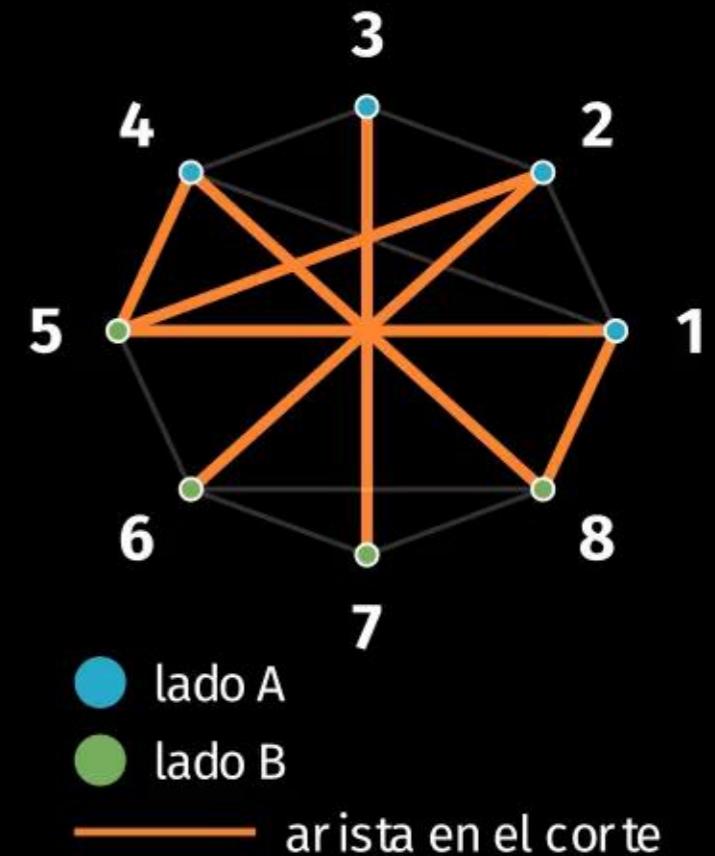


Problema:

Partition the vertices into two groups to maximize the sum of weights of edges crossing between groups.

Intuición:

Paint each vertex BLUE/GREEN. Edges between different colors are highlighted ORANGE and contribute to the objective.



Problema:

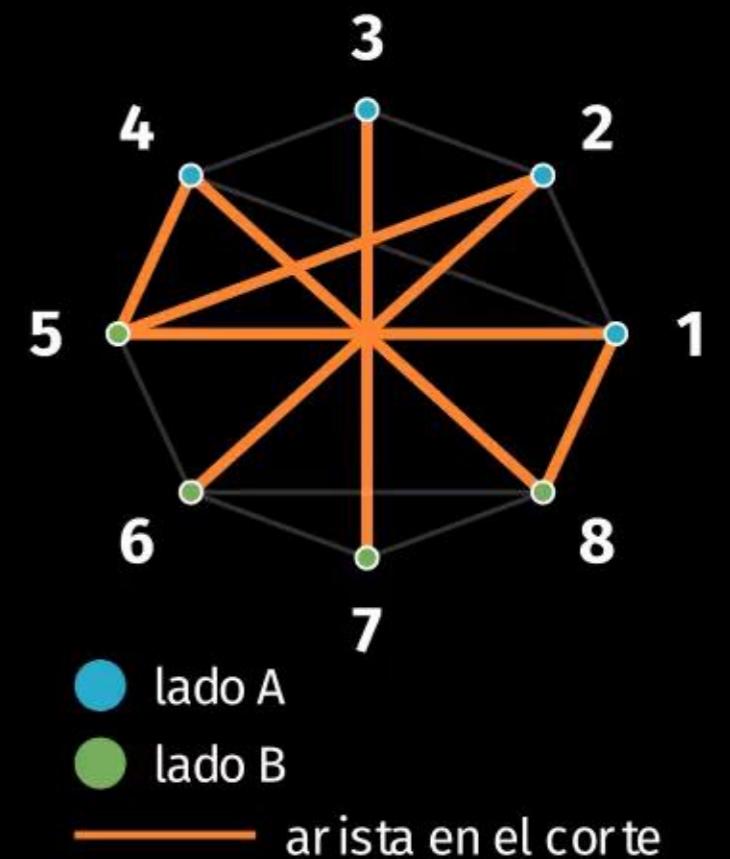
Partition the vertices into two groups to maximize the sum of weights of edges crossing between groups.

Intuición:

Paint each vertex BLUE/GREEN. Edges between different colors are highlighted ORANGE and contribute to the objective.

Formulación:

$$\max \sum_{(u,v) \in E} w_{uv} \mathbf{1}\{s(u) \neq s(v)\}$$



Problema:

Partitionar los vértices en dos grupos para maximizar la suma de pesos de las aristas que cruzan entre grupos.

Intuición:

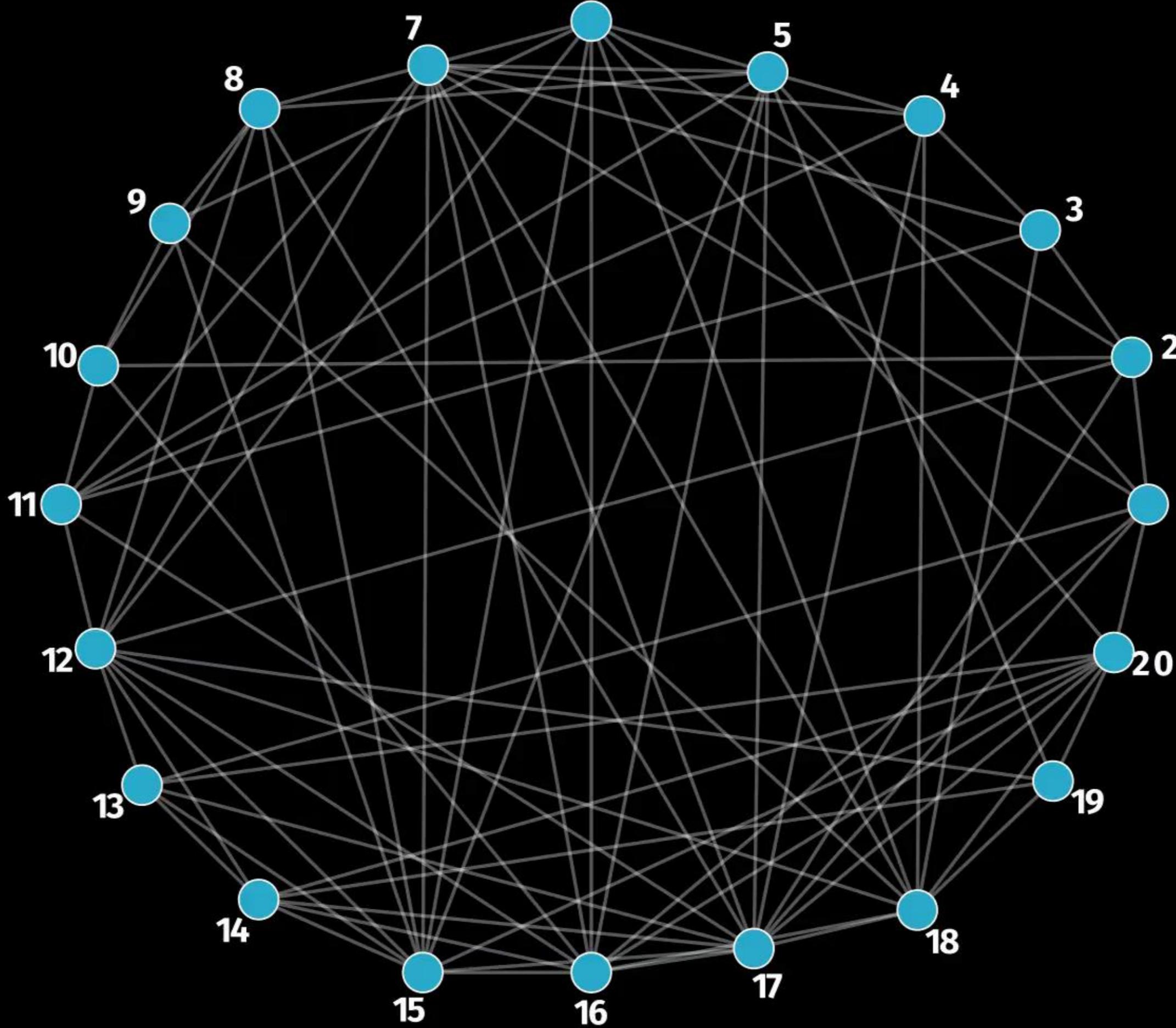
Pinta cada vértice AZUL/VERDE. Las aristas entre colores distintos se iluminan NARANJO y suman al objetivo.

Formulación:

$$\max \sum_{(u,v) \in E} w_{uv} \mathbf{1}\{s(u) \neq s(v)\}$$

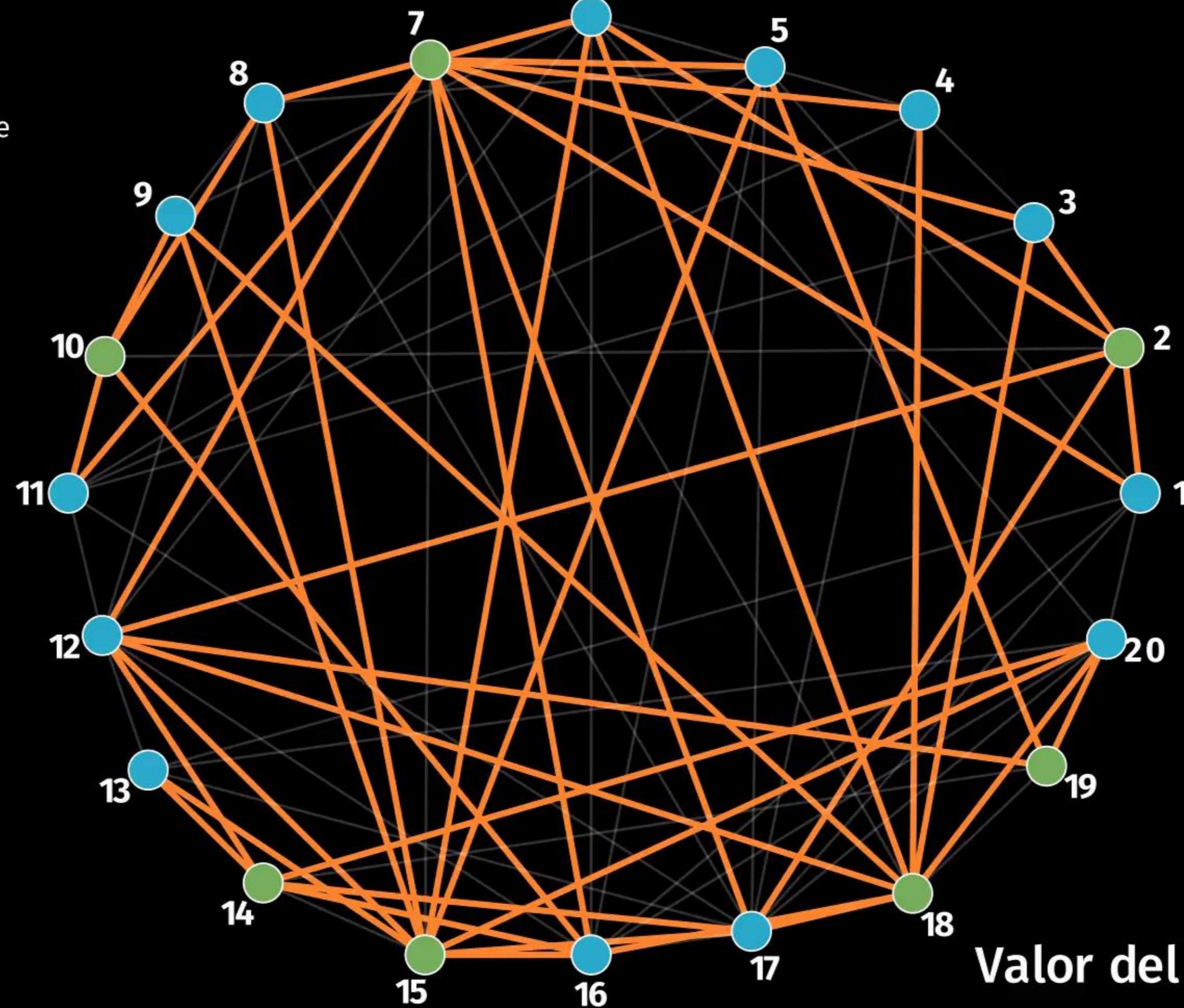
ILS – Max-Cut

ILS – Max-Cut



ILS – Max-Cut

- lado A
- lado B
- arista en el corte

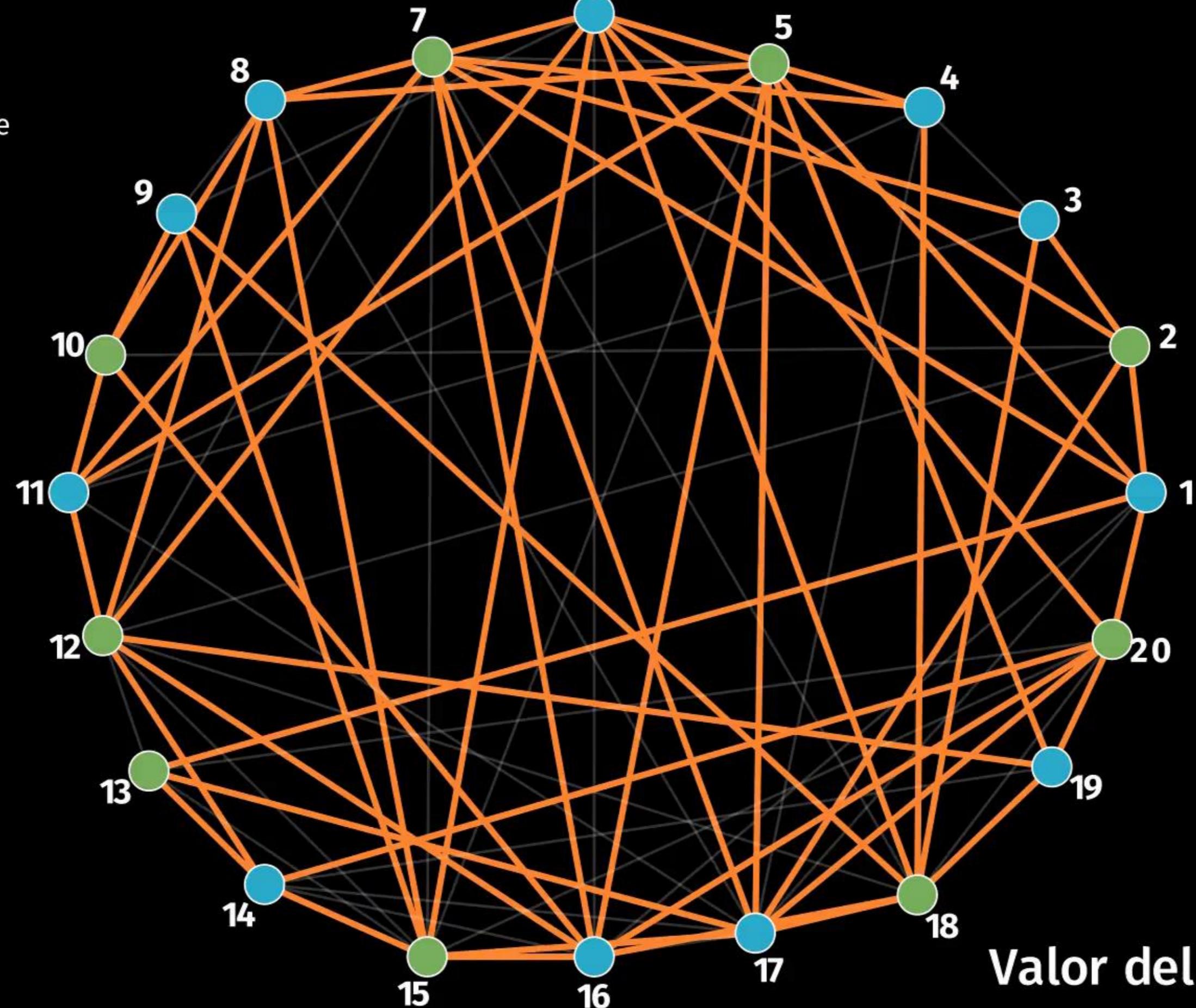


Greedy

Valor del corte: 234

ILS – Max-Cut

- lado A
- lado B
- arista en el corte

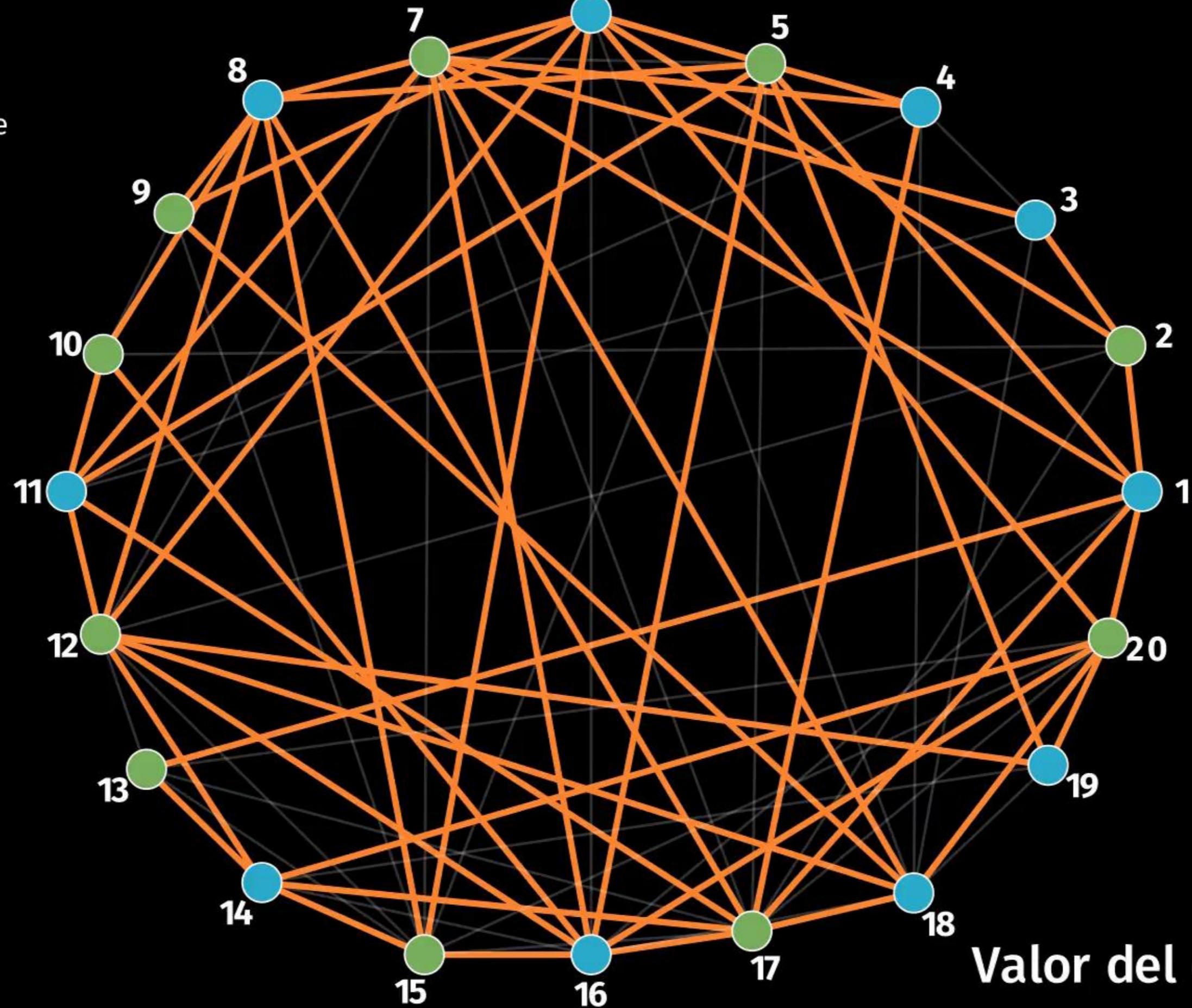


Greedy

Valor del corte: 311

ILS – Max-Cut

- lado A
- lado B
- arista en el corte

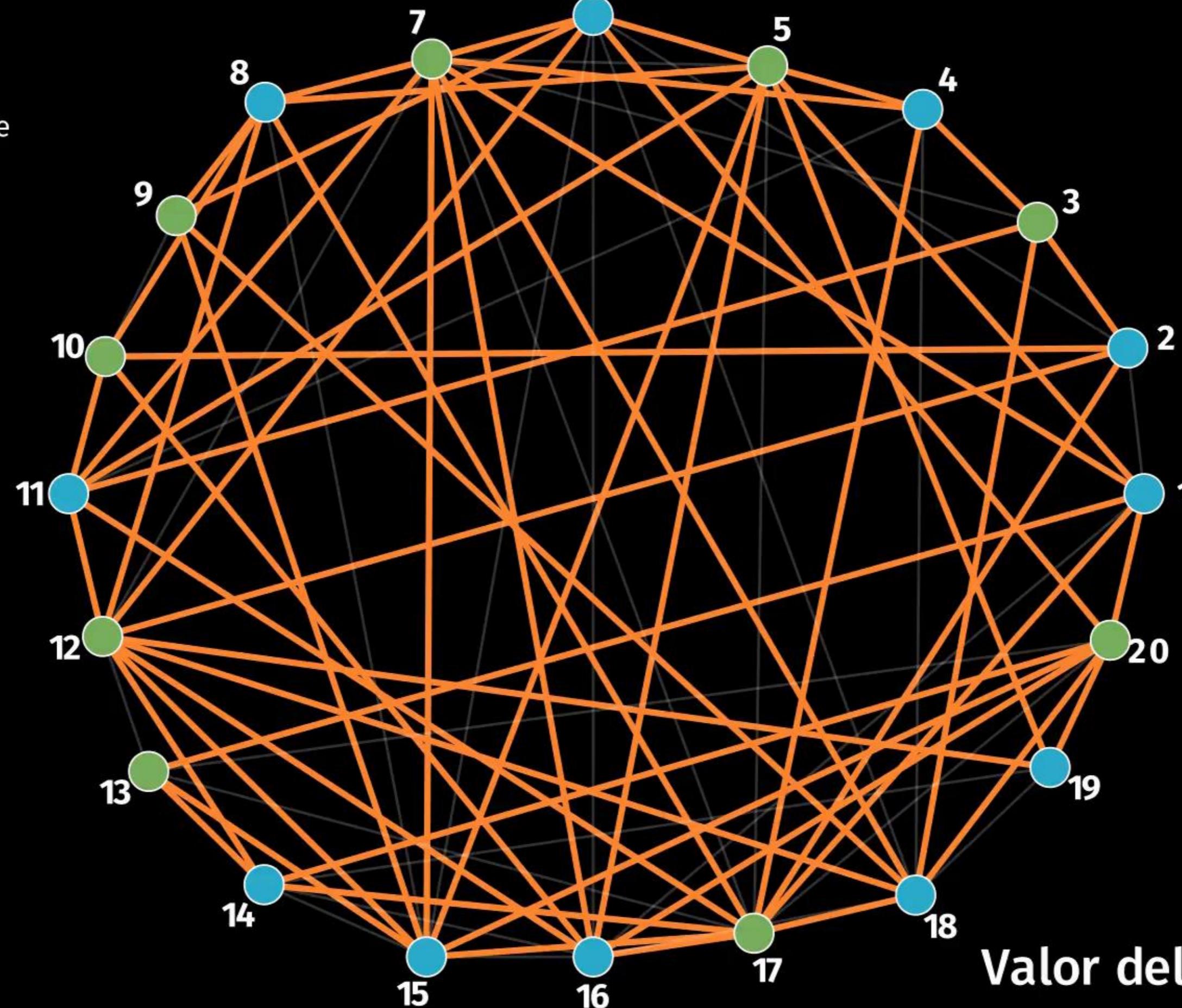


Greedy

Valor del corte: **285**

ILS – Max-Cut

- lado A
- lado B
- arista en el corte

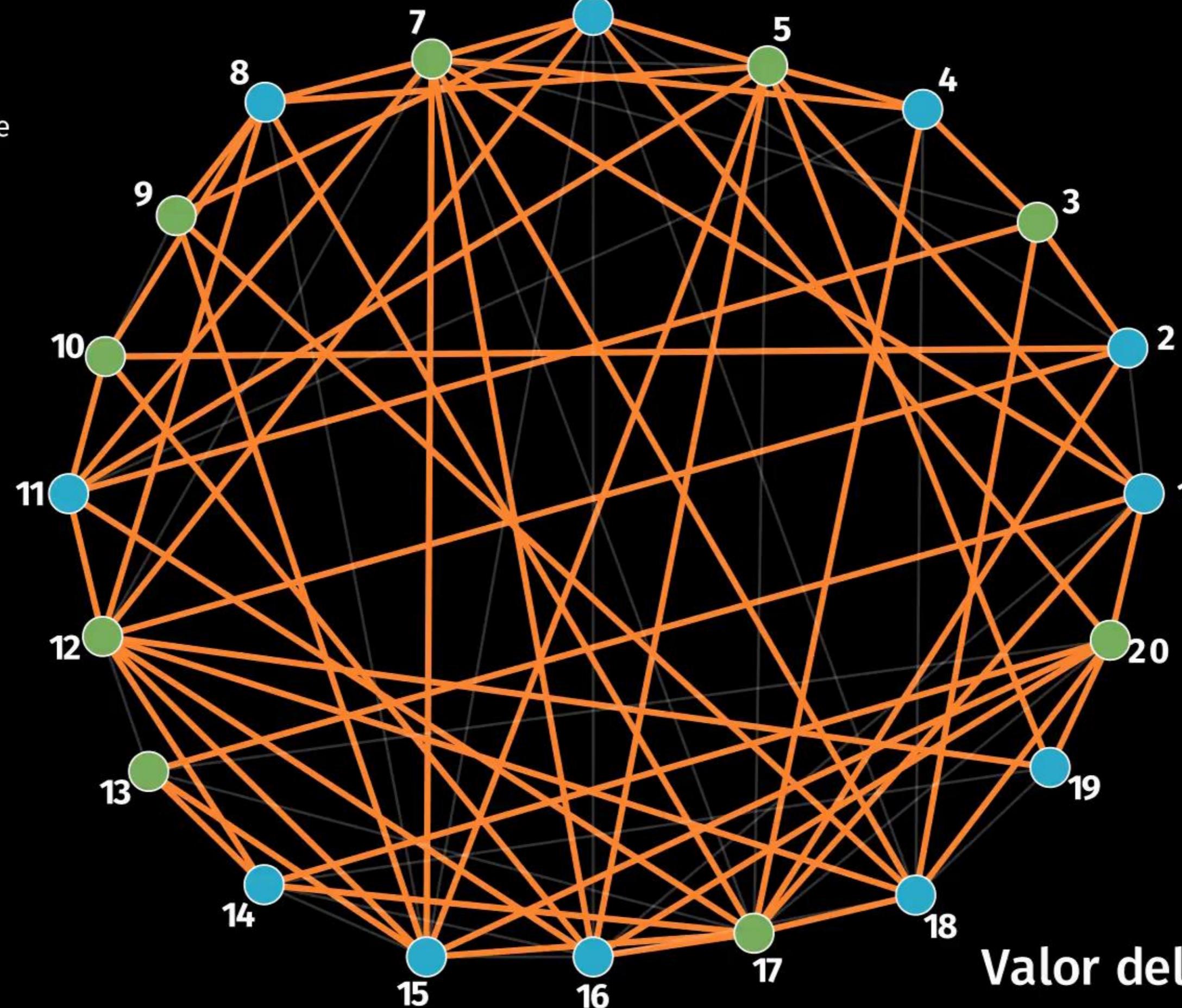


Greedy

Valor del corte: 312

ILS – Max-Cut

- lado A
- lado B
- arista en el corte

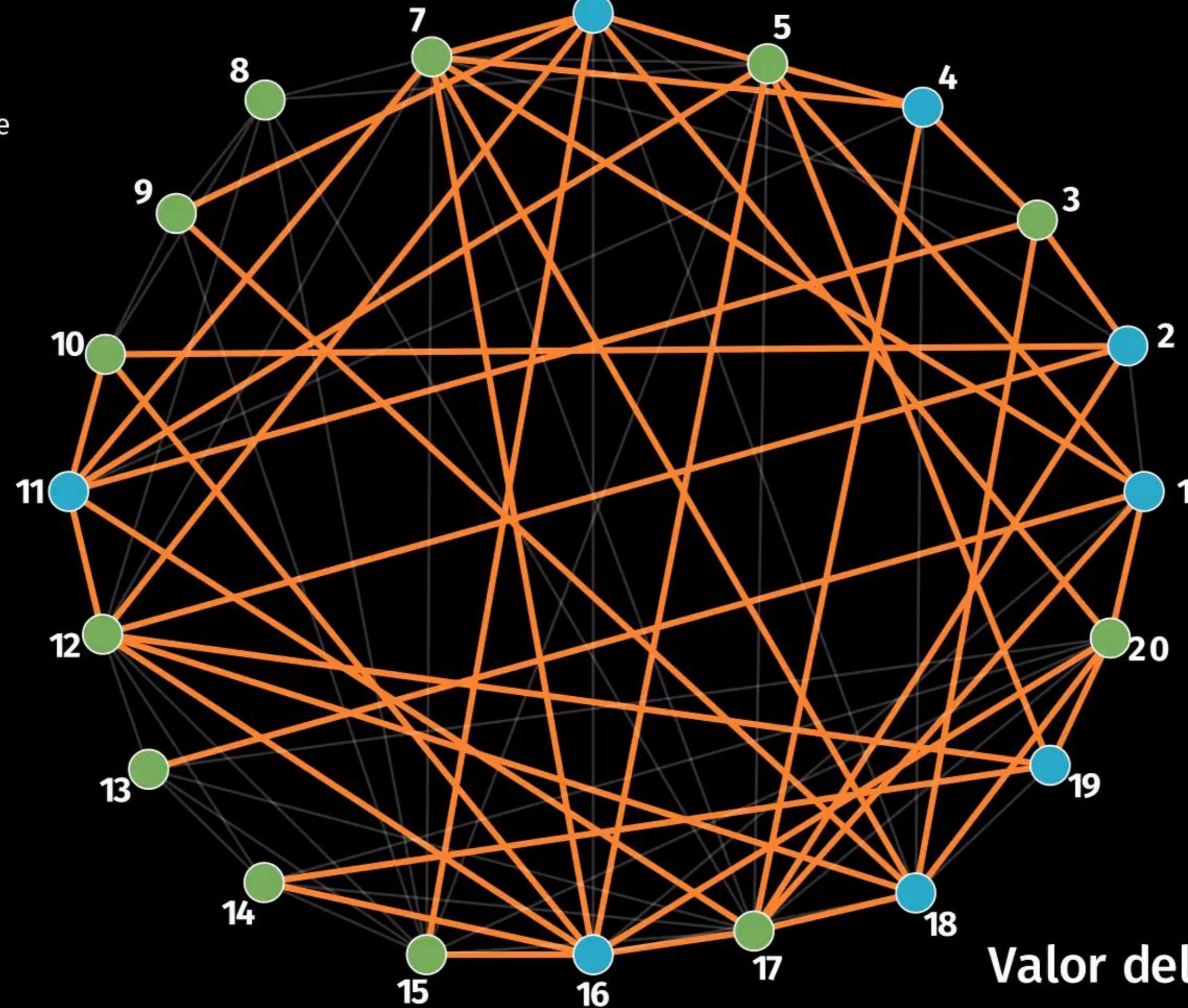


Greedy

Valor del corte: 312

ILS – Max-Cut

- lado A
- lado B
- arista en el corte

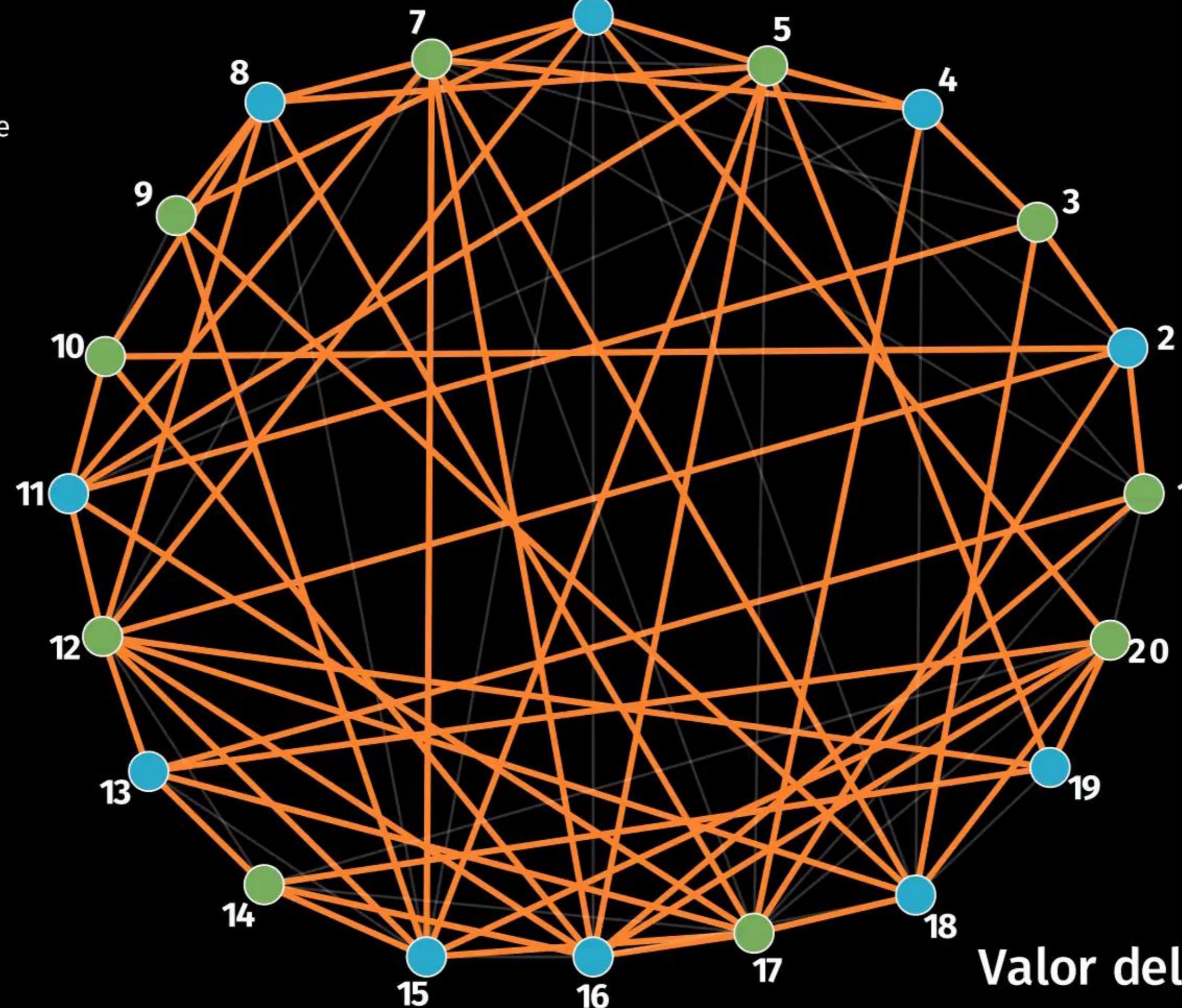


Greedy

Valor del corte: 231

ILS – Max-Cut

- lado A
- lado B
- arista en el corte

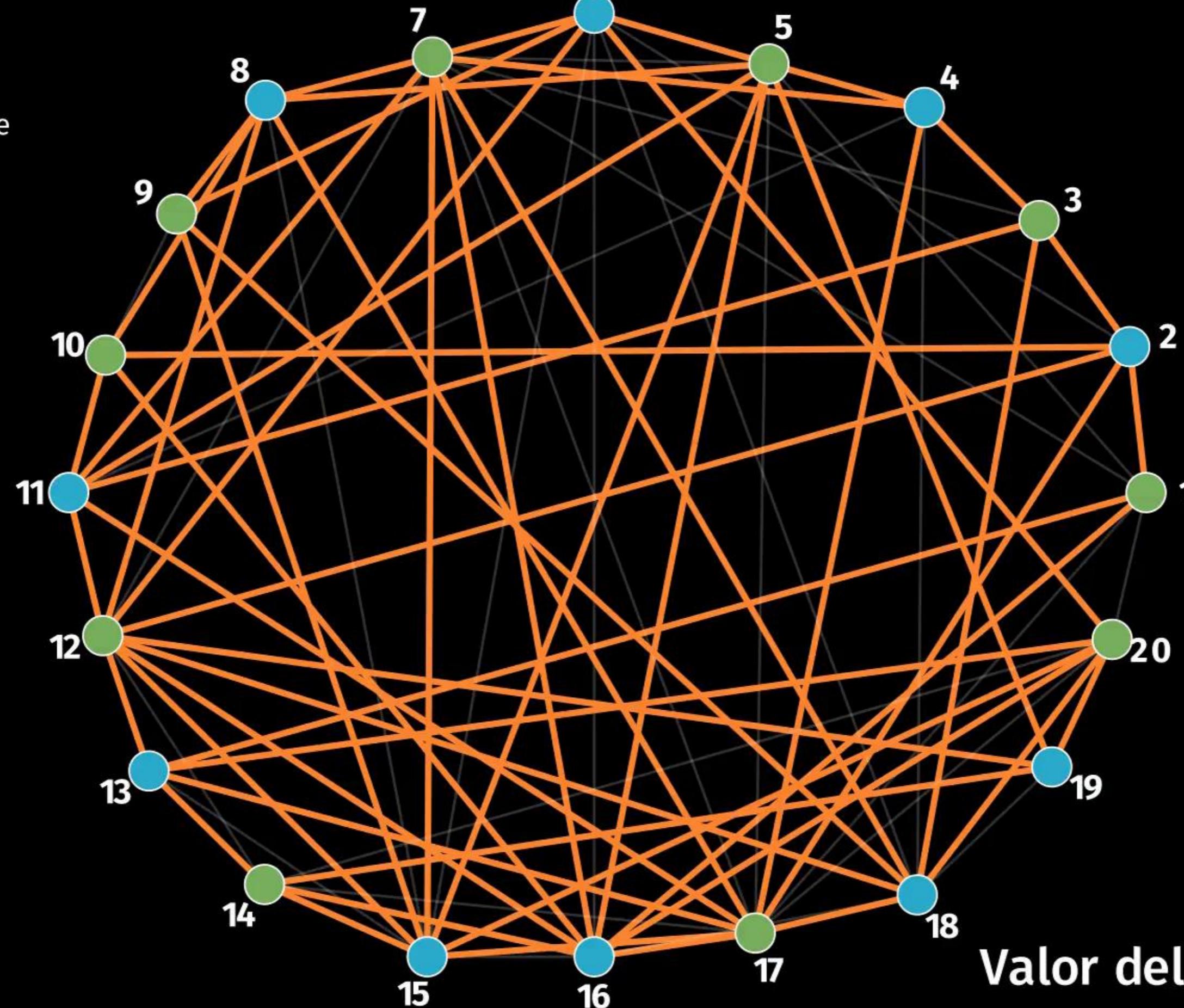


Greedy

Valor del corte: 316

ILS – Max-Cut

- lado A
- lado B
- arista en el corte



Greedy

Valor del corte: 316

