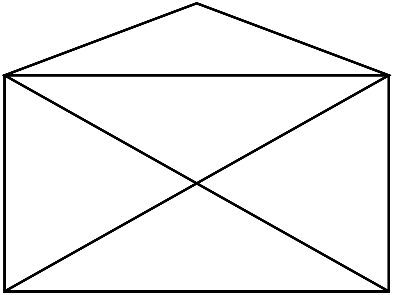
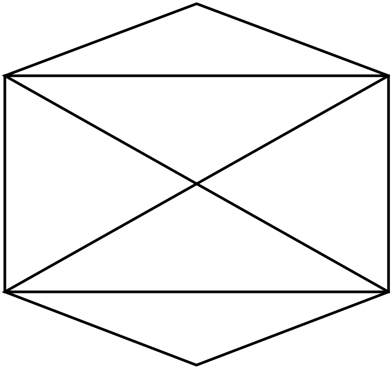
¿Se puede dibujar la siguiente figura, empezando y terminando en el mismo punto, sin levantar e lápiz del papel?



¿Y esta otra?

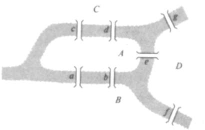


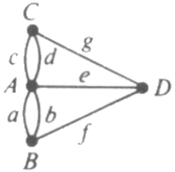
# Los puentes de Königsberg

Königsberg es famosa por ser la ciudad natal de Immanuel Kant, pero también es famosa por sus siete puentes y por el problema que consistía en saber si una persona podría cruzar todos los puentes una sola vez, volviendo al lugar de donde partió.

# Los puentes de Königsberg

Este problema fue resuelto por Euler en 1736, quien demostró que no era posible. Para eso modeló el problema como un problema de grafos: recorrer todas las aristas de un grafo una y solo una vez, volviendo al vértice inicial.

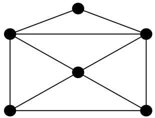


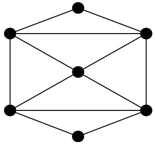



# Definiciones

- Un circuito Euleriano en un grafo o multigrafo $G$ es un circuito que recorre cada arista una y sólo una vez.

Ej:

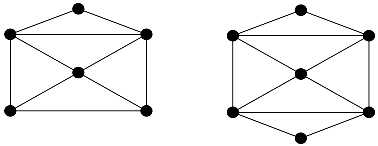




# Definiciones

- Un circuito Euleriano en un grafo o multigrafo $G$ es un circuito que recorre cada arista una y sólo una vez.
- Un grafo o multigrafo es Euleriano si tiene un circuito Euleriano.

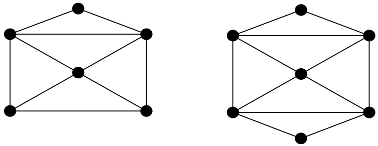
Ej:



# Definiciones

- Un circuito Euleriano en un grafo o multigrafo $G$ es un circuito que recorre cada arista una y sólo una vez.
- Un grafo o multigrafo es Euleriano si tiene un circuito Euleriano.

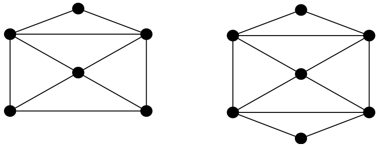
Ej:



# Definiciones

- Un circuito Euleriano en un grafo o multigrafo $G$ es un circuito que recorre cada arista una y sólo una vez.
- Un grafo o multigrafo es Euleriano si tiene un circuito Euleriano.

Ej:



Obs: Si un grafo tiene al menos dos componentes conexas no triviales, no puede tener camino ni circuito Euleriano.

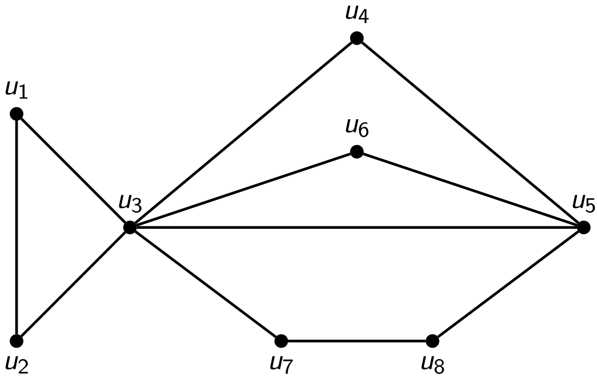
# El Teorema de Euler

**Teorema**

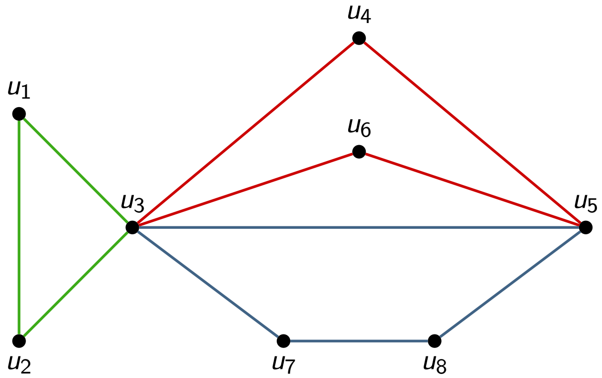
Son equivalentes, para $G$ grafo o multigrafo conexo:

1. $G$ es Euleriano.
2. Todo vértice de $G$ tiene grado par.
3. Las aristas de $G$ pueden particionarse en circuitos.

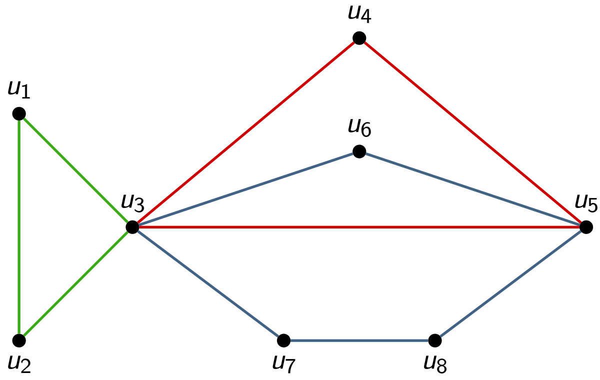
# Grafos Eulerianos



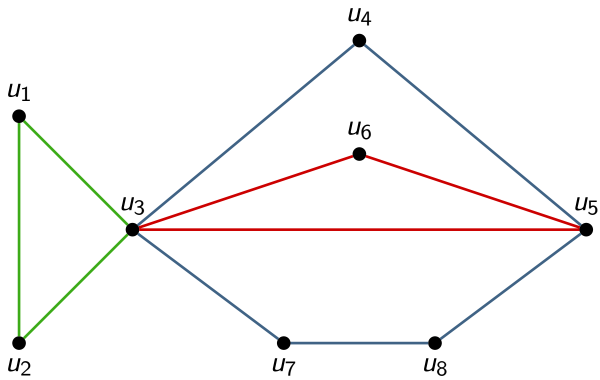
# Grafos Eulerianos



# Grafos Eulerianos



# Grafos Eulerianos



- En base a los teoremas anteriores, ¿cuál es la complejidad computacional de saber si un grafo es o no Euleriano?

- En base a los teoremas anteriores, ¿cuál es la complejidad computacional de saber si un grafo es o no Euleriano?
- A partir de la demostración, se puede escribir un algoritmo para construir un circuito Euleriano para un grafo que tiene todos sus vértices de grado par.

# Grafos Eulerianos

**Entrada:** $G = (V, E)$ conexo con $d(v)$ par para todo $v \in V$.
**Salida:** Un circuito Euleriano de $G$.

- Comenzar por cualquier vértice $v$ y construir un ciclo $Z$.
- **Mientras** $E \setminus Z \neq \emptyset$ **hacer**:
  - Elegir $w$ tal que exista $(w, u) \in Z$ y $(w, z) \in E \setminus Z$.
  - Desde $w$ construir un ciclo $D$ con $D \cap Z = \emptyset$.
  - $Z :=$ unir $Z$ y $D$ por medio de $w$.
- **Fin mientras**
- **Retornar** $Z$.

# Grafos Eulerianos

- Un camino Euleriano en un grafo o multigrafo $G$ es un camino que recorre cada arista una y sólo una vez.

- Un grafo orientado o digrafo, se dice Euleriano si tiene un circuito orientado que pasa por cada arco de $G$ exactamente una vez.

# Grafos Eulerianos

- Un camino Euleriano en un grafo o multigrafo $G$ es un camino que recorre cada arista una y sólo una vez.

- Un grafo orientado o digrafo, se dice Euleriano si tiene un circuito orientado que pasa por cada arco de $G$ exactamente una vez.

**Teorema**

Un grafo o multigrafo conexo tiene un camino Euleriano si y sólo si todos sus vértices tienen grado par salvo dos.

# Grafos Eulerianos

- Un camino Euleriano en un grafo o multigrafo $G$ es un camino que recorre cada arista una y sólo una vez.

- Un grafo orientado o digrafo, se dice Euleriano si tiene un circuito orientado que pasa por cada arco de $G$ exactamente una vez.

**Teorema**

Un grafo o multigrafo conexo tiene un camino Euleriano si y sólo si todos sus vértices tienen grado par salvo dos.

**Teorema**

Un digrafo conexo es Euleriano si y sólo si para todo vértice $v$ de $G$ se verfica que $d_{in}(v) = d_{out}(v)$.

# Problema del cartero chino (Kuan, 1962)

**Definición.** Dado un grafo $G = (V, E)$ con longitudes asignadas a sus aristas, $\ell : E \to \mathbb{R}_+$, el problema del cartero chino consiste en encontrar un circuito de longitud mínima que pase por cada arista de $G$ **al menos** una vez.

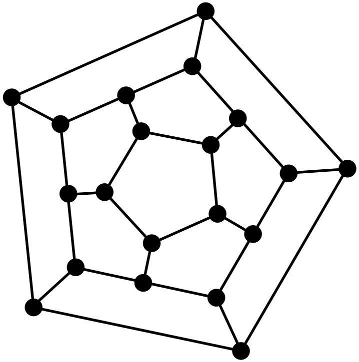
# Problema del cartero chino (Kuan, 1962)

**Definición.** Dado un grafo $G = (V, E)$ con longitudes asignadas a sus aristas, $\ell : E \to \mathbb{R}_+$, el problema del cartero chino consiste en encontrar un circuito de longitud mínima que pase por cada arista de $G$ **al menos** una vez.

- Si $G$ es Euleriano, un circuito Euleriano es la solución del problema del cartero chino.
- Existen algoritmos polinomiales para el problema del cartero chino cuando $G$ es orientado o no orientado.
- Sin embargo, no se conocen algoritmos polinomiales si el grafo es mixto (tiene tanto aristas orientadas como aristas no orientadas).

# El juego de Hamilton

En 1859 Hamilton inventó un juego que consistía en encontrar un recorrido de todos los vértices de un dodecaedro sin repetir vértices y volviendo al original.



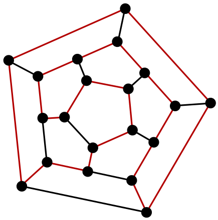
# Definiciones

- Un camino Hamiltoniano en un grafo $G$ es un camino que recorre cada vértice una y sólo una vez.

# Definiciones

- Un camino Hamiltoniano en un grafo $G$ es un camino que recorre cada vértice una y sólo una vez.
- Un circuito Hamiltoniano en un grafo $G$ es un circuito que recorre cada vértice una y sólo una vez.

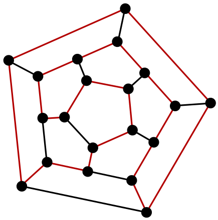
Ej:



# Definiciones

- Un camino Hamiltoniano en un grafo $G$ es un camino que recorre cada vértice una y sólo una vez.
- Un circuito Hamiltoniano en un grafo $G$ es un circuito que recorre cada vértice una y sólo una vez.
- Un grafo es Hamiltoniano si tiene un circuito Hamiltoniano.

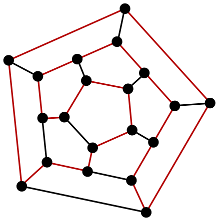
Ej:



# Definiciones

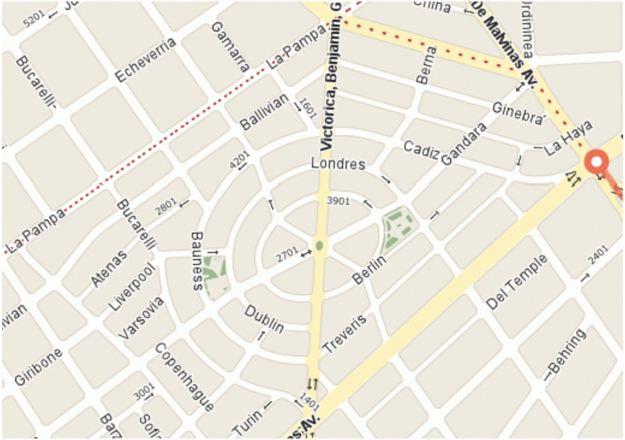
- Un camino Hamiltoniano en un grafo $G$ es un camino que recorre cada vértice una y sólo una vez.
- Un circuito Hamiltoniano en un grafo $G$ es un circuito que recorre cada vértice una y sólo una vez.
- Un grafo es Hamiltoniano si tiene un circuito Hamiltoniano.

Ej:



Obs: Si un grafo no es conexo, no puede tener camino ni circuito Hamiltoniano.

# Grafos Hamiltonianos: Ejemplo



# Grafos Hamiltonianos: Ejemplo

# Grafos Hamiltonianos: Ejemplo

# Grafos Hamiltonianos

**Teorema (condición necesaria).** Sea $G$ un grafo conexo. Si existe $W \subset V$ tal que $G \setminus W$ tiene $c$ componentes conexas con $c > |W|$ entonces $G$ no es Hamiltoniano.

¿Es cierta la recíproca de este teorema?

# Grafos Hamiltonianos

**Teorema (condición necesaria).** Sea $G$ un grafo conexo. Si existe $W \subset V$ tal que $G \setminus W$ tiene $c$ componentes conexas con $c > |W|$ entonces $G$ no es Hamiltoniano.

¿Es cierta la recíproca de este teorema?

**Teorema (Dirac, 1952) (condición suficiente).** Sea $G$ un grafo con $n \geq 3$ y tal que para todo $v \in V$ se verifica que $d(v) \geq n/2$. Entonces $G$ es Hamiltoniano.

¿Es cierta la recíproca de este teorema?

# Grafos Hamiltonianos

- No se conocen condiciones necesarias y suficientes que caractericen en forma “elegante” a los grafos Hamiltonianos.

# Grafos Hamiltonianos

- No se conocen condiciones necesarias y suficientes que caractericen en forma "elegante" a los grafos Hamiltonianos.
- No se conocen algoritmos polinomiales para determinar si un grafo es Hamiltoniano o no (algoritmos de reconocimiento).

# Grafos Hamiltonianos

- No se conocen condiciones necesarias y suficientes que caractericen en forma “elegante” a los grafos Hamiltonianos.

- No se conocen algoritmos polinomiales para determinar si un grafo es Hamiltoniano o no (algoritmos de reconocimiento).

- Más aún, se sospecha que **no existen** (!) algoritmos polinomiales para este problema (¿cómo se demuestra esto?).

# El problema del viajante de comercio (TSP)

- El problema del viajante de comercio se trata de un viajante que debe recorrer lo más pronto posible cierta cantidad de ciudades y volver finalmente a la ciudad donde vive.

# El problema del viajante de comercio (TSP)

- El problema del viajante de comercio se trata de un viajante que debe recorrer lo más pronto posible cierta cantidad de ciudades y volver finalmente a la ciudad donde vive.
- En términos de grafos, es encontrar un camino Hamiltoniano de longitud mínima en un grafo completo con longitudes asociadas a sus aristas.

# El problema del viajante de comercio (TSP)

- El problema del viajante de comercio se trata de un viajante que debe recorrer lo más pronto posible cierta cantidad de ciudades y volver finalmente a la ciudad donde vive.
- En términos de grafos, es encontrar un camino Hamiltoniano de longitud mínima en un grafo completo con longitudes asociadas a sus aristas.
- En su versión de decisión, la entrada es un grafo completo $G$ con longitudes asociadas a sus aristas y un número $k$, y la pregunta es "Existe en $G$ un circuito Hamiltoniano de longitud $\leq k$?".

# El problema del viajante de comercio (TSP)

**Entrada:** Un grafo $G = (V, E)$ completo y una función de distancias $\ell : E \to \mathbb{R}_+$.

**Salida:** Un circuito Hamiltoniano $C \subseteq E$ que minimice la distancia total $\ell(C) = \sum_{ij \in C} \ell(ij)$.

- Se trata de una generalización del problema de camino Hamiltoniano (¿por qué?).

# El problema del viajante de comercio (TSP)

**Entrada:** Un grafo $G = (V, E)$ completo y una función de distancias $\ell : E \to \mathbb{R}_+$.

**Salida:** Un circuito Hamiltoniano $C \subseteq E$ que minimice la distancia total $\ell(C) = \sum_{ij \in C} \ell(ij)$.

- Se trata de una generalización del problema de camino Hamiltoniano (¿por qué?).
- Como consecuencia, no se conocen algoritmos polinomiales para resolver el TSP (¿¿por qué??).

# El problema del viajante de comercio (TSP)

**Entrada:** Un grafo $G = (V, E)$ completo y una función de distancias $\ell : E \to \mathbb{R}_+$.

**Salida:** Un circuito Hamiltoniano $C \subseteq E$ que minimice la distancia total $\ell(C) = \sum_{ij \in C} \ell(ij)$.

- Se trata de una generalización del problema de camino Hamiltoniano (¿por qué?).
- Como consecuencia, no se conocen algoritmos polinomiales para resolver el TSP (¿¿por qué??).
- **Pausa filosófica**:

# El problema del viajante de comercio (TSP)

**Entrada:** Un grafo $G = (V, E)$ completo y una función de distancias $\ell : E \to \mathbb{R}_+$.

**Salida:** Un circuito Hamiltoniano $C \subseteq E$ que minimice la distancia total $\ell(C) = \sum_{ij \in C} \ell(ij)$.

- Se trata de una generalización del problema de camino Hamiltoniano (¿por qué?).
- Como consecuencia, no se conocen algoritmos polinomiales para resolver el TSP (¿¿por qué??).
- **Pausa filosófica**: ... entonces qué hacemos?

# El problema del viajante de comercio (TSP)

**Opciones inmediatas:**

Si las instancias que tenemos que resolver no son muy grandes ...

# El problema del viajante de comercio (TSP)

**Opciones inmediatas:**

Si las instancias que tenemos que resolver no son muy grandes ...

- Un esquema de fuerza bruta puede no ser mala idea.

# El problema del viajante de comercio (TSP)

**Opciones inmediatas:**

Si las instancias que tenemos que resolver no son muy grandes ...

- Un esquema de fuerza bruta puede no ser mala idea.
- En caso contrario, intentar con un backtracking.

# El problema del viajante de comercio (TSP)

**Opciones inmediatas:**

Si las instancias que tenemos que resolver no son muy grandes ...

- Un esquema de fuerza bruta puede no ser mala idea.
- En caso contrario, intentar con un backtracking.

Si las instancias hacen imposibles estos procedimientos ...

# El problema del viajante de comercio (TSP)

**Opciones inmediatas:**

Si las instancias que tenemos que resolver no son muy grandes ...

- Un esquema de fuerza bruta puede no ser mala idea.
- En caso contrario, intentar con un backtracking.

Si las instancias hacen imposibles estos procedimientos ...

- Intentar una heurística golosa.

# El problema del viajante de comercio (TSP)

**Opciones inmediatas:**

Si las instancias que tenemos que resolver no son muy grandes ...

- Un esquema de fuerza bruta puede no ser mala idea.
- En caso contrario, intentar con un backtracking.

Si las instancias hacen imposibles estos procedimientos ...

- Intentar una heurística golosa.
- Generar aleatoriamente muchas soluciones y quedarse con la mejor!

# El problema del viajante de comercio (TSP)

**Opciones inmediatas:**

Si las instancias que tenemos que resolver no son muy grandes ...

- Un esquema de fuerza bruta puede no ser mala idea.
- En caso contrario, intentar con un backtracking.

Si las instancias hacen imposibles estos procedimientos ...

- Intentar una heurística golosa.
- Generar aleatoriamente muchas soluciones y quedarse con la mejor!
- Variantes más sofisticadas: búsqueda local,

# El problema del viajante de comercio (TSP)

**Opciones inmediatas:**

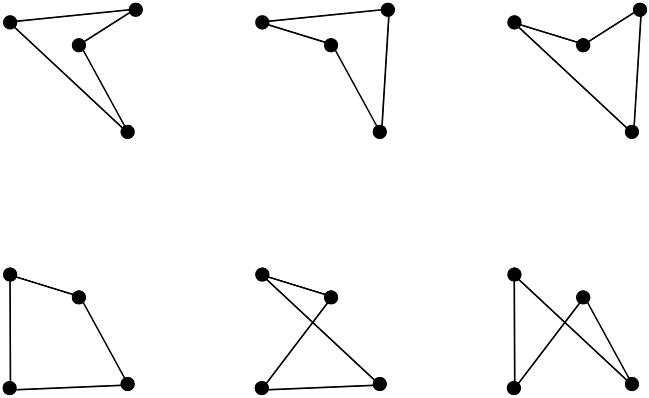
Si las instancias que tenemos que resolver no son muy grandes ...

- Un esquema de fuerza bruta puede no ser mala idea.
- En caso contrario, intentar con un backtracking.

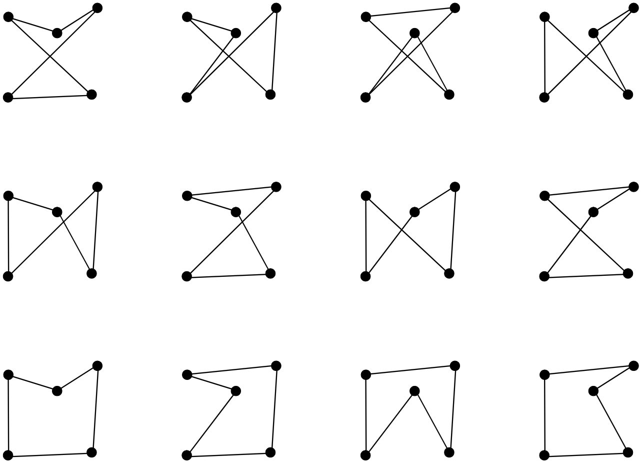
Si las instancias hacen imposibles estos procedimientos ...

- Intentar una heurística golosa.
- Generar aleatoriamente muchas soluciones y quedarse con la mejor!
- Variantes más sofisticadas: búsqueda local, búsqueda tabú, simulated annealing, etc.

# Recorridos con cuatro ciudades



# Recorridos con cinco ciudades



# Recorridos con más de cinco ciudades

- ¿Cuántos recorridos tengo en un caso con 10 ciudades?

# Recorridos con más de cinco ciudades

- ¿Cuántos recorridos tengo en un caso con 10 ciudades?

$$181440$$

# Recorridos con más de cinco ciudades

- ¿Cuántos recorridos tengo en un caso con 10 ciudades?

$$181440$$

- ¿Cuántos recorridos tengo en un caso con 50 ciudades?

## Recorridos con más de cinco ciudades

- ¿Cuántos recorridos tengo en un caso con 10 ciudades?

  181440

- ¿Cuántos recorridos tengo en un caso con 50 ciudades?

  304140932017133780436126081660647688443776415689605120000000000

# Recorridos con más de cinco ciudades

- ¿Cuántos recorridos tengo en un caso con 10 ciudades?

  181440

- ¿Cuántos recorridos tengo en un caso con 50 ciudades?

  30414093201713378043612608166064768844377641568960512000000000

- ¿Cuántos recorridos tengo en un caso con 100 ciudades?

# Recorridos con más de cinco ciudades

- ¿Cuántos recorridos tengo en un caso con 10 ciudades?

  181440

- ¿Cuántos recorridos tengo en un caso con 50 ciudades?

  3041409320171337804361260816606476884437764156896
  0512000000000000

- ¿Cuántos recorridos tengo en un caso con 100 ciudades?

  4666310772197207634084961942813335024535798413219
  0810734296481947608799996614957804470731988078259
  1431268489604136118791255926054584320000000000000
  000000000

# Recorridos con más de cinco ciudades

- ¿Cuántos recorridos tengo en un caso con 10 ciudades?

  181440

- ¿Cuántos recorridos tengo en un caso con 50 ciudades?

  304140932017133780436126081660647688443776415689605120000000000

- ¿Cuántos recorridos tengo en un caso con 100 ciudades?

  466631077219720763408496194281333502453579841321908107342964819476087999966149578044707319880782591431268489604136118791255926054584320000000000000000000000

- ¿Cuántos recorridos tengo en un caso con $n$ ciudades?

# Recorridos con más de cinco ciudades

- ¿Cuántos recorridos tengo en un caso con 10 ciudades?

$$181440$$

- ¿Cuántos recorridos tengo en un caso con 50 ciudades?

  304140932017133780436126081660647688443776415689605120000000000

- ¿Cuántos recorridos tengo en un caso con 100 ciudades?

  466631077219720763408496194281333502453579841321908107342964819476087999966149578044707319880782591431268489604136118791255926054584320000000000000000000000

- ¿Cuántos recorridos tengo en un caso con $n$ ciudades?

$$\frac{(n-1)!}{2}$$

# El problema del viajante de comercio (TSP)

**Hipótesis:** Las distancias cumplen la desigualdad triangular:
$\ell(ij) + \ell(jk) \geq \ell(ik)$ para todo $i, j, k \in V$.

# El problema del viajante de comercio (TSP)

**Hipótesis:** Las distancias cumplen la desigualdad triangular: $\ell(ij) + \ell(jk) \geq \ell(ik)$ para todo $i, j, k \in V$.

- Obtener un árbol generador mínimo $T = (V_T, E_T)$ de $G$.
- Duplicar las aristas de $E_T$, obteniendo un nuevo árbol $T'$.
- Encontrar un circuito Euleriano $C$ en $T'$ (siempre existe!).
- Transformar $C$ en un circuito Hamiltoniano salteando vértices ya visitados.

# El problema del viajante de comercio (TSP)

**Teorema.** Si $\ell_{\text{min}}$ es la longitud de la solución óptima del TSP para la instancia $(G, \ell)$ y $\ell_{\text{heur}}$ es la longitud de la solución generada por la heurística anterior, entonces

$$\frac{\ell_{\text{heur}}}{\ell_{\text{min}}} \leq 2.$$

- En virtud de este teorema, decimos que esta heurística es un algoritmo 2-aproximado.

# El problema del viajante de comercio (TSP)

**Teorema.** Si $\ell_{\text{min}}$ es la longitud de la solución óptima del TSP para la instancia $(G, \ell)$ y $\ell_{\text{heur}}$ es la longitud de la solución generada por la heurística anterior, entonces

$$\frac{\ell_{\text{heur}}}{\ell_{\text{min}}} \leq 2.$$

- En virtud de este teorema, decimos que esta heurística es un algoritmo 2-aproximado.
- ¿Se puede mejorar?

# Heurística de Cristofides (1976)

- Obtener un árbol generador mínimo $T = (V_T, E_T)$ de $G$.
- Sea $I \subseteq V_T$ el conjunto de vértices con grado impar en $T$. Encontrar un matching perfecto de peso mínimo $M$ en el subgrafo de $G$ inducido por $I$.
- Combinar las aristas de $M$ y $T$ para formar un multigrafo $H$.
- Encontrar un circuito Euleriano $C$ en $H$ (siempre existe!).
- Transformar $C$ en un circuito Hamiltoniano salteando vértices ya visitados.

# Heurística de Cristofides (1976)

**Teorema (Cristofides, 1976).** Si $\ell_{\text{min}}$ es la longitud de la solución óptima del TSP para la instancia $(G, \ell)$ y $\ell_{\text{heur}}$ es la longitud de la solución generada por la heurística anterior, entonces

$$\frac{\ell_{\text{heur}}}{\ell_{\text{min}}} \leq 3/2.$$

- ¿Se puede mejorar?

# Heurística de Cristofides (1976)

**Teorema (Cristofides, 1976).** Si $\ell_{\mathrm{min}}$ es la longitud de la solución óptima del TSP para la instancia $(G, \ell)$ y $\ell_{\mathrm{heur}}$ es la longitud de la solución generada por la heurística anterior, entonces

$$\frac{\ell_{\mathrm{heur}}}{\ell_{\mathrm{min}}} \leq 3/2.$$

- ¿Se puede mejorar?
- Si las distancias $\ell$ son euclídeas en el plano, entonces existe un algoritmo $(1 + 1/c)$-aproximado con complejidad $O(n(\log n)^{O(c\sqrt{2})})$.

# El problema del viajante de comercio (TSP)

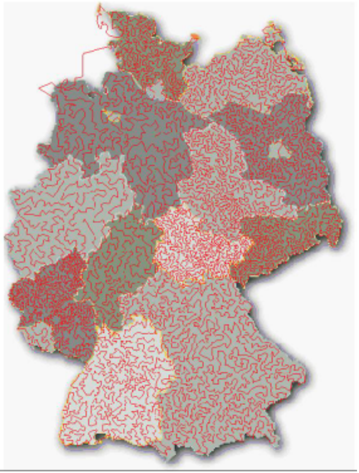
Desde el punto de vista de algoritmos exactos, el enfoque más exitoso a la fecha está dado por algoritmos basados en programación lineal entera.

# Soluciones óptimas para el TSP

- En 1954 Dantzig, Fulkerson y Johnson resolvieron un caso de 49 ciudades del TSP.
- "Resolvieron" significa que D,F&J demostraron que la solución que presentaban era la mejor de un conjunto de 60 decillones de soluciones posibles.

# Solución record (en 2001) de 15112 ciudades de Alemania

- Resuelta en una red de 110 máquinas en las universidades de Rice y Princeton, por Applegate, Bixby, Chvátal y Cook.
- Tiempo total de cómputo de 22.6 años de una PC de 500 MHz.
- Longitud total de aproximadamente 66.000 Km (Un poco más de una vuelta y media a la tierra por el ecuador).



# Solución record (en 2004) de 24978 ciudades de Suecia

- Resuelta por Applegate, Bixby, Chvátal, Cook y Helsgaun.
- Longitud total de aproximadamente 72.500 Km.

| **Año** | **Equipo** | **Ciudades** |
|---|---|---|
| 1954 | G. Dantzig, R. Fulkerson y S. Johnson | 49 |

| **Año** | **Equipo** | **Ciudades** |
|---|---|---|
| 1954 | G. Dantzig, R. Fulkerson y S. Johnson | 49 |
| 1971 | M. Held y R.M. Karp | 64 |
| 1975 | P. M. Camerini, L. Fratta y F. Maffioli | 100 |
| 1977 | M. Grötschel | 120 |

| Año | Equipo | Ciudades |
|---|---|---|
| 1954 | G. Dantzig, R. Fulkerson y S. Johnson | 49 |
| 1971 | M. Held y R.M. Karp | 64 |
| 1975 | P. M. Camerini, L. Fratta y F. Maffioli | 100 |
| 1977 | M. Grötschel | 120 |
| 1980 | H. Crowder y M. W. Padberg | 318 |
| 1987 | M. Padberg y G. Rinaldi | 532 |
| 1987 | M. Grötschel y O. Holland | 666 |
| 1987 | M. Padberg y G. Rinaldi | 2,392 |

| Año | Equipo | Ciudades |
|---|---|---|
| 1954 | G. Dantzig, R. Fulkerson y S. Johnson | 49 |
| 1971 | M. Held y R.M. Karp | 64 |
| 1975 | P. M. Camerini, L. Fratta y F. Maffioli | 100 |
| 1977 | M. Grötschel | 120 |
| 1980 | H. Crowder y M. W. Padberg | 318 |
| 1987 | M. Padberg y G. Rinaldi | 532 |
| 1987 | M. Grötschel y O. Holland | 666 |
| 1987 | M. Padberg y G. Rinaldi | 2,392 |
| 1994 | D. Applegate, R. Bixby, V. Chvátal y W. Cook | 7,397 |
| 1998 | D. Applegate, R. Bixby, V. Chvátal y W. Cook | 13,509 |
| 2001 | D. Applegate, R. Bixby, V. Chvátal y W. Cook | 15,112 |
| 2004 | D. Applegate, R. Bixby, V. Chvátal y W. Cook | 24,978 |
| 2005 | D. Applegate, R. Bixby, V. Chvátal, W. Cook, | 33,810 |
| 2006 | D. Espinoza, M. Goycoolea y K. Helsgaun | |

| **Año** | **Equipo** | **Ciudades** |
|---|---|---|
| 1954 | G. Dantzig, R. Fulkerson y S. Johnson | 49 |
| 1971 | M. Held y R.M. Karp | 64 |
| 1975 | P. M. Camerini, L. Fratta y F. Maffioli | 100 |
| 1977 | M. Grötschel | 120 |
| 1980 | H. Crowder y M. W. Padberg | 318 |
| 1987 | M. Padberg y G. Rinaldi | 532 |
| 1987 | M. Grötschel y O. Holland | 666 |
| 1987 | M. Padberg y G. Rinaldi | 2,392 |
| 1994 | D. Applegate, R. Bixby, V. Chvátal y W. Cook | 7,397 |
| 1998 | D. Applegate, R. Bixby, V. Chvátal y W. Cook | 13,509 |
| 2001 | D. Applegate, R. Bixby, V. Chvátal y W. Cook | 15,112 |
| 2004 | D. Applegate, R. Bixby, V. Chvátal y W. Cook | 24,978 |
| 2005 | D. Applegate, R. Bixby, V. Chvátal, W. Cook, | 33,810 |
| 2006 | D. Espinoza, M. Goycoolea y K. Helsgaun | |

| Año | Equipo | Ciudades |
|---|---|---|
| 1954 | G. Dantzig, R. Fulkerson y S. Johnson | 49 |
| 1971 | M. Held y R.M. Karp | 64 |
| 1975 | P. M. Camerini, L. Fratta y F. Maffioli | 100 |
| 1977 | M. Grötschel | 120 |
| 1980 | H. Crowder y M. W. Padberg | 318 |
| 1987 | M. Padberg y G. Rinaldi | 532 |
| 1987 | M. Grötschel y O. Holland | 666 |
| 1987 | M. Padberg y G. Rinaldi | 2,392 |
| 1994 | D. Applegate, R. Bixby, V. Chvátal y W. Cook | 7,397 |
| 1998 | D. Applegate, R. Bixby, V. Chvátal y W. Cook | 13,509 |
| 2001 | D. Applegate, R. Bixby, V. Chvátal y W. Cook | 15,112 |
| 2004 | D. Applegate, R. Bixby, V. Chvátal y W. Cook | 24,978 |
| 2005 | D. Applegate, R. Bixby, V. Chvátal, W. Cook, | 33,810 |
| 2006 | D. Espinoza, M. Goycoolea y K. Helsgaun | 85,900 |