Tarea 1 \Complejidad computacional

Victor Hugo Gallegos Mota
316160456

José Demian Jiménez Salgado
314291707

Luis Alberto Hernández Aguilar
314208682

25 de agosto de 2022

Ejercicio 1

Supón que un algoritmo A, que resuelve un problema Π, tiene un tiempo de ejecución de $T_A(n) = 20n^3 + 10n + 16$, además supón que un algoritmo B, que resuelve el mismo problema Π, tiene un tiempo de ejecución de $T_B(n) = 22n^2 \log_2(8n)$. ¿Qué algoritmo es mejor?. Justifica tu respuesta respondiendo las siguientes preguntas:

- ¿Para qué valores de n el algoritmo A es mejor que el B ?

  Decimos que $T(n)$ es $O(f(n))$ si hay constantes c y n_0 tales que $T(n)$ ¡= $cf(n)$ para todo $n \geq n_0$. Un programa cuyo tiempo de ejecucion es $O(f(n))$ se dice que tiene una tasa de crecimiento de $f(n)$. En esta notacion $f(n)$ es una cota superior de la tasa de crecimiento de $T(n)$. Entonces la funcion $T_A(n) = 20n^3+10n+16$ es $O(n^3)$ (cubico). Intuitivamente, sólo se considera el término más importante y se ignoran los factores constantes. por lo tanto no hay caso donde el algoritmo A sea mejor que el B pues la tasa de crecimiento es superior considerablemente al introducir un numero mas grande como entrada de n en comparacion a un algoritmo con complejidad cuadrática por la siguiente jerarquia

  $$O(1) \subset O(logn) \subset O(n) \subset O(nlogn) \subset O(n^2) \subset O(n^3) \subset O(2^n)$$

- ¿Para qué valores de n el algoritmo B es mejor que el A ?

  Por la misma razón que mencionamos en el inciso anterior, concluimos que para cualquier valor de n, con $n > 0$, el algoritmo B es mejor que el A.

Ejercicio 2

¿Cuáles de las siguientes afirmaciones son verdades y cuáles falsas? Justifica tu respuesta en cada caso

- Sí $f(n) = n^k$ y $g(n) = c^n$, para algunas constantes $c > 1, k \geq 0$, entonces $g(n) = \Omega(f(n))$

  Usaremos la definicion de $g(n) = \Omega(f(n))$ para resolver este ejercicio

  $\forall n \geq n_0 \rightarrow 0 \leq$ c $\cdot g(n) \leq f(n)$

  $\rightarrow c_k \leq n_k \; \forall n \geq 1$
  $\rightarrow 2c^n \leq 2n^k$
  $\rightarrow 2c^n \; \forall n \geq 1$
  Por lo tanto $2c^n \leq 2n^k \forall n \geq 1$
  $\rightarrow c = 2 \; n_0 = 1 \; 0 \leq$ c $\cdot c^n \leq f(n)$

- Sí $f(n) = n!$ y $g(n) = 2^n$, $f(n) = O(g(n))$

  Procedemos mediante induccion sobe n. **Caso Base:** $4! = 24y2^4 = 16y24 \geq 16$ **Hipóteis de inducción:** $k! \geq 2^k$ **para** $k \geq 4$ **Paso inductivo:** $(k+1)! \geq 2^k + 1$ **para** $k \geq 4$ **Se tiene que** $(k+1)! = (k+1)k!$ **Luego entonces, por H.I se tiene que :**

  $$k! \geq 2$$

  $$(k+1)k! \geq (k+1)2^k$$

  $(k+1)! = (k+1)k! = k \times k! + k! \geq (k+1)2^k = k \times 2^k + 2^k > 2^k + 2^k = 2^k + 1$

  Asi, $\exists c = 1, \exists n_0 = 4$ **tal que** $\forall n \geq n_0$ **entonces 0** $\geq x \times g(n) \geq f(n)$ **, i.e., n!** $\in \Omega(2^n)$

- Sí $f(n) = O(n)$ y $g(n) = \Theta(n)$, entonces:

  - $f(n) + g(n) = O(n)$
  - $f(n)g(n) = O(n^2)$
  - $f(n) - g(n) = O(1)$

  Como $f(n) = O(n)$ y $g(n) = \Theta(n)$, entonces podríamos decir que $f(n) \leq n$ y $g(n) = n$.
  Dado lo anterior, tenemos que $f(n) + g(n) \leq n + n$, lo cual implica que $f(n) + g(n) \leq 2n$, es decir, $f(n) + g(n) = O(2n)$, que es lo mismo que $f(n) + g(n) = O(n)$.
  Luego, $f(n)g(n) \leq n \bullet n$, lo cual implica que $f(n)g(n) \leq n^2$ y, por lo tanto, $f(n)g(n) = O(n^2)$

- Existen funciones f, g tales que no se cumple que $f(n) = O(g(n))$ ni $f(n) = \Omega(g(n))$

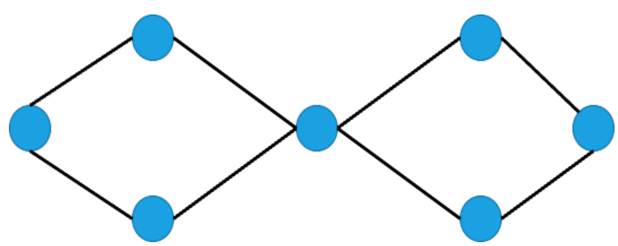
Sean f y g funciones.
Propondremos el caso en el que f=g entonces las funciones seran las siguientes:

f(n)= n y g(n)=n
P.D. que f y g no cumplen que f(n)=O(g(n)) ni f(n)=$\Omega(g(n))$
Aplicamos la definicion de la notacion O(g(n))
$n \leq n \ \forall n_0 \geq 1$
$1n \leq 1n \ \forall n_0 \geq 1$
$1 \cdot 1 \leq 1 \cdot 1$
$1 \leq 1$
entonces $n_0 = 1$ y c=1
Ahora aplicamos la defimicion de $\Omega(g(n))$
$c \cdot g(n) \leq f(n) \ \forall n \geq 1$
$n \leq n$
$1n \leq 1n \ \forall n_0 \geq 1$
$1 \cdot 1 \leq 1 \cdot 1$
$1 \leq 1$
c=1 y $n_0 = 1$
f y g no cumplen con f(n)=O(g(n)) ni f(n)=$\Omega(g(n)$ ya que como las dos funciones son iguales entonces las dos funciones se empalman por lo que esto ocaciona de que no halla costas superior e inferior por lo que f(n)=O(g(n)) ni f(n)=$\Omega(g(n)$ no se cumplen

Ejercicio 3

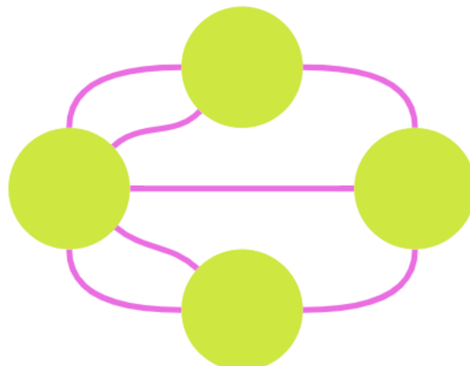
Da ejemplos de gráficas que cumplan lo siguiente:

- Contiene un Ciclo Euleriano, pero no Ciclos Hamiltonianos



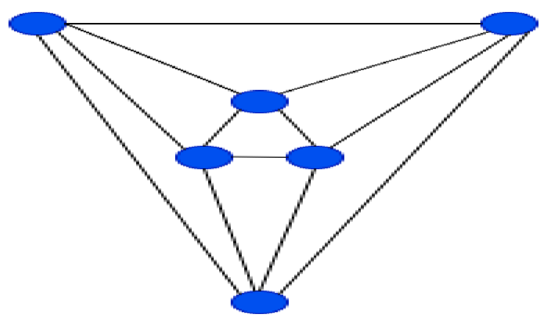
Un ciclo Euleriano utiliza cada vértice exactamente una vez, comenzando y terminando en el mismo vértice, tomando en cuenta que puede visitar los vertices mas de una vez, en cambio un ciclo Hamiltoniano visita cada vértice exactamente una vez, por lo tanto la gráfica propuiesta contine un ciclo Euleriano (se comienza desde cualquier vertice y se recorrera cada nodo 8 veces al volver a donde se comenzó, pero no será Hamiltoniano porque cualquier ruta que visite todos los vértices tendría que visitar el vértice central demasiadas veces.)

- Contiene un Ciclo Hamiltoniano, pero no Ciclos Eulerianos



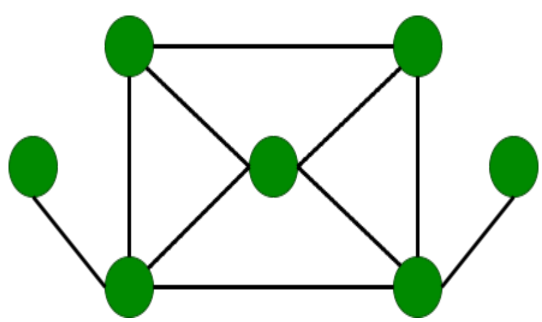
  Contiene un Ciclo Hamiltoniano porque podemos pasar por los vértices al menos una sola vez sin repetirlos y no contiene un Ciclo Euleriano porque no podemos pasar por las aristas al menos una vez sin repetirlas.

- Contiene al menos un Ciclo Hamiltoniano y un Ciclo Euleriano



  Una grafica ens elueriana cuando solo se pueden recorrer sus aristas una sola vez y e la vez es hamiltoniana por que solo se pueden visitar sus vertices una sola vez.

- No contiene Ciclos Hamiltonianos ni Ciclos Eulerianos



Recordando que una gráfica conexa es euleriana si hay un camino cerrado que incluye cada arista de G, tal camino se llama camino euleriano y una gráfica conexa es hamiltoniano si existe un ciclo que incluye todos los vértices de G; tal ciclo se llama ciclo hamiltoniano por lo tanto la grafica prouesta NO es ni euleriano ni hamiltioniano

Ejercicio 4

Investiga en qué consiste el problema de árboles de Steiner. ¿Es un problema de decisión, búsqueda u optimización?

Enuncia una versión del problema de decisión de árboles de Steiner y una versión del problema de optimización de árboles de Steiner. Para cada caso, menciona cuáles son los parámetros, y da algunos ejemplares concretos y sus soluciones.

Los **Problemas del Árbol de Steiner** consisten en encontrar una conexión óptima para un conjunto de objetos y una función objetivo predefinida.
Estos problemas pueden ser tanto de decisión como de optimización.
En gráficas podemos encontrar ambas variantes del problema.
Para entender en qué consisten dichos problemas en gráficas primero debemos saber qué es un árbol de Steiner en gráficas.
Dada una gráfica y un subconjunto de vértices en la gráfica cuyas aristas tiene pesos no negativos, un árbol de Steiner es aquel que se genera a través de los vértices del conjunto y puede contener vértices que no estén en el conjunto pero que sirvan para conectarlos.
Al conjunto de vértices para el árbol de Steiner se le llama **Vértices Terminales** y a los otros vértices usados para construir el árbol de Steiner se les llaman **Vértices de Steiner**.
Ahora sí, el problema de optimización asociado con árboles de Steiner consiste en encontrar el árbol de Steiner de costo mínimo, mientras que para el problema de decisión se tiene que determinar si existe un árbol de Steiner cuyo peso total no exceda un número natural k dado.
Los parámetros para el problema de optimización son: la gráfica, los vértices terminales y los pesos de las aristas. Los parámetros para el problema de optimización son los mismos que para el problema de optimización más el natural k que no debe ser excedido por el peso total del árbol.
Algunos ejemplos aplicados que encontramos son los siguientes:

- **Redes eléctricas**
  Las redes eléctricas encargadas de distribuir la electricidad se pueden modelar mediante un árbol de Steiner teniendo en consideración una serie de restricciones. Dentro de los aspectos a tener en cuenta en la explotación de una red de distribución de energía eléctrica se presenta:

  1. La reducción de las pérdidas por efecto Joule (pérdidas técnicas) resulta ser de vital importancia para mejorar la eficiencia en la prestación del servicio.

2. Las redes de distribución eléctrica deben estar configuradas, en su mayoría, en forma radial para una fácil operación y conveniente coordinación de las protecciones.
3. Se deben satisfacer todos los requerimientos de carga. Estos requisitos conducen a un complejo problema de optimización que, planteados en términos matemáticos, resulta ser NP-Completo. La solución óptima exacta puede ser obtenida al examinar todas las posibles combinaciones de caminos que transporten la electricidad, demandando su resolución un excesivo tiempo de operación computacional.

Por ello, se utilizan en este sector, entre otros, algoritmos genéticos aplicados al problema de Steiner para obtener una solución admisible buena en un tiempo computacional razonable.

- **Infraestructuras**
  La optimización de caminos en el ámbito de la ingeniería civil es la aplicación por excelencia del algoritmo de Steiner. A pesar de haber ganado terreno en diversos campos, el árbol de Steiner sigue siendo primordial en la conexión óptima de ciudades, regiones, países, continentes e incluso en el sector privado para reducir la inversión económica de proyectos y conseguir otros objetivos como el mejor rendimiento posible. En la actualidad, las redes más interesantes en la optimización de recursos son; redes de tuberías de agua o gas, redes eléctricas o carreteras, vías ferroviarias e incluso caminos en minería.
  Entre las aplicaciones nombradas dentro del campo de las infraestructuras, se pasa a la explicación de tres ejemplos; redes ferroviarias, redes eléctricas y redes de minas subterráneas que sirven de referencia para poder construir las redes típicas de este sector.

- **Redes de telecomunicaciones**
  En el último siglo, las telecomunicaciones han tenido un gran impacto en todos los aspectos de la vida. No cabe duda de que la transformación de la revolución industrial a la era de la información ha sido principalmente influenciada por los avances tecnológicos en el campo de las telecomunicaciones.
  La minimización de costes, de energía o encontrar la conexión óptima son ejemplos de objetivos primordiales en los problemas de optimización que también se utilizan en este campo. Mediante el problema de Steiner aplicado a redes de fibra óptica y redes inalámbricas como el enrutamiento multicast, se podrá ver el interés del problema de Steiner en esta rama de la ingeniería.

Ejercicio 5

Estudiamos problemas de optimización en términos de sus versiones en lenguajes. Considera el problema del CLAN, en su versión de optimización*, y muestra que existe un algoritmo de tiempo polinomial para el problema original (de búsqueda) sí y solo sí existe un algoritmo de tiempo polinomial para la versión del problema de decisión.

Sea H una grafica tal que el oraculo nos dice que su H tiene estos vertices $C = \{v1, v2, ..., vn\}$ entonces $n \leq | v |$.

Sea G=H:

Tomamos a un vertice cualquiera de G y preguntamos al oraculo si el vertice v_i pertenece al clan C.

Como toda grafica tiene un clan tribial en el que ∀v ∈ C → v ∈ C entonces si pertenece.

Como el vertice v_i si esta en C entonces preguntamos al oraculo si la grafica G+v_i todavia tiene un clan.

Tenemos los siguientes casos:
Caso 1:
G+v_i todavia tiene un clan entonces tenemos que las aristas que inciden en v_i tambien inciden en otros vertices que siguen en C. Por esto podemos agregarlo sin ningun problema.

Caso 2:
Si G+v_i ya no tiene un clan entonces cualquier clan en G ya no puede tener al vertice v_i

Por lo tanto afirmar que la grafica G' con la que terminamos tiene un clan max por construccion.

Como nuestro oraculo puede decirnos en tiempo polinomial si la grafica tiene un clan , podemos decir que el tiempo que tomara esta acotado por un polinomio P(n). El proceso anterior se repite n veces.

Ahora Este algoritmo de busqueda inicio con toda la grafica y supingamos que ∃ un oraculo que G tiene un clan max. Entonces , el clan x es max? si , no

→ es una Descision

Referencias

[1] Scanftree. (2022). *Eulerian and Hamiltonian Graphs.* 22 de agosto de 2022, de Scanftree.Com en `https://scanftree.com/Graph-Theory/Eulerian-and-Hamiltonian-Graphs{watson53}`

[2] Arooj Fatima. (2022). *What is the Steiner Tree Problem?.* 24 de agosto de 2022, de educative en `https://www.educative.io/answers/what-is-the-steiner-tree-problem`

[3] Wikipedia. (2022). *Steiner tree problem.* 24 de agosto de 2022, de Wikipedia en `https://en.wikipedia.org/wiki/Steiner_tree_problem`

[4] GeeksforGeeks. (2022). *Steiner Tree Problem.* 24 de agosto de 2022, de GeeksforGeeks en `https://www.geeksforgeeks.org/steiner-tree/`

[5] Virginia Martínez Lacañina. (2018). *Aplicaciones en la industria del problema de Steiner y su resolución mediante algoritmos genéticos.* 24 de agosto de 2022, de Virginia Martínez Lacañina en `https://biblus.us.es/bibing/proyectos/abreproy/92106/fichero/TFG-2106-MARTINEZ.pdf`

[6] Apuntes de Estructuras de datos