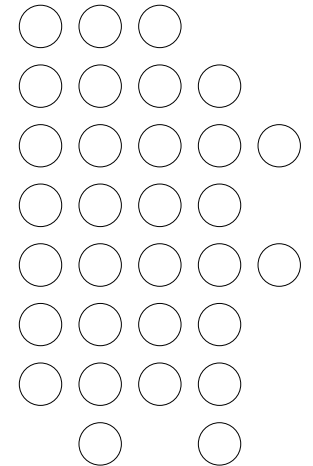


Diseño y análisis de algoritmos

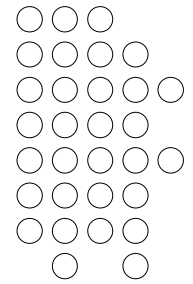


UNIVERSIDAD
DE LOS ANDES

Ejercicios

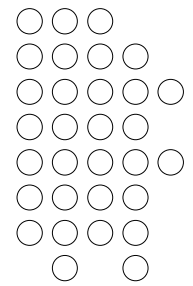


Análisis de algoritmos



1. Diseñar un algoritmo para hacer una búsqueda secuencial de un elemento dado X dentro de un vector $V[a, b]$ y calcular la complejidad en el peor de los casos y en el promedio de los casos.
2. Diseñar un algoritmo para ordenar ascendentemente los elementos de un vector $A [1, n]$ utilizando el método de ordenamiento por inserción lineal. Calcular la complejidad en el peor de los casos, en el mejor de los casos y en el promedio de los casos.

Análisis de algoritmos



3. Diseñar un algoritmo para ordenar ascendentemente los elementos de un vector $A [1, n]$ utilizando el método de ordenamiento por selección lineal. Calcular la complejidad en el peor de los casos, en el mejor de los casos y en el promedio de los casos
4. Diseñar un algoritmo para imprimir todos los subconjuntos de un conjunto finito dado. Ejemplo: dado el conjunto $\{a, b, c\}$ todos los subconjuntos son

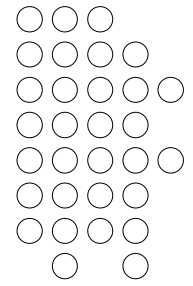
$\{\}$ $\{a\}$ $\{b\}$ $\{c\}$

$\{a, b\}$ $\{a, c\}$ $\{b, c\}$

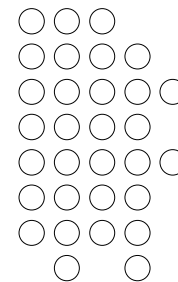
$\{a, b, c\}$

Encontrar las cotas asintóticas de su algoritmo.

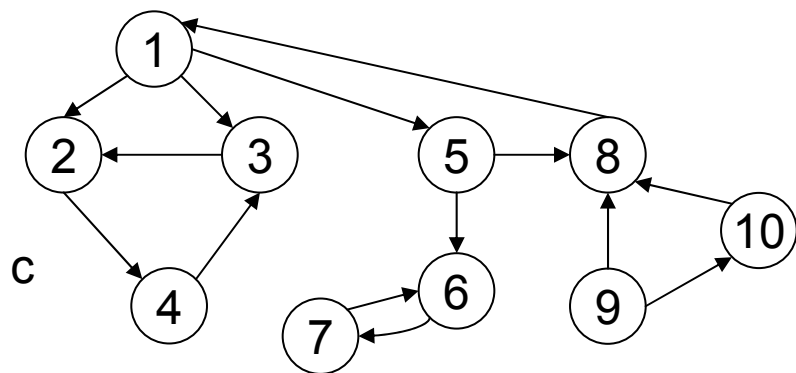
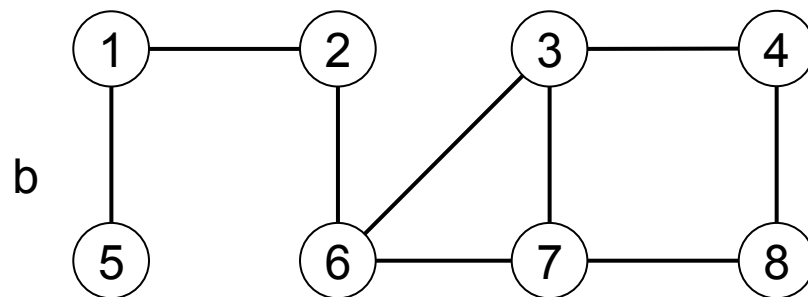
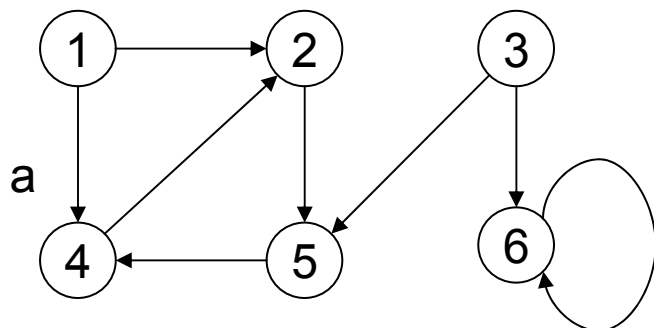
Análisis de algoritmos



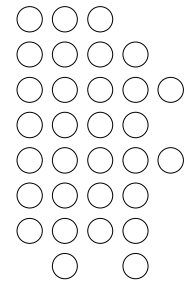
5. Diseñar un algoritmo iterativo para encontrar el número de parejas de conejos al cabo de n meses, dada una pareja inicial que tiene dos descendientes a los dos meses (macho y hembra) y que luego de la primera camada, se sigue reproduciendo con dos descendientes (macho y hembra) cada mes. Cada pareja obtenida se reproduce de la misma manera. Encontrar las cotas asintóticas O , Θ , Ω del algoritmo.



1. Cuál es el resultado de la ejecución del algoritmo de búsqueda en amplitud desde el nodo 3, para:

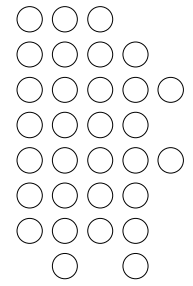


- d Modifique el algoritmo para que despliegue en pantalla cada arco de G

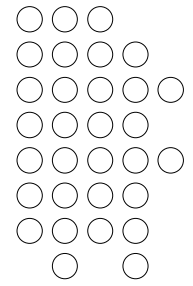


2. El grafo transpuesto $G^t = (\mathbf{N}, A^t)$ donde $A^t = \{(v, u) \in A \times A / (u, v) \in A\}$. Proponga un algoritmo eficiente para calcular G^t a partir de G implementado como:
 - a) Matriz de adyacencia, calcule $T(n)$
 - b) Listas de adyacencia, calcule $T(n)$

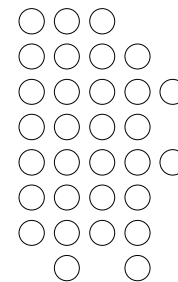
3. El grafo cuadrado $G^2 = (\mathbf{N}, A^2) / (u, w) \in A^2$ si y solo sí para algún $v \in \mathbf{N}$, $(u, v) \in A$ y $(v, w) \in A$. Proponga un algoritmo eficiente para calcular G^2 como:
 - a) Matriz de adyacencia, calcule $T(n)$
 - b) Listas de adyacencia, calcule $T(n)$



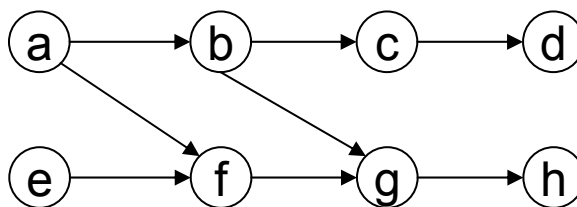
4. Proponga un algoritmo para conocer si un grafo G implementado con matriz de adyacencia contiene o no un nodo con grado de incidencia positivo igual a $|\mathbf{N}|-1$ y grado de incidencia negativo igual a 0 , en $T(n)=O(\mathbf{N})$
5. Realice otro algoritmo para calcular lo mismo que el ejercicio anterior, pero con G implementado con listas de adyacencia
6. Realice un algoritmo eficiente para determinar si un grafo G es bipartito

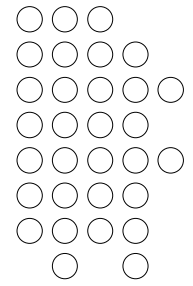


7. Proponga un algoritmo en $T(n)=O(N)$ para saber si un grafo G contiene o no un ciclo
8. ¿Cuál es $W(n)$ para la implementación del TAD Digrafo implementado con matriz de adyacencia y cuál la de listas de adyacencia? Compare sus resultados
9. Realice un algoritmo que liste todos los órdenes topológicos de un dag G dado



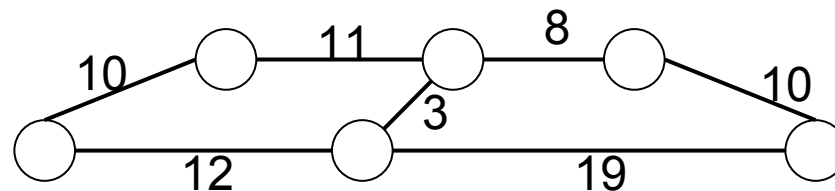
10. La longitud (# de aristas) del camino más corto entre u y v en G es $d(u, v)$. Si u y v no están conectados $d(u, v) = \infty$. Defina la relación “ u está cerca de v ” cuando $d(u, v) \leq 10$. ¿Cuál de las propiedades (reflexividad, simetría, antisimetría y transitividad) está presente en G ?
11. Verifique que el dag G acíclico tiene 47 órdenes topológicos

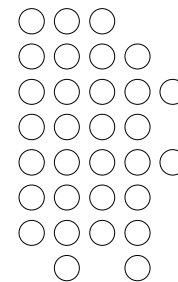




12. Haga un algoritmo no recursivo para la búsqueda en profundidad en G
13. Una región contiene ciudades conectadas por carreteras. Se desea construir un hospital en una de ellas, para que sirva a todas las ciudades, minimizando la distancia máxima desde cualquiera del resto de las ciudades hasta la escogida. Haga un algoritmo para resolver el problema, dado G con etiquetas no negativas.

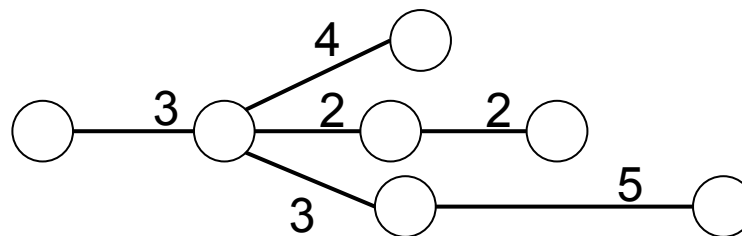
Pruébalo con el grafo

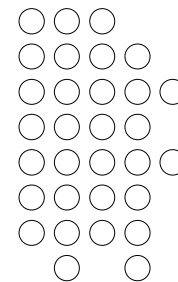




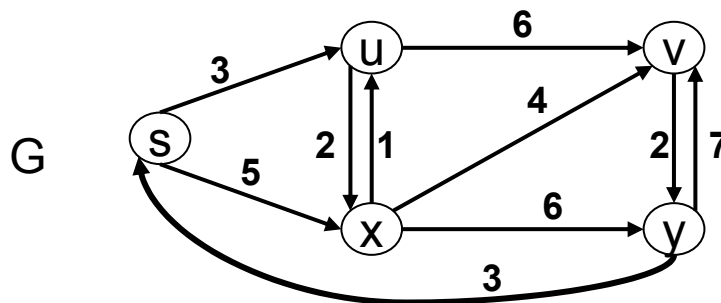
14. Haga un algoritmo para determinar cuáles aristas de un G conexo se pueden eliminar para que G sea no conexo
15. Sea T un árbol no dirigido etiquetado y defina el diámetro de T como el costo del camino más largo en T . Realice un algoritmo de complejidad lineal para encontrar el diámetro de T

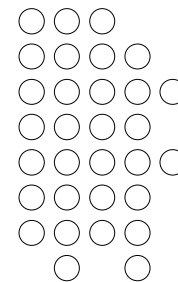
Pruébalo con el árbol



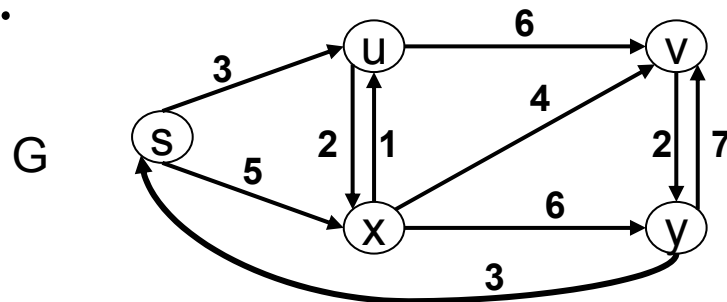


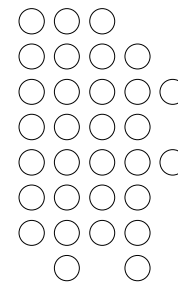
17. Suponga que los pesos de G están en $[1, |N|]$
 ¿Qué tan rápido corre el algoritmo de Kruskal?
 ¿Qué tan rápido corre si los pesos están en $[1, W]$,
 con $W > 0$?
18. Igual al ejercicio anterior, pero con el algoritmo de Prim
19. Para el digrafo G , encuentre 4 árboles de caminos mínimo



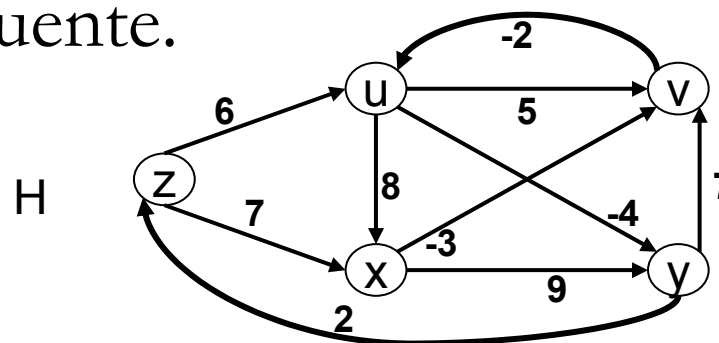


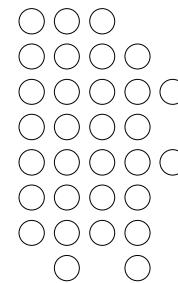
20. Pruebe el algoritmo de Dijkstra para el digrafo G teniendo como nodo inicial s . Pruebe nuevamente con el nodo inicial y . Muestre los valores de d y p en cada vuelta del lazo.
21. De un ejemplo de un grafo con pesos negativos donde el algoritmo de Dijkstra de resultados incorrectos.





22. Pruebe el algoritmo de Bellman-Ford para el digrafo H usando como nodo inicial y. Relaje los arcos en el orden lexicográfico y despliegue d y p en cada pasada.
23. Para el mismo digrafo H, cambie el peso de (y, v) por 4 y pruebe el algoritmo de Bellman-Ford usando z como nodo fuente.





24. Pruebe el algoritmo de C+Cdag para el digrafo Q usando como nodo fuente r.

