



**UNIVERSIDAD NACIONAL
AUTÓNOMA DE MÉXICO**



Facultad de Ingeniería

Estructuras Discretas

Grupo 06

Profesor(a): Ing. Orlando Zaldívar Zamorategui

Alumno: David Romero Molina

CUESTIONARIO DEL ALGORITMO DE DIJKSTRA

1. ¿Qué tipo de grafo puede ser resuelto por el algoritmo de Dijkstra?
 - a) Grafo no dirigido
 - b) Grafo dirigido con pesos negativos
 - c) Grafo con ciclos negativos
 - d) Grafo conexo y ponderado **

2. ¿Qué estructura de datos se utiliza en el algoritmo de Dijkstra para llevar un registro de los vértices visitados?
 - a) Pila
 - b) Cola
 - c) Set
 - d) Heap **

3. ¿Cuál es el objetivo del algoritmo de Dijkstra?
 - a) Encontrar la ruta más corta entre dos vértices en un grafo ponderado
 - b) Encontrar el camino más largo entre dos vértices en un grafo ponderado
 - c) Encontrar todos los caminos posibles entre dos vértices en un grafo ponderado
 - d) Encontrar la ruta más corta entre un vértice origen y todos los demás vértices en un grafo ponderado **

4. ¿Cuál es el tiempo de complejidad del algoritmo de Dijkstra?
 - a) $O(V+E)$
 - b) $O(V \log V)$
 - c) $O(E \log V)$
 - d) $O(V^2)$ **

5. ¿Qué algoritmo es similar al algoritmo de Dijkstra, pero es más rápido en grafos densos?
 - a) Algoritmo de Bellman-Ford
 - b) Algoritmo de Floyd-Warshall
 - c) Algoritmo de Prim
 - d) Algoritmo de Johnson **

6. ¿Cuál es la principal desventaja del algoritmo de Dijkstra?
 - a) No siempre encuentra la ruta más corta
 - b) Es muy lento en grafos densos
 - c) Requiere que los pesos de las aristas sean no negativos **
 - d) Solo funciona en grafos conexos

7. ¿Qué hace el algoritmo de Dijkstra si encuentra una arista con un peso negativo?
 - a) Continúa su ejecución normalmente
 - b) Termina la ejecución y devuelve un error
 - c) Corrige el peso de la arista y continua su ejecución
 - d) No funciona en grafos con pesos negativos **

8. ¿Qué es la lista de prioridad utilizada en el algoritmo de Dijkstra?
- a) Una lista que mantiene los vértices visitados
 - b) Una lista que mantiene la distancia más corta conocida desde el origen a cada vértice
 - c) Una lista que mantiene los vértices por visitar
 - d) Una cola de prioridad que mantiene los vértices por visitar, ordenados por su distancia desde el origen **
9. ¿Qué tipo de búsqueda utiliza el algoritmo de Dijkstra?
- a) Búsqueda en profundidad
 - b) Búsqueda en anchura
 - c) Búsqueda de costo uniforme
 - d) No utiliza una búsqueda específica **
10. ¿Cuál es el mejor caso de complejidad del algoritmo de Dijkstra?
- a) $O(V)$
 - b) $O(E)$
 - c) $O(1)$
 - d) $O(\log V)$ **
11. ¿Cómo se inicializan las distancias en el algoritmo de Dijkstra?
- a) Todas las distancias se inicializan en cero
 - b) Todas las distancias se inicializan en infinito
 - c) Las distancias de los vértices adyacentes al origen se inicializan en cero, y las demás en infinito **
 - d) Las distancias se inicializan con valores aleatorios
12. ¿Cuál es el nombre del paso donde se actualizan las distancias mínimas en el algoritmo de Dijkstra?
- a) Paso de inicialización
 - b) Paso de selección
 - c) Paso de relajación **
 - d) Paso de terminación
13. ¿Qué es una cota inferior en el contexto del algoritmo de Dijkstra?
- a) La distancia entre un vértice y su vecino más cercano
 - b) La distancia más corta conocida desde el origen a un vértice
 - c) Una estimación conservadora de la distancia más corta desde el origen a un vértice, que se utiliza para ordenar la lista de prioridad **
 - d) Un límite superior para la distancia entre dos vértices
14. ¿Qué es una cola de Fibonacci?
- a) Una estructura de datos utilizada para mantener los vértices visitados
 - b) Una estructura de datos utilizada para mantener las distancias más cortas conocidas
 - c) Una estructura de datos utilizada para mantener los vértices por visitar
 - d) Una estructura de datos utilizada para implementar la lista de prioridad en el algoritmo de Dijkstra **

15. ¿Qué es la complejidad espacial del algoritmo de Dijkstra?

- a) $O(V+E)$
- b) $O(V \log V)$
- c) $O(E \log V)$
- d) $O(V)$ **

16. ¿Qué es el camino más corto en el contexto del algoritmo de Dijkstra?

- a) Un camino que visita todos los vértices del grafo
- b) Un camino que visita los vértices más cercanos al origen
- c) Un camino que visita los vértices en orden lexicográfico
- d) Un camino que visita los vértices más cercanos al origen y cuya longitud es la menor posible **

17. ¿Qué es la matriz de adyacencia en el contexto del algoritmo de Dijkstra?

- a) Una matriz que contiene la distancia más corta conocida desde el origen a cada vértice
- b) Una matriz que contiene los vértices adyacentes a cada vértice
- c) Una matriz que contiene los pesos de las aristas entre cada par de vértices **
- d) Una matriz que contiene la distancia entre cada par de vértices

18. ¿Qué es una arista negativa en el contexto del algoritmo de Dijkstra?

- a) Una arista cuyo peso es cero
- b) Una arista cuyo peso es menor que cero
- c) Una arista cuyo peso es mayor que cero
- d) Una arista que no tiene peso **

19. ¿Por qué el algoritmo de Dijkstra no funciona en grafos con aristas negativas?

- a) Porque el algoritmo utiliza una cota inferior que no puede ser negativa
- b) Porque el algoritmo utiliza una cola de prioridad que no puede manejar valores negativos
- c) Porque el algoritmo no puede garantizar que el camino más corto no pase por una arista negativa **
- d) Porque el algoritmo no puede manejar ciclos negativos

20. ¿Cuál es la complejidad temporal del algoritmo A* en el peor caso?

- a) $O(V+E)$
- b) $O(V \log V)$
- c) $O(E \log V)$
- d) Depende de la calidad de la heurística utilizada y del tamaño del grafo **

21. ¿Qué son los grafos y dígrafos y cómo se utilizan en informática?

- a) Son estructuras de datos que implementan el grafo o el dígrafo ADT y se utilizan para modelar problemas en informática.
- b) Son estructuras de datos que implementan el grafo o el dígrafo ADT y se utilizan para modelar problemas en matemáticas.
- c) Son estructuras de datos que implementan el grafo o el dígrafo ADT y se utilizan

para modelar problemas en física.

d) Son estructuras de datos que implementan el grafo o el dígrafo ADT y se utilizan para modelar problemas en química.

22. ¿Qué papel desempeñan los grafos y dígrafos en Internet y las redes de comunicación?

a) No tienen ningún papel en Internet y las redes de comunicación.

b) Desempeñan un papel importante en la estructura subyacente de estas redes y se utilizan para modelarlas.

c) Desempeñan un papel importante en la comunicación de datos en estas redes.

d) Desempeñan un papel secundario en la estructura subyacente de estas redes.

23. ¿Qué problemas fundamentales para grafos ponderados con amplias aplicaciones se mencionan en el texto?

a) El problema del árbol mínimo y el problema del camino más largo.

b) El problema del árbol máximo y el problema del camino más largo.

c) El problema del árbol mínimo y el problema del camino más corto.

d) El problema del árbol máximo y el problema del camino más corto

24. ¿Qué es el peso de un árbol de expansión?

a) La suma de los pesos de las aristas del grafo G

b) La suma de los pesos de las aristas del árbol de expansión T

c) La multiplicación de los pesos de las aristas del grafo G

d) La multiplicación de los pesos de las aristas del árbol de expansión T

25. ¿Qué es un árbol de expansión mínima?

a) Un árbol de expansión que tiene un peso mínimo sobre todos los árboles de expansión de G.

b) Un árbol de expansión que tiene un peso máximo sobre todos los árboles de expansión de G.

c) Un árbol de expansión que tiene el mismo peso que todos los árboles de expansión de G.

d) Un árbol de expansión que no tiene peso.

26. ¿Por qué es inviable una búsqueda enumerativa por fuerza bruta para encontrar árboles de expansión en grafos?

a) Porque el número de árboles de expansión suele ser muy pequeño, incluso para grafos grandes.

b) Porque el número de árboles de expansión suele ser muy grande, incluso para grafos pequeños.

c) Porque no es posible hacer una búsqueda enumerativa por fuerza bruta en grafos.

d) Porque la búsqueda enumerativa por fuerza bruta es el método más eficiente para encontrar árboles de expansión en grafos.

27. ¿Qué algoritmo se estudiará en esta sección para determinar la trayectoria más corta entre dos vértices en una gráfica ponderada?

a) Algoritmo de Kruskal

b) Algoritmo de Prim

c) Algoritmo de Dijkstra

d) Algoritmo de Floyd-Warshall

28. ¿Qué algoritmo se puede utilizar para encontrar el camino más corto entre dos ciudades en un mapa de carreteras ponderado?

a) Algoritmo de búsqueda en amplitud

b) Algoritmo de búsqueda en profundidad

c) Algoritmo de Floyd

d) Algoritmo de Dijkstra

29. ¿Por qué no se pueden utilizar los algoritmos de búsqueda en amplitud y búsqueda en profundidad directamente en un grafo ponderado?

a) Porque suponen que todas las aristas del grafo tienen un costo de 1

b) Porque son ineficientes en grafos ponderados

c) Porque solo funcionan en grafos no ponderados

d) Porque solo encuentran caminos más cortos entre nodos adyacentes

30. ¿Qué es necesario hacer antes de encontrar la longitud del camino mínimo desde un nodo hasta otro en un grafo ponderado?

a) Generar las longitudes de los caminos mínimos desde cada nodo hasta todos los demás nodos del grafo

b) Seleccionar el nodo final v_n y darle el valor cero a la distancia más corta desde ese nodo hasta sí mismo

c) Seleccionar el nodo inicial v_1 y darle el valor cero a la distancia más corta desde ese nodo hasta sí mismo

d) Ignorar todos los nodos que están más alejados de v_1

31. ¿Qué es el conjunto S en el algoritmo de Dijkstra?

a) El conjunto de nodos que se han encontrado sucesivamente más alejados de v_1 y se han ignorado en cuanto a su distancia mínima a v_1

b) El conjunto de nodos que están inmediatamente más próximos a v_n

c) El conjunto de nodos que están inmediatamente más próximos a v_1

d) El conjunto de nodos que no se han visitado aún en el algoritmo de Dijkstra

32. ¿Qué es el algoritmo de Floyd y en qué se basa?

a) Es un algoritmo para hallar el camino más corto en un grafo y se basa en la representación matricial de una lista de adyacencia del grafo.

b) Es un algoritmo para hallar todas las distancias más cortas y todos los caminos de un grafo y se basa en una representación de matriz de adyacencia del grafo.

c) Es un algoritmo para hallar el camino más corto en un grafo y se basa en la representación matricial de un grafo denso.

d) Es un algoritmo para hallar todas las distancias más cortas y todos los caminos de un grafo y se basa en una representación de matriz de incidencia del grafo.

33. ¿Por qué las búsquedas en amplitud y en profundidad no son utilizables directamente para encontrar el camino más corto en un grafo ponderado?

- a) Porque suponen que todos las aristas del grafo tienen un coste de 1.
- b) Porque no son capaces de manejar grafos densos.
- c) Porque no están diseñadas para trabajar con grafos ponderados.
- d) Porque solo son útiles para encontrar el camino más corto desde el nodo inicial hasta un único nodo destino.

34. ¿Cuál es la ventaja del enfoque de Dijkstra en comparación con el algoritmo de Floyd para grafos dispersos?

- a) Es más fácil de implementar.
- b) Es capaz de manejar grafos densos.
- c) Es más rápido en términos de análisis temporal.
- d) Es capaz de encontrar todos los caminos de un grafo.

35. ¿Qué sucede si no hay ningún camino desde v_1 hasta un cierto nodo en el enfoque de Dijkstra?

- a) La distancia actual de ese nodo se establece en cero.
- b) La distancia actual de ese nodo se establece en infinito.
- c) El algoritmo Dijkstra deja de funcionar.
- d) El algoritmo de Floyd se utiliza en su lugar.

36. ¿Qué es un grafo ponderado?

- a) Un grafo en el que todos los vértices tienen el mismo peso.
- b) Un grafo en el que cada vértice tiene un peso diferente.
- c) Un grafo en el que todas las aristas tienen el mismo peso.
- d) Un grafo en el que cada arista tiene un peso diferente.

37. ¿Cómo se asignan las etiquetas a los vértices en el algoritmo de Dijkstra?

- a) Por orden alfabético.
- b) Por orden de llegada.
- c) Por medio de un procedimiento iterativo.
- d) Al azar.

38. ¿Cuál es el conjunto V_1 en la primera iteración del algoritmo de Dijkstra?

- a) $V_0 - \{v_d\}$
- b) $V_0 + \{v_d\}$
- c) $V_0 \times \{v_d\}$
- d) $V_0 / \{v_d\}$

39. ¿Cómo se identifica la trayectoria más corta en el algoritmo de Dijkstra?

- a) Yendo de atrás hacia delante desde el vértice inicial hasta el vértice final.
- b) Yendo de atrás hacia delante desde el vértice final hasta el vértice inicial.
- c) Yendo de adelante hacia atrás desde el vértice inicial hasta el vértice final.
- d) Yendo de adelante hacia atrás desde el vértice final hasta el vértice inicial.

40. ¿Qué es lo que busca el algoritmo de Dijkstra en una gráfica ponderada?

- a) La trayectoria más larga entre dos vértices

b) La trayectoria más corta entre dos vértices *

c) El número de vértices en la gráfica

d) La cantidad de aristas en la gráfica

41. ¿Qué representa la etiqueta temporal de un vértice en el algoritmo de Dijkstra?

a) La longitud de la trayectoria más corta hasta el vértice

b) La longitud de la trayectoria más larga hasta el vértice

c) La cantidad de aristas hasta el vértice

d) Ninguna de las anteriores *

42. ¿En qué consiste la iteración inicial del algoritmo de Dijkstra?

a) Asignar etiquetas permanentes a todos los vértices de la gráfica

b) Asignar etiquetas temporales a todos los vértices de la gráfica

c) Asignar etiqueta permanente al vértice inicial y etiquetas temporales a todos los demás *

d) Ninguna de las anteriores

43. ¿Qué sucede en cada iteración del algoritmo de Dijkstra?

a) Se revisan las etiquetas permanentes de los vértices y se actualizan las temporales

b) Se revisan las etiquetas temporales de los vértices y se actualizan si es necesario *

c) Se eliminan vértices de la gráfica

d) Ninguna de las anteriores

44. ¿Cómo se determina la longitud de la trayectoria más corta entre dos vértices en el algoritmo de Dijkstra?

a) Se suman los pesos de todas las aristas en la trayectoria más corta

b) Se toma la etiqueta permanente del vértice final

c) Se va de atrás hacia adelante desde el vértice final, incluyendo aquellos vértices etiquetados permanentemente *

d) Ninguna de las anteriores

45. ¿Qué sucede si un vértice no alcanzable tiene una etiqueta temporal diferente de infinito?

a) Se considera como un error en el algoritmo.

b) Se considera como un vértice aislado.

c) Se considera como un vértice no alcanzable y se ignora en la búsqueda de la trayectoria más corta.

d) Ninguna de las anteriores.

46. ¿Qué es el peso de una arista en el contexto del algoritmo de Dijkstra?

a) El número de vértices que conecta la arista.

b) La longitud de la arista medida en unidades de tiempo.

c) El costo asociado a recorrer esa arista.

d) Ninguna de las anteriores.

47. ¿Cuál es la complejidad temporal del algoritmo de Dijkstra para grafos densos?

a) $O(n \log n)$

b) $O(n^2)$

c) $O(n^3)$

d) $O(2^n)$

48. ¿Cómo se identifica la trayectoria más corta en el algoritmo de Dijkstra?

a) Se sigue la ruta de etiquetas permanentes desde el vértice final hasta el inicial.

b) Se sigue la ruta de etiquetas temporales desde el vértice inicial hasta el final.

c) Se sigue la ruta de etiquetas permanentes desde el vértice inicial hasta el final.

d) Ninguna de las anteriores.

49. ¿Cuál es la principal ventaja del algoritmo de Dijkstra sobre el algoritmo de Bellman-Ford?

a) Mayor precisión en la determinación de la trayectoria más corta.

b) Menor complejidad temporal en grafos densos.

c) Capacidad para manejar grafos con ciclos negativos.

d) Ninguna de las anteriores.

50. ¿Cuál es la complejidad temporal del algoritmo de Dijkstra en el peor de los casos para encontrar la ruta más corta entre dos vértices en una gráfica ponderada con V vértices y E aristas?

a) $O(V \log V)$

b) $O(E \log V)$

c) $O(V^2)$

d) $O(E + V \log V)$