Terminología – Dado el grafo dirigido acíclico de la Figura 1, completar las siguientes sentencias:

a) Los vértices 5 y 6 son adyacentes al vértice 3.

b) El grado del vértice 4 es: 3

c) La secuencia de vértices 3, 5, 4, 2 es un camino desde 3 a 2.

d) La longitud del camino más corto desde 3 a 2 es: 3

e) Insertando en el DAG el arco (2,5) deja de serlo, pues se forma un ciclo entre los vértices

2, 5 y 4

f) Los vértices 1 y 3 tienen grado\_in igual a 0 (cero) y grado igual a: 2

g) El grado\_out del vértice 4 es: 1 y el del vértice 2 es: 0

h) La secuencia de vértices 3, 6, 5, 4, 2 es el camino más largo desde 3 a 2.

i) Enuncie 5 sentencias usando términos o definiciones que no hayan sido expuestas en la

sentencias anteriores :

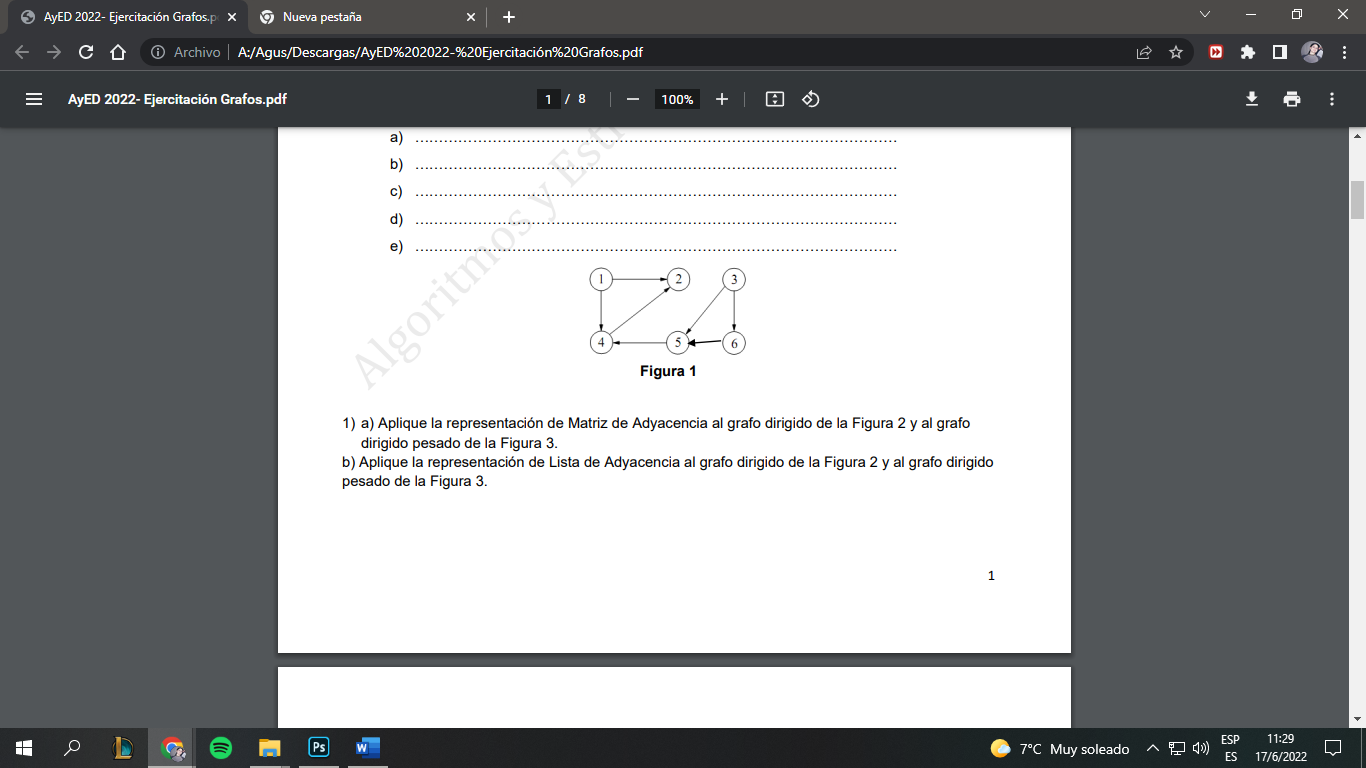
a) El grado\_out del vertice 5 es 1

b) Insertando en el grafo el arco (6,6) se forma un bucle.

c) No se trata de un grafo ponderado

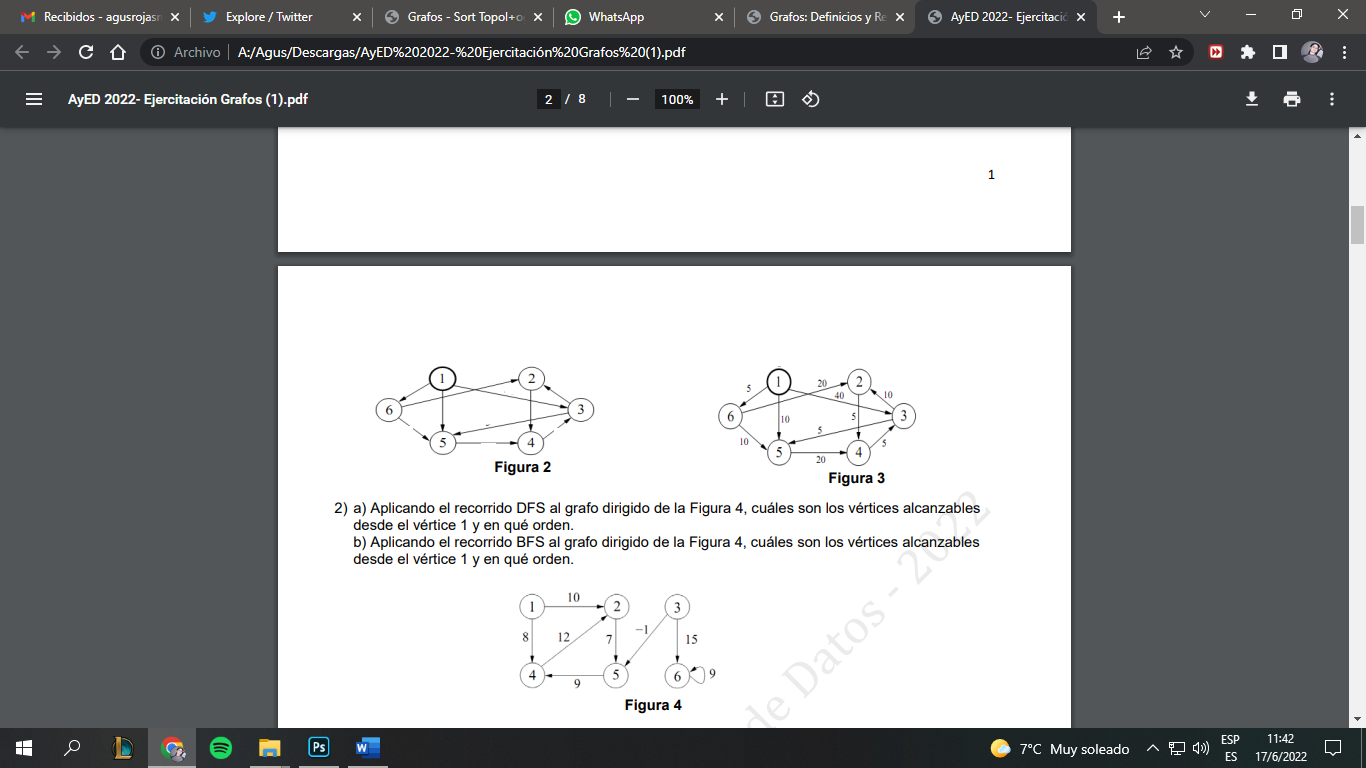
d) No se trata de un grafo fuertemente conexo

e) El grado del grafo es 3.



a) Aplique la representación de Matriz de Adyacencia al grafo dirigido de la Figura 2 y al grafo dirigido pesado de la Figura 3.

b) Aplique la representación de Lista de Adyacencia al grafo dirigido de la Figura 2 y al grafo dirigido pesado de la Figura 3.



MATRIZ FIGURA 2

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 |
| 1 | 0 | 0 | 1 | 0 | 1 | 1 |
| 2 | 0 | 0 | 0 | 1 | 0 | 0 |
| 3 | 0 | 1 | 0 | 0 | 1 | 0 |
| 4 | 0 | 0 | 1 | 0 | 0 | 0 |
| 5 | 0 | 0 | 0 | 1 | 0 | 0 |
| 6 | 0 | 1 | 0 | 0 | 1 | 0 |

MATRIZ FIGURA 3

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 |
| 1 | 0 | 0 | 40 | 0 | 10 | 5 |
| 2 | 0 | 0 | 0 | 5 | 0 | 0 |
| 3 | 0 | 10 | 0 | 0 | 5 | 0 |
| 4 | 0 | 0 | 5 | 0 | 0 | 0 |
| 5 | 0 | 0 | 0 | 20 | 0 | 0 |
| 6 | 0 | 20 | 0 | 0 | 10 | 0 |

LISTA FIGURA 2

6

5

5

2

4

4

5

2

3

3

|  |
| --- |
| 1 |
| 2 |
| 3 |
| 4 |
| 5 |
| 6 |

LISTA FIGURA 3

6 | 5

5| 10

2 | 20

3 | 5

4 | 20

5| 5

2 | 10

4 | 5

5 | 10

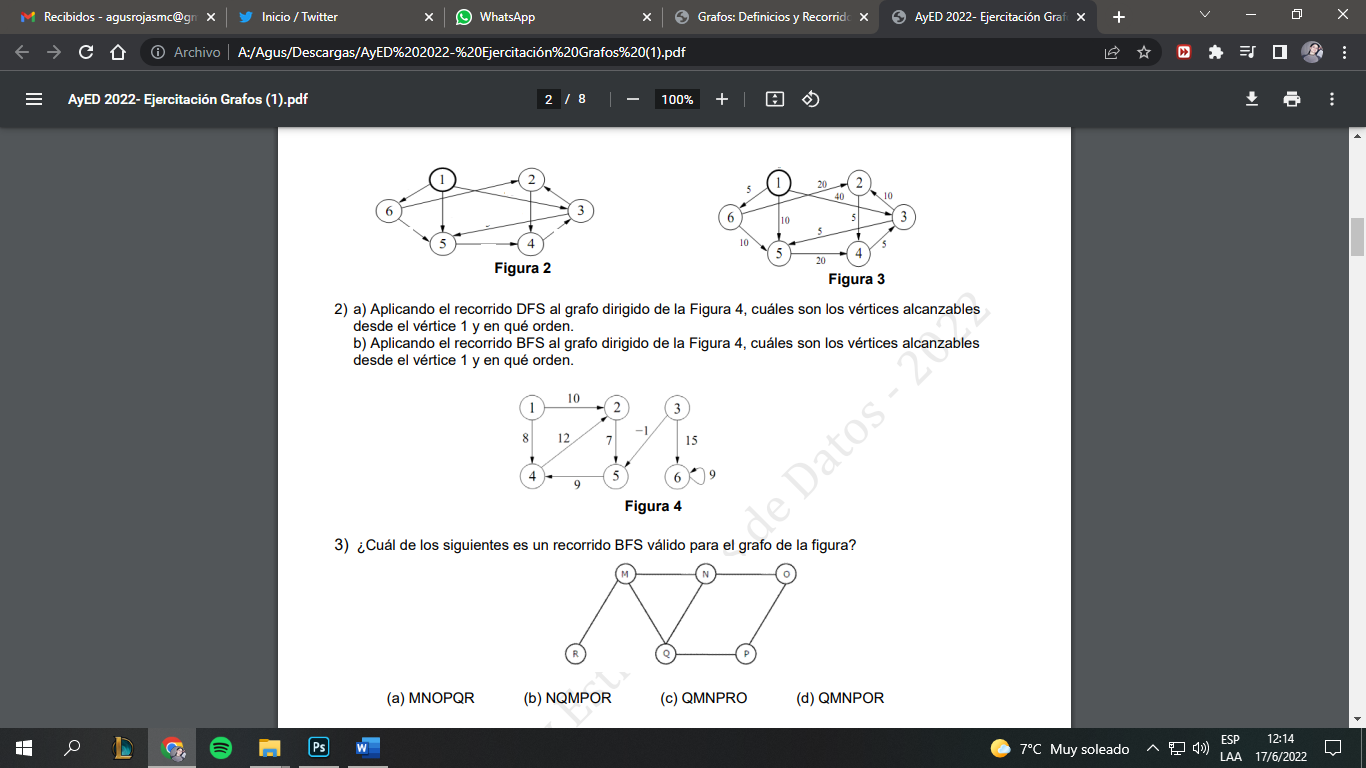
3 | 40

|  |
| --- |
| 1 |
| 2 |
| 3 |
| 4 |
| 5 |
| 6 |

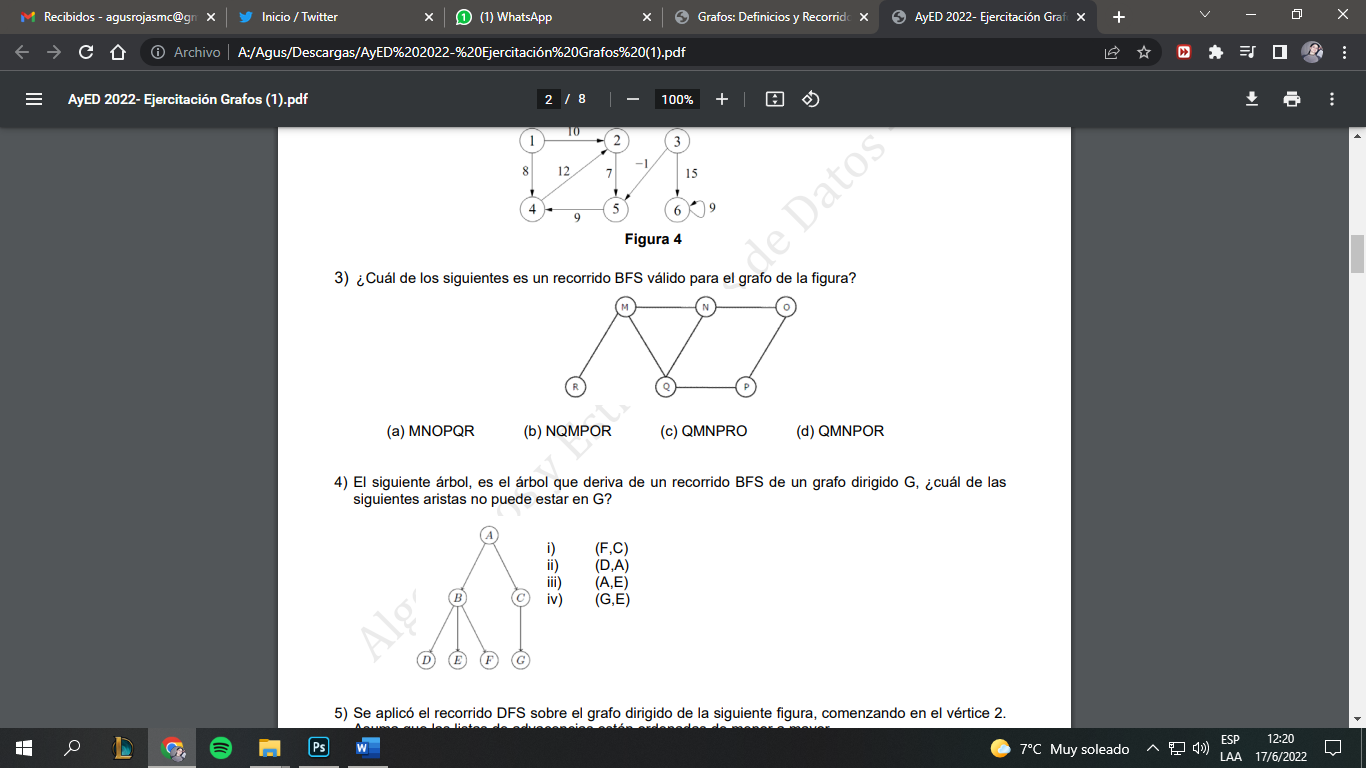
2)

a) Aplicando el recorrido DFS al grafo dirigido de la Figura 4, cuáles son los vértices alcanzables desde el vértice 1 y en qué orden.

b) Aplicando el recorrido BFS al grafo dirigido de la Figura 4, cuáles son los vértices alcanzables desde el vértice 1 y en qué orden.

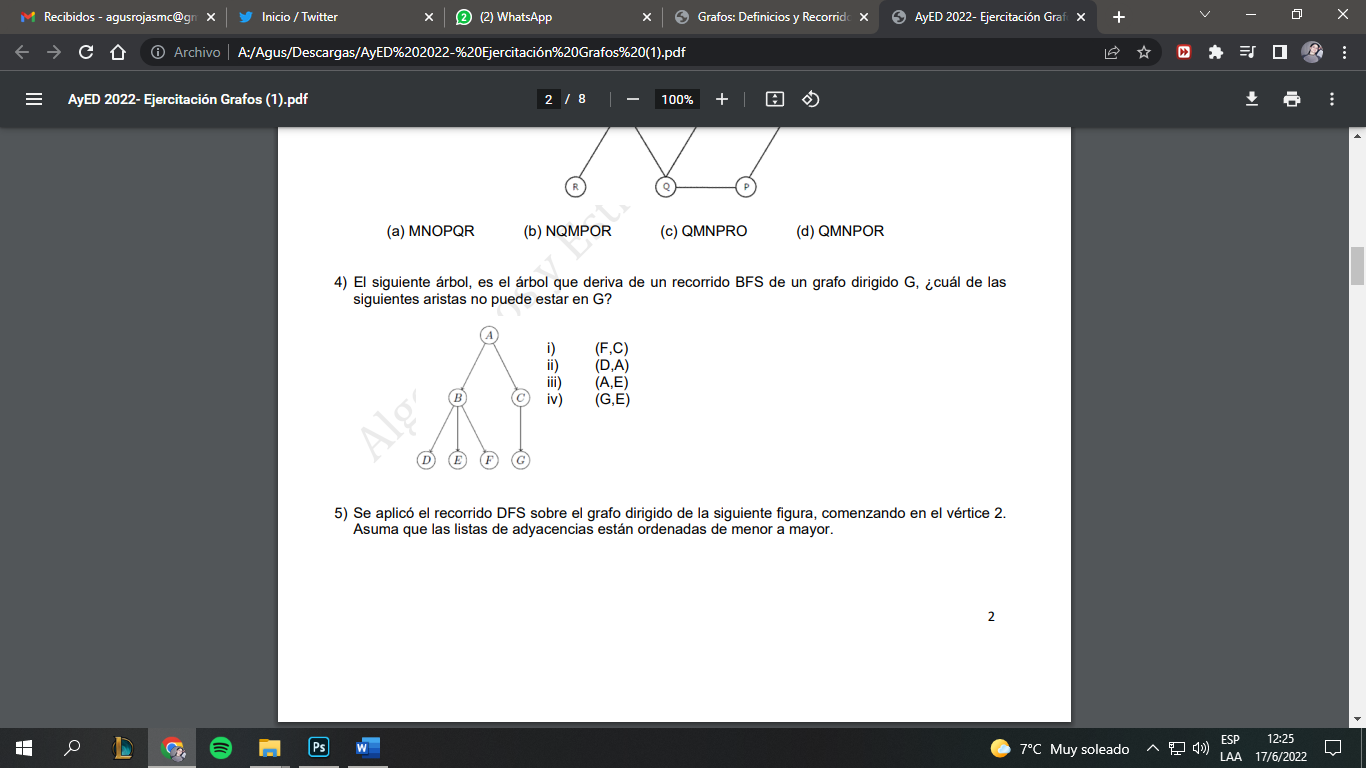


1. 1, 2, 5, 4.
2. 1, 2, 4, 5
3. ¿Cuál de los siguientes es un recorrido BFS válido para el grafo de la figura?



Respuesta (c)

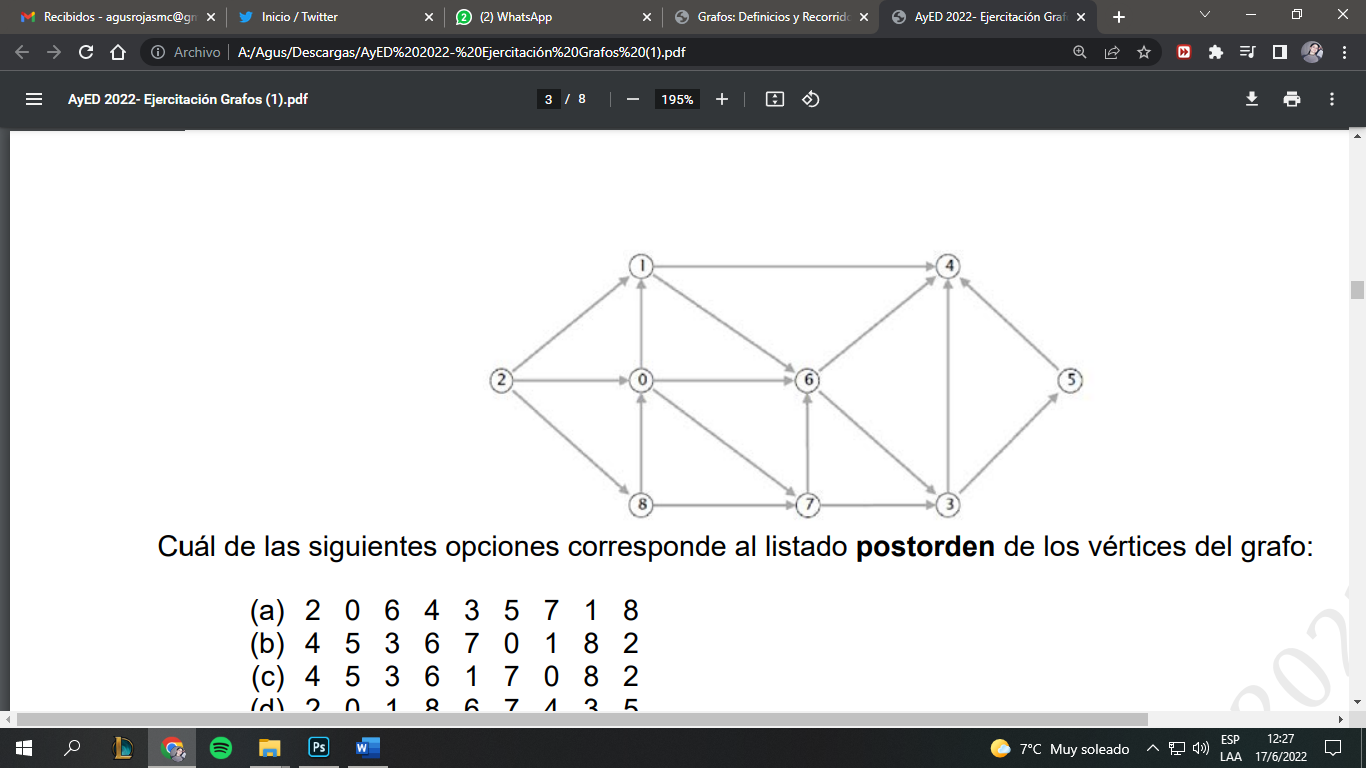
1. El siguiente árbol, es el árbol que deriva de un recorrido BFS de un grafo dirigido G, ¿cuál de las siguientes aristas no puede estar en G?



Respuesta iii) (A,E).

Si estuviera E estaría en el mismo nivel que B y C.

1. Se aplicó el recorrido DFS sobre el grafo dirigido de la siguiente figura, comenzando en el vértice 2. Asuma que las listas de adyacencias están ordenadas de menor a mayor.



Cuál de las siguientes opciones corresponde al listado postorden de los vértices del grafo:

(a) 2 0 6 4 3 5 7 1 8

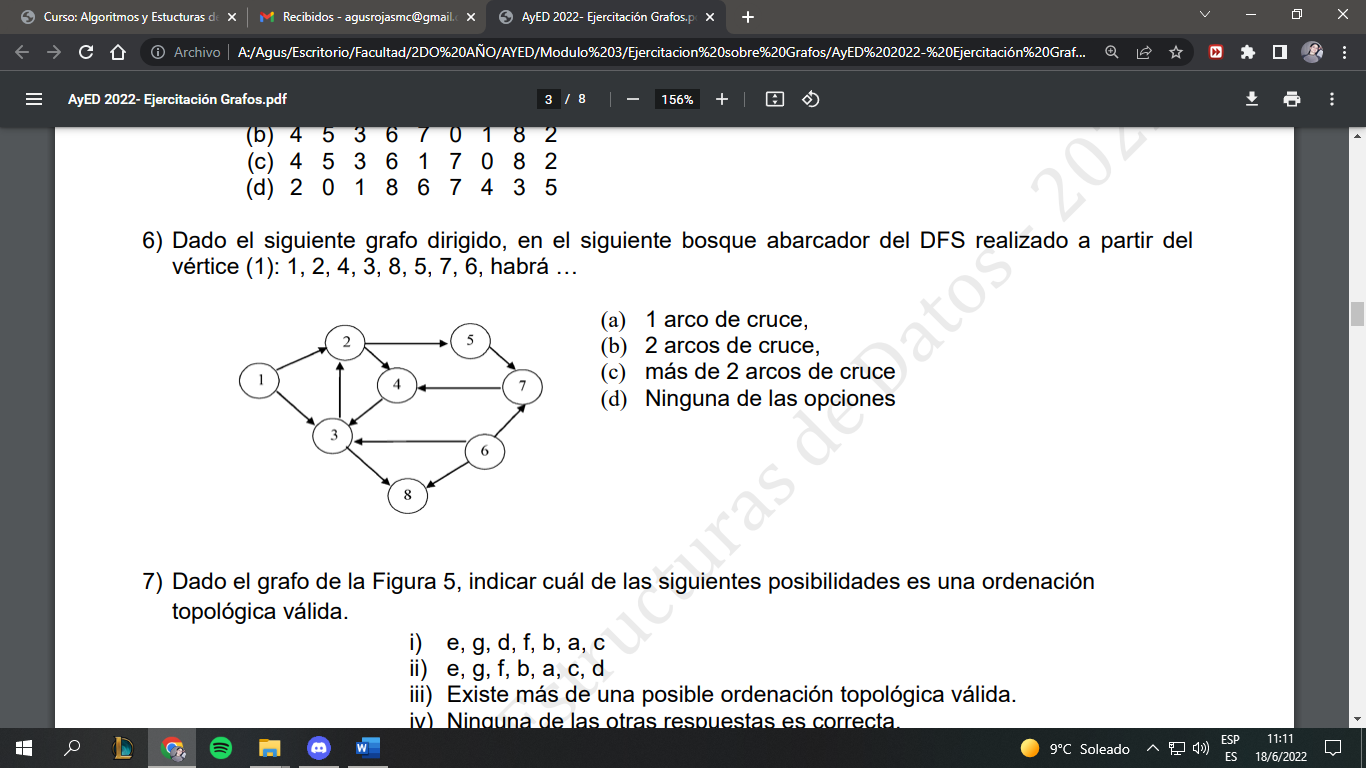
(b) 4 5 3 6 7 0 1 8 2

(c) 4 5 3 6 1 7 0 8 2

(d) 2 0 1 8 6 7 4 3 5

Respuesta (c)

1. Dado el siguiente grafo dirigido, en el siguiente bosque abarcador del DFS realizado a partir del vértice (1): 1, 2, 4, 3, 8, 5, 7, 6, habrá …

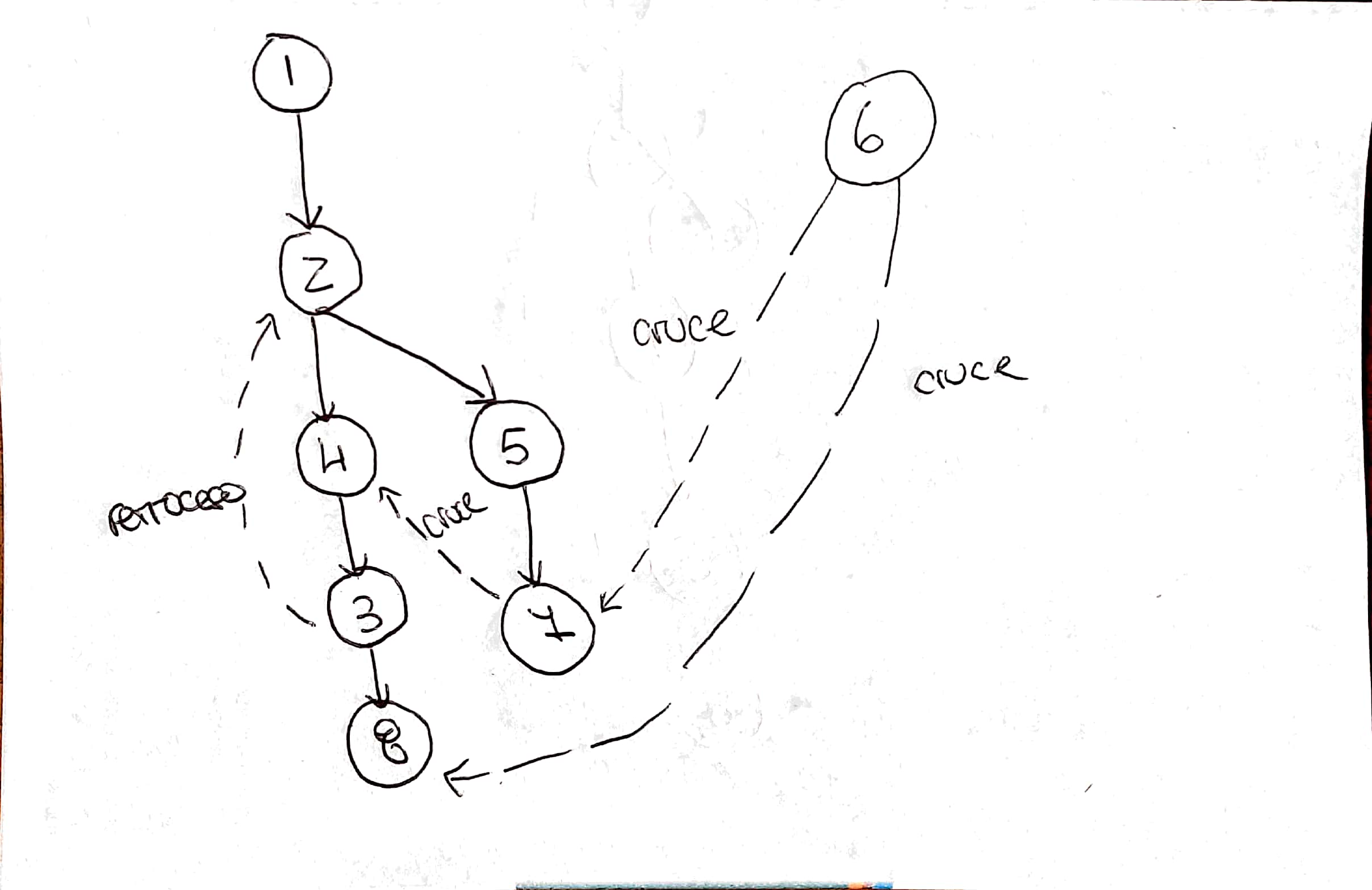


(a) 1 arco de cruce,

(b) 2 arcos de cruce,

(c) más de 2 arcos de cruce

(d) Ninguna de las opciones

Respuesta (c)

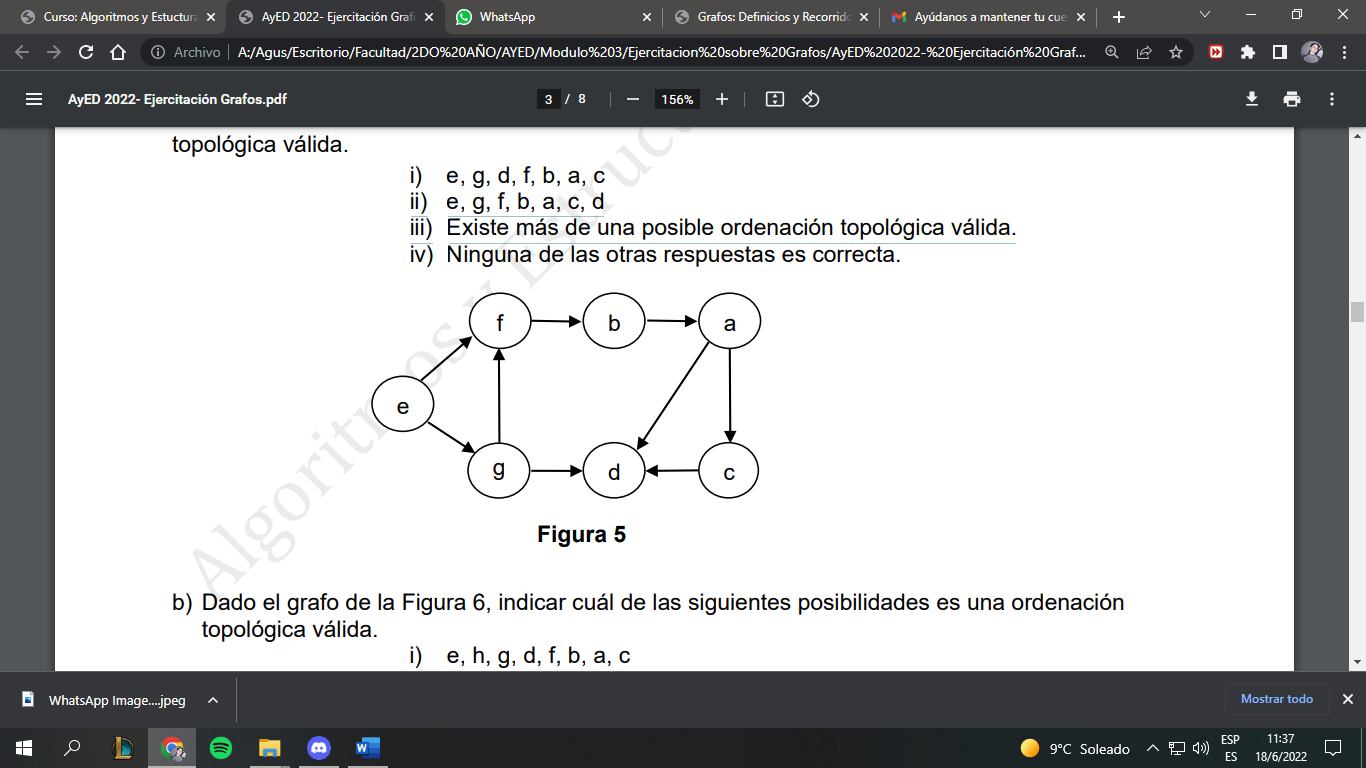
7) Dado el grafo de la Figura 5, indicar cuál de las siguientes posibilidades es una ordenación topológica válida.

i) e, g, d, f, b, a, c

ii) e, g, f, b, a, c, d

iii) Existe más de una posible ordenación topológica válida.

iv) Ninguna de las otras respuestas es correcta.



Respuesta ii)

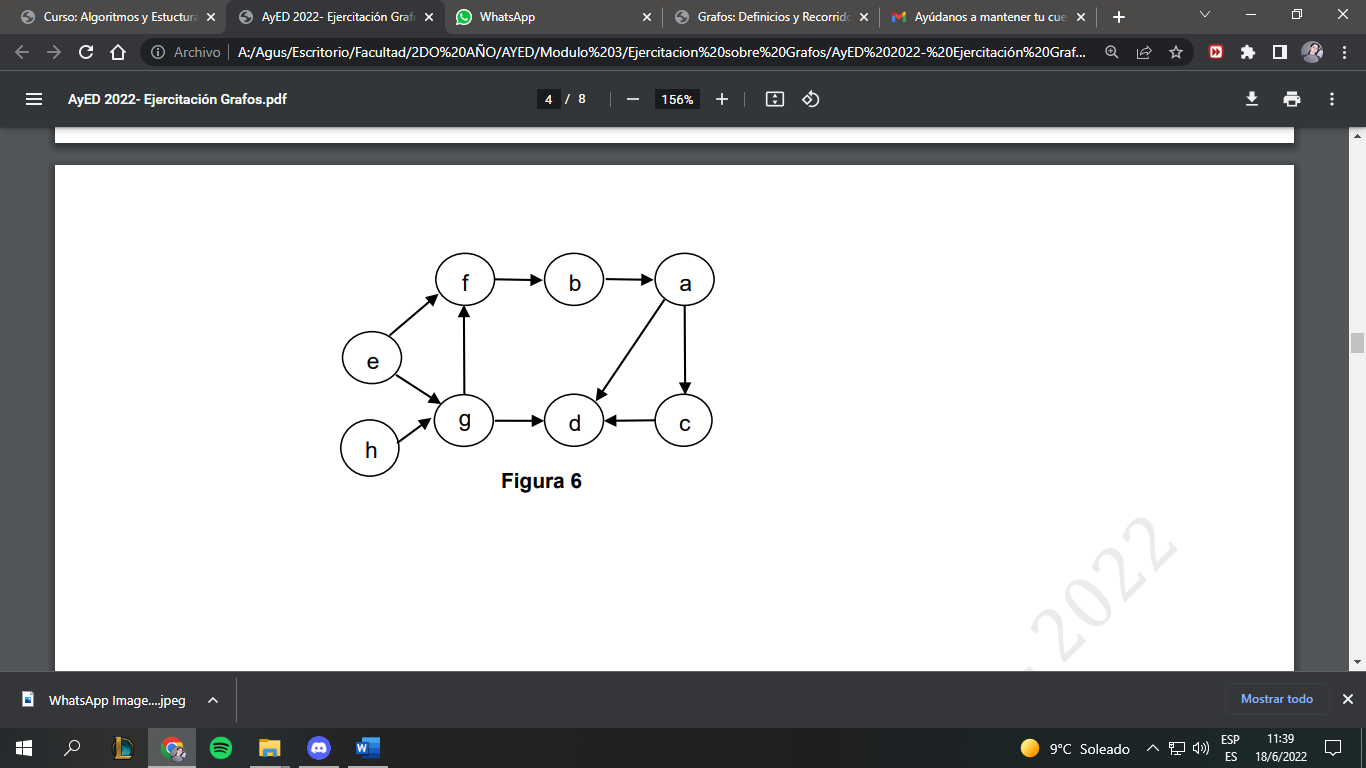
b) Dado el grafo de la Figura 6, indicar cuál de las siguientes posibilidades es una ordenación topológica válida.

i) e, h, g, d, f, b, a, c

ii) e, g, f, b, a, c, d, h

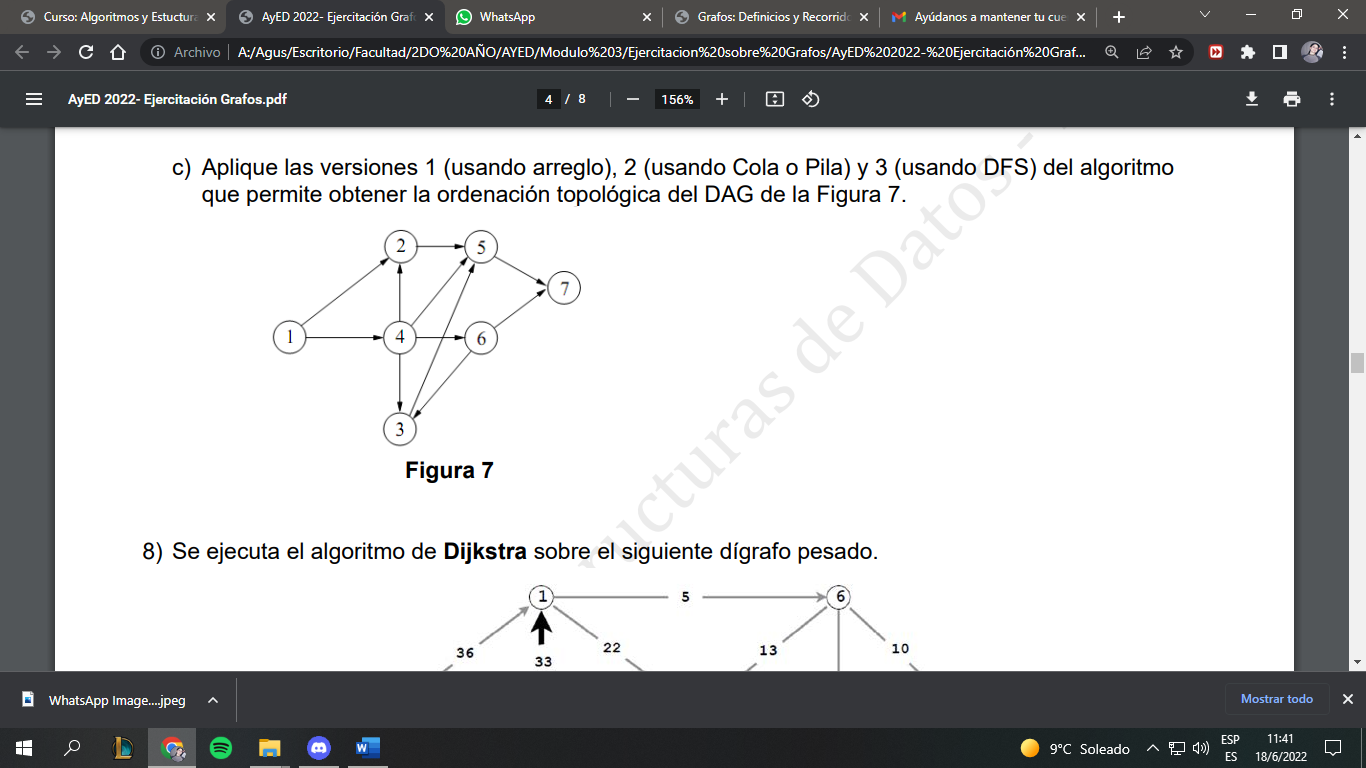
iii) Existe más de una posible ordenación topológica válida.

iv) Ninguna de las otras respuestas es correcta.



Respuesta iii)

1. Aplique las versiones 1 (usando arreglo), 2 (usando Cola o Pila) y 3 (usando DFS) del algoritmo que permite obtener la ordenación topológica del DAG de la Figura 7.



Versión 1

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 0 | 2 | 2 | 1 | 3 | 1 | 2 |
| 0 | 1 | 2 | 0 | 3 | 1 | 2 |
| 0 | 0 | 1 | 0 | 2 | 0 | 2 |
| 0 | 0 | 1 | 0 | 1 | 0 | 2 |
| 0 | 0 | 0 | 0 | 1 | 0 | 1 |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

SORT TOPOLOGICO: 1, 4, 2, 6, 3, 5, 7

Versión 2

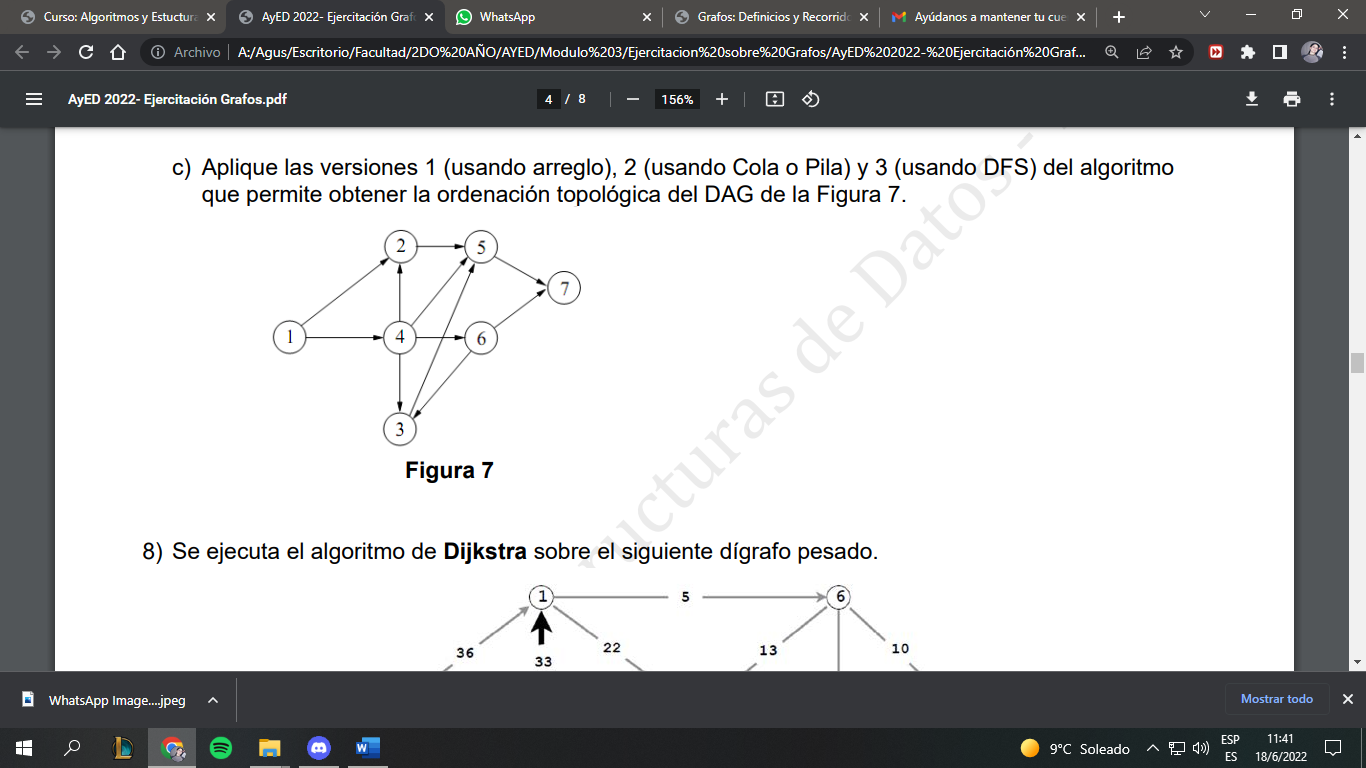
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 0 | 0 | 2 | 2 | 1 | 3 | 1 | 2 |
| 1 | 0 | 1 | 2 | 0 | 3 | 1 | 2 |
| 2 | 0 | 0 | 1 | 0 | 2 | 0 | 2 |
| 3 | 0 | 0 | 0 | 0 | 2 | 0 | 1 |
| 4 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |
| 5 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 6 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

PILA:

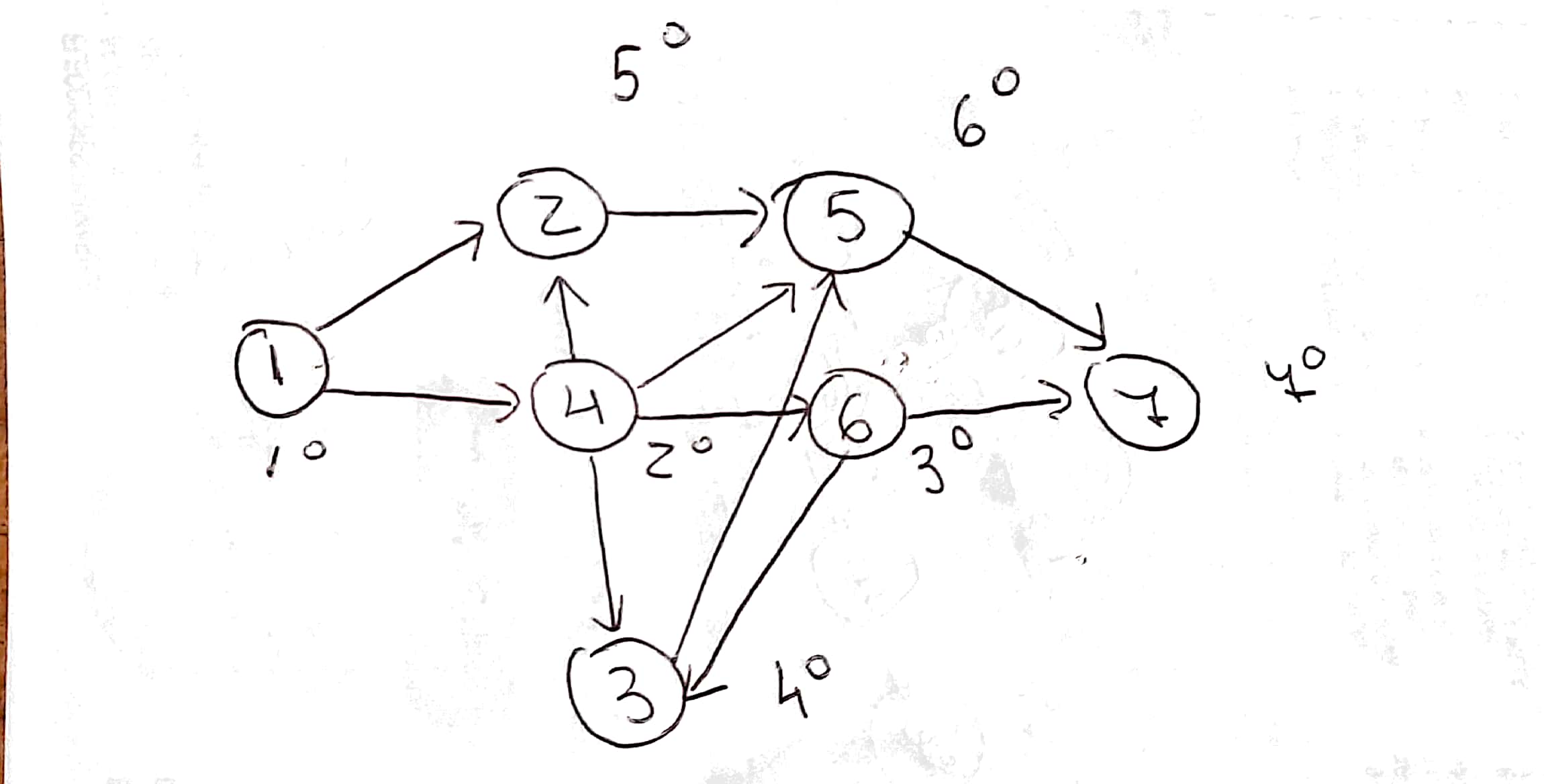
1. 1 // - 1
2. 4 // + 4 - 4
3. 2 - 6 // +2 + 6 -6
4. 2 - 3 // + 3 – 3
5. 2 // -2
6. 5 // +5 -5
7. 7 // -7

SORT TOPOLOGICO: 1, 4, 6, 3, 2, 5, 7

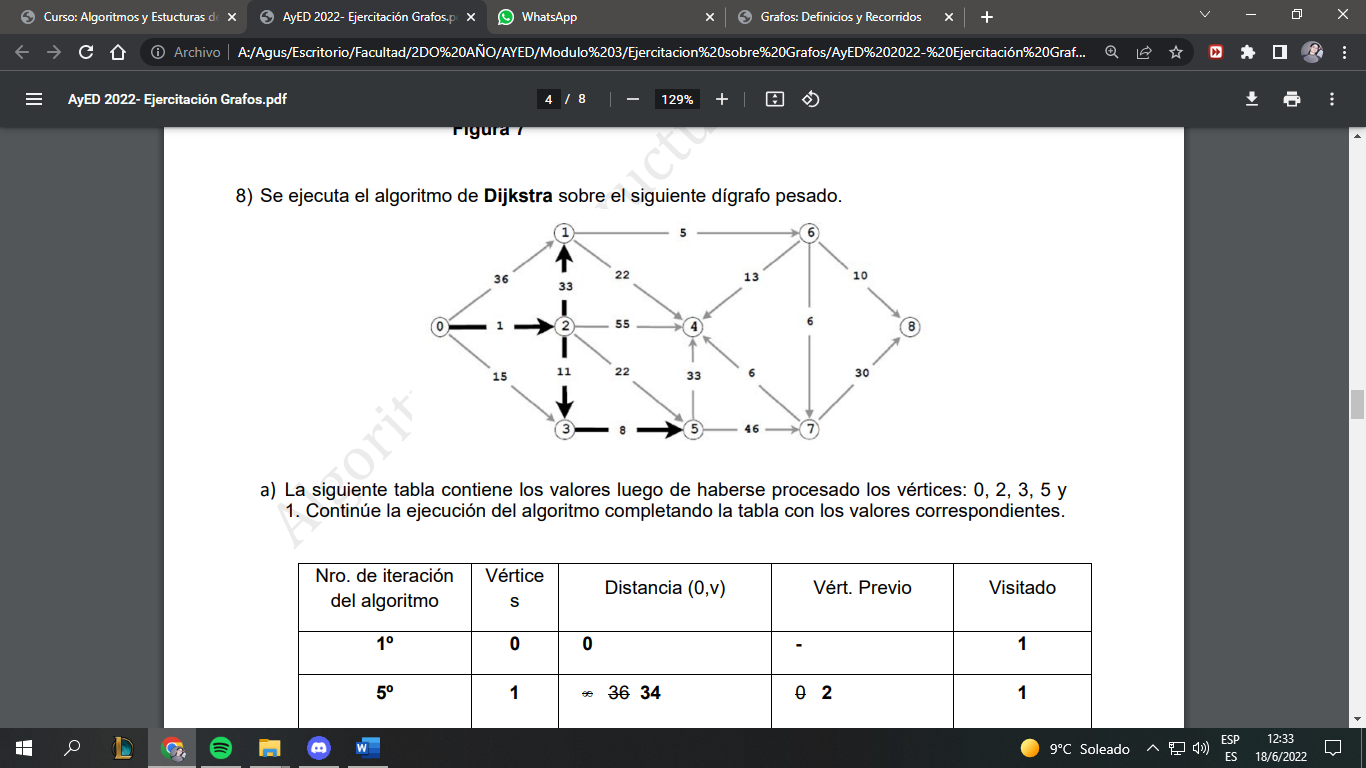
Versión 3 Opción a



1, 4, 6, 3, 2, 5, 7



8) Se ejecuta el algoritmo de Dijkstra sobre el siguiente dígrafo pesado.



a) La siguiente tabla contiene los valores luego de haberse procesado los vértices: 0, 2, 3, 5 y 1. Continúe la ejecución del algoritmo completando la tabla con los valores correspondientes.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nro. de iteración del algoritmo | Vértices | Distancia (0,v) | Vért. Previo | Visitado |
| 1º | 0 | 0 | - | 1 |
| 5º | 1 | 34 | 2 | 1 |
| 2º | 2 | 1 | 0 | 1 |
| 3º | 3 | 12 | 2 | 1 |
|  | 4 | 53 | 5 | 0 |
| 4º | 5 | 20 | 3 | 1 |
|  | 6 | 39 | 1 | 0 |
|  | 7 | 66 | 5 | 0 |
|  | 8 | ∞ | - | 0 |

* Selecciono Vértice 6 y actualizo el Vértice 8

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nro. de iteración del algoritmo | Vértices | Distancia (0,v) | Vért. Previo | Visitado |
| 1º | 0 | 0 | - | 1 |
| 5º | 1 | 34 | 2 | 1 |
| 2º | 2 | 1 | 0 | 1 |
| 3º | 3 | 12 | 2 | 1 |
|  | 4 | 53 | 5 | 0 |
| 4º | 5 | 20 | 3 | 1 |
| 6º | 6 | 39 | 1 | 1 |
|  | 7 | 66 | 5 | 0 |
|  | 8 | 49 | 6 | 0 |

* Selecciono Vértice 8 y no actualizo nada.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nro. de iteración del algoritmo | Vértices | Distancia (0,v) | Vért. Previo | Visitado |
| 1º | 0 | 0 | - | 1 |
| 5º | 1 | 34 | 2 | 1 |
| 2º | 2 | 1 | 0 | 1 |
| 3º | 3 | 12 | 2 | 1 |
|  | 4 | 53 | 5 | 0 |
| 4º | 5 | 20 | 3 | 1 |
| 6º | 6 | 39 | 1 | 1 |
|  | 7 | 66 | 5 | 0 |
| 7º | 8 | 49 | 6 | 1 |

* Selecciono Vértice 4 y no actualizo nada.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nro. de iteración del algoritmo | Vértices | Distancia (0,v) | Vért. Previo | Visitado |
| 1º | 0 | 0 | - | 1 |
| 5º | 1 | 34 | 2 | 1 |
| 2º | 2 | 1 | 0 | 1 |
| 3º | 3 | 12 | 2 | 1 |
| 8º | 4 | 53 | 5 | 1 |
| 4º | 5 | 20 | 3 | 1 |
| 6º | 6 | 39 | 1 | 1 |
|  | 7 | 66 | 5 | 0 |
| 7º | 8 | 49 | 6 | 1 |

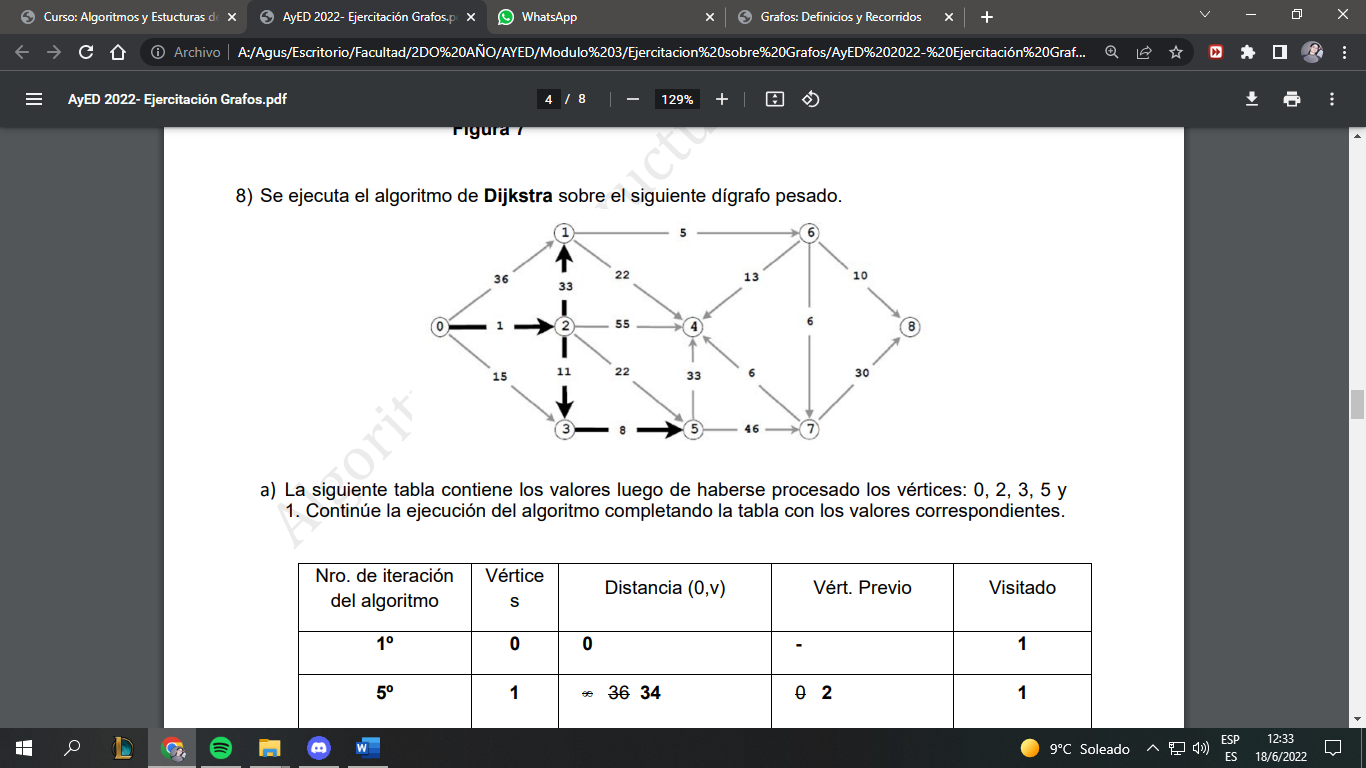
* Selecciono Vértice 7 y no actualizo nada.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nro. de iteración del algoritmo | Vértices | Distancia (0,v) | Vért. Previo | Visitado |
| 1º | 0 | 0 | - | 1 |
| 5º | 1 | 34 | 2 | 1 |
| 2º | 2 | 1 | 0 | 1 |
| 3º | 3 | 12 | 2 | 1 |
| 8º | 4 | 53 | 5 | 1 |
| 4º | 5 | 20 | 3 | 1 |
| 6º | 6 | 39 | 1 | 1 |
| 9º | 7 | 66 | 5 | 1 |
| 7º | 8 | 49 | 6 | 1 |

b) Complete la secuencia de vértices según el orden en el que el algoritmo de Dijkstra los toma (es decir, los considera “visitados”). Recuerde que la ejecución del algoritmo comenzó por el vértice “0

0 2 3 5 1 6 8 4 7

1. Dibuje sobre el grafo, los arcos (con trazo más grueso) del árbol abarcador resultante.



1. Recupere los vértices que componen los caminos de costo mínimo obtenidos con el algoritmo de Dijkstra, para los siguientes pares: (0,5) (0,7)

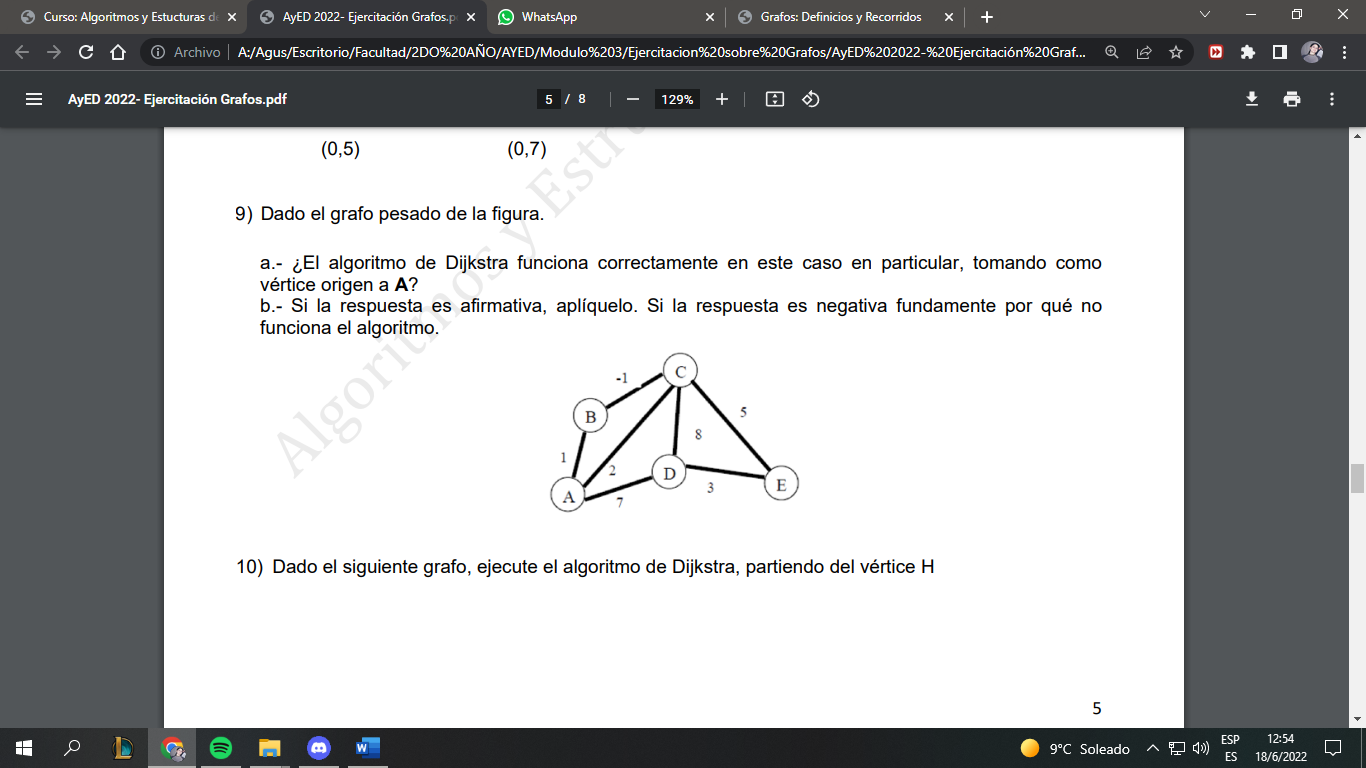
(0,5) 🡪 0 2 3 5

(0,7) 🡪 0 2 3 5 7

9) Dado el grafo pesado de la figura.

a.- ¿El algoritmo de Dijkstra funciona correctamente en este caso en particular, tomando como vértice origen a A?

b.- Si la respuesta es afirmativa, aplíquelo. Si la respuesta es negativa fundamente por qué no funciona el algoritmo.



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nro. de iteración del algoritmo | Vértices | Distancia (0,v) | Vért. Previo | Visitado |
|  | A | 0 | - | 0 |
|  | B | ∞ | 0 | 0 |
|  | C | ∞ | 0 | 0 |
|  | D | ∞ | 0 | 0 |
|  | E | ∞ | 0 | 0 |

* Selecciono vértice A y actualizo B, C Y D

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nro. de iteración del algoritmo | Vértices | Distancia (0,v) | Vért. Previo | Visitado |
|  | A | 0 | - | 1 |
|  | B | 1 | A | 0 |
|  | C | 2 | A | 0 |
|  | D | 7 | A | 0 |
|  | E | ∞ | 0 | 0 |

* Selecciono vértice C y lo marco como visitado

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nro. de iteración del algoritmo | Vértices | Distancia (0,v) | Vért. Previo | Visitado |
|  | A | 0 | - | 1 |
|  | B | 1 | A | 0 |
|  | C | 2 | A | 1 |
|  | D | 7 | A | 0 |
|  | E | ∞ | 0 | 0 |

No funciona el algoritmo, ya que una vez marcado C como visitado dice que la distancia mínima es 2, cuando en realidad es 0. Esto sucede porque se marca el C como visitado cuando puede ser que se siga actualizando ya que hay valores negativos.

10) Dado el siguiente grafo, ejecute el algoritmo de Dijkstra, partiendo del vértice H



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nro. de iteración del algoritmo | Vértices | Distancia (0,v) | Vért. Previo | Visitado |
|  | A | ∞ | 0 | 0 |
|  | B | ∞ | 0 | 0 |
|  | C | ∞ | 0 | 0 |
|  | D | ∞ | 0 | 0 |
|  | E | ∞ | 0 | 0 |
|  | F | ∞ | 0 | 0 |
|  | G | ∞ | 0 | 0 |
|  | H | 0 | - | 0 |

Selecciono Vértice H y actualizo a A F B D G

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nro. de iteración del algoritmo | Vértices | Distancia (0,v) | Vért. Previo | Visitado |
|  | A | 10 | H | 0 |
|  | B | 6 | H | 0 |
|  | C | ∞ | 0 | 0 |
|  | D | 14 | H | 0 |
|  | E | ∞ | 0 | 0 |
|  | F | 9 | H | 0 |
|  | G | 3 | H | 0 |
| 1º | H | 0 | - | 1 |

Selecciono Vértice G y actualizo a C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nro. de iteración del algoritmo | Vértices | Distancia (0,v) | Vért. Previo | Visitado |
|  | A | 10 | H | 0 |
|  | B | 6 | H | 0 |
|  | C | 12 | G | 0 |
|  | D | 14 | H | 0 |
|  | E | 18 | G | 0 |
|  | F | 9 | H | 0 |
| 2º | G | 3 | H | 1 |
| 1º | H | 0 | - | 1 |

Selecciono Vértice B y actualizo a A C E

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nro. de iteración del algoritmo | Vértices | Distancia (0,v) | Vért. Previo | Visitado |
|  | A | 9 | B | 0 |
| 3º | B | 6 | H | 1 |
|  | C | 11 | B | 0 |
|  | D | 14 | H | 0 |
|  | E | 10 | B | 0 |
|  | F | 9 | H | 0 |
| 2º | G | 3 | H | 1 |
| 1º | H | 0 | - | 1 |

Selecciono Vértice A y actualizo a D

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nro. de iteración del algoritmo | Vértices | Distancia (0,v) | Vért. Previo | Visitado |
| 4º | A | 9 | B | 1 |
| 3º | B | 6 | H | 1 |
|  | C | 11 | B | 0 |
|  | D | 11 | A | 0 |
|  | E | 10 | B | 0 |
|  | F | 9 | H | 0 |
| 2º | G | 3 | H | 1 |
| 1º | H | 0 | - | 1 |

Selecciono Vértice F y no actualizo a nadie

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nro. de iteración del algoritmo | Vértices | Distancia (0,v) | Vért. Previo | Visitado |
| 4º | A | 9 | B | 1 |
| 3º | B | 6 | H | 1 |
|  | C | 11 | B | 0 |
|  | D | 11 | A | 0 |
|  | E | 10 | B | 0 |
| 5º | F | 9 | H | 1 |
| 2º | G | 3 | H | 1 |
| 1º | H | 0 | - | 1 |

Selecciono Vértice E y no actualizo a nadie

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nro. de iteración del algoritmo | Vértices | Distancia (0,v) | Vért. Previo | Visitado |
| 4º | A | 9 | B | 1 |
| 3º | B | 6 | H | 1 |
|  | C | 11 | B | 0 |
|  | D | 11 | A | 0 |
| 6º | E | 10 | B | 1 |
| 5º | F | 9 | H | 1 |
| 2º | G | 3 | H | 1 |
| 1º | H | 0 | - | 1 |

Selecciono Vértice C y no actualizo a nadie

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nro. de iteración del algoritmo | Vértices | Distancia (0,v) | Vért. Previo | Visitado |
| 4º | A | 9 | B | 1 |
| 3º | B | 6 | H | 1 |
| 7º | C | 11 | B | 1 |
|  | D | 11 | A | 0 |
| 6º | E | 10 | B | 1 |
| 5º | F | 9 | H | 1 |
| 2º | G | 3 | H | 1 |
| 1º | H | 0 | - | 1 |

Selecciono Vértice D y no actualizo a nadie

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nro. de iteración del algoritmo | Vértices | Distancia (0,v) | Vért. Previo | Visitado |
| 4º | A | 9 | B | 1 |
| 3º | B | 6 | H | 1 |
| 7º | C | 11 | B | 1 |
| 8º | D | 11 | A | 1 |
| 6º | E | 10 | B | 1 |
| 5º | F | 9 | H | 1 |
| 2º | G | 3 | H | 1 |
| 1º | H | 0 | - | 1 |

a) ¿Cuáles fueron los costos intermedios encontrados por el algoritmo para encontrar el camino mínimo a E?

- 14, 11

- 18, 10 🡪 Respuesta Correcta

- 18, 11, 10

- 12, 11, 10

- Ninguna de las anteriores

b) ¿Cuáles fueron los vértices intermedios encontrados por el algoritmo para encontrar el camino mínimo a E?

- A, C, B

- A, B, G

- G, B 🡪 Respuesta Correcta

- D, B

- A, C - B

c) ¿En qué Iteración del algoritmo fue tomado el vértice C?

- 4°

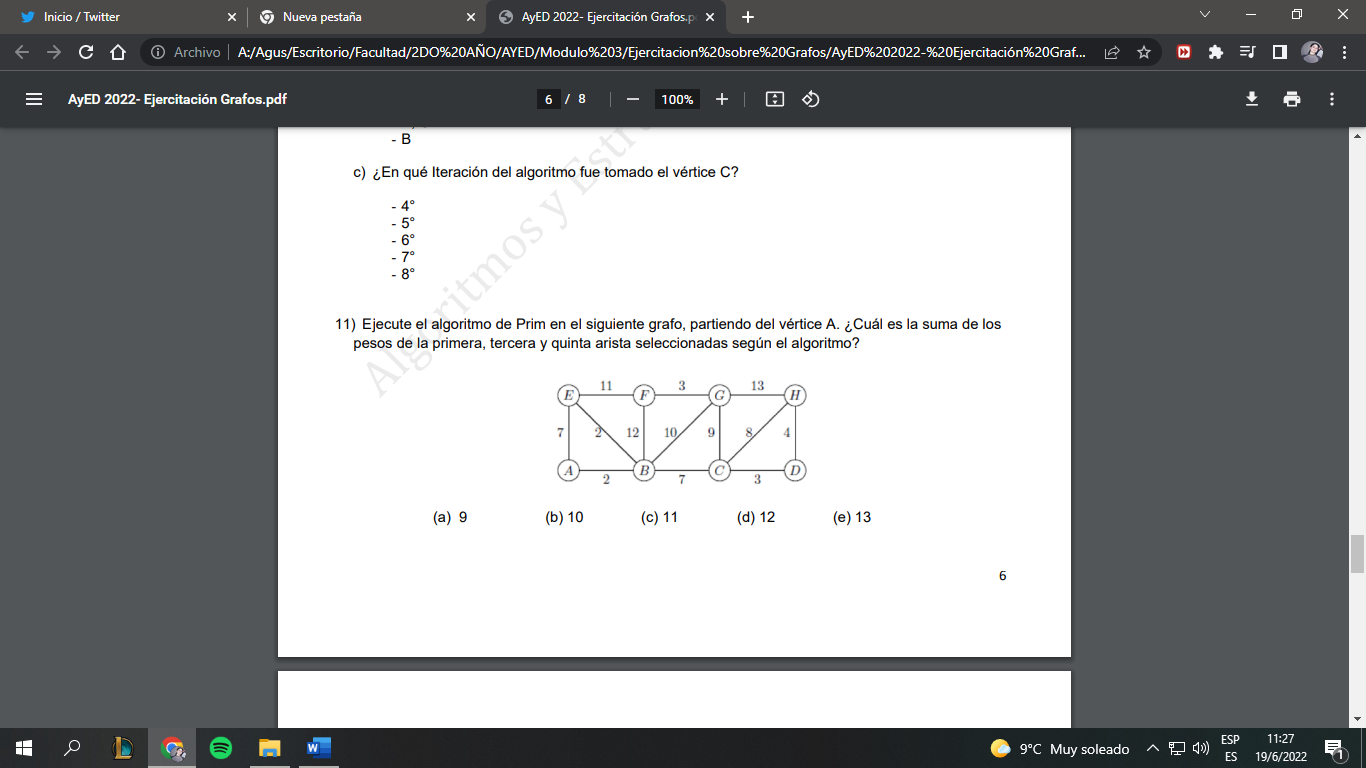
- 5°

- 6°

- 7° 🡪 Respuesta Correcta

- 8°

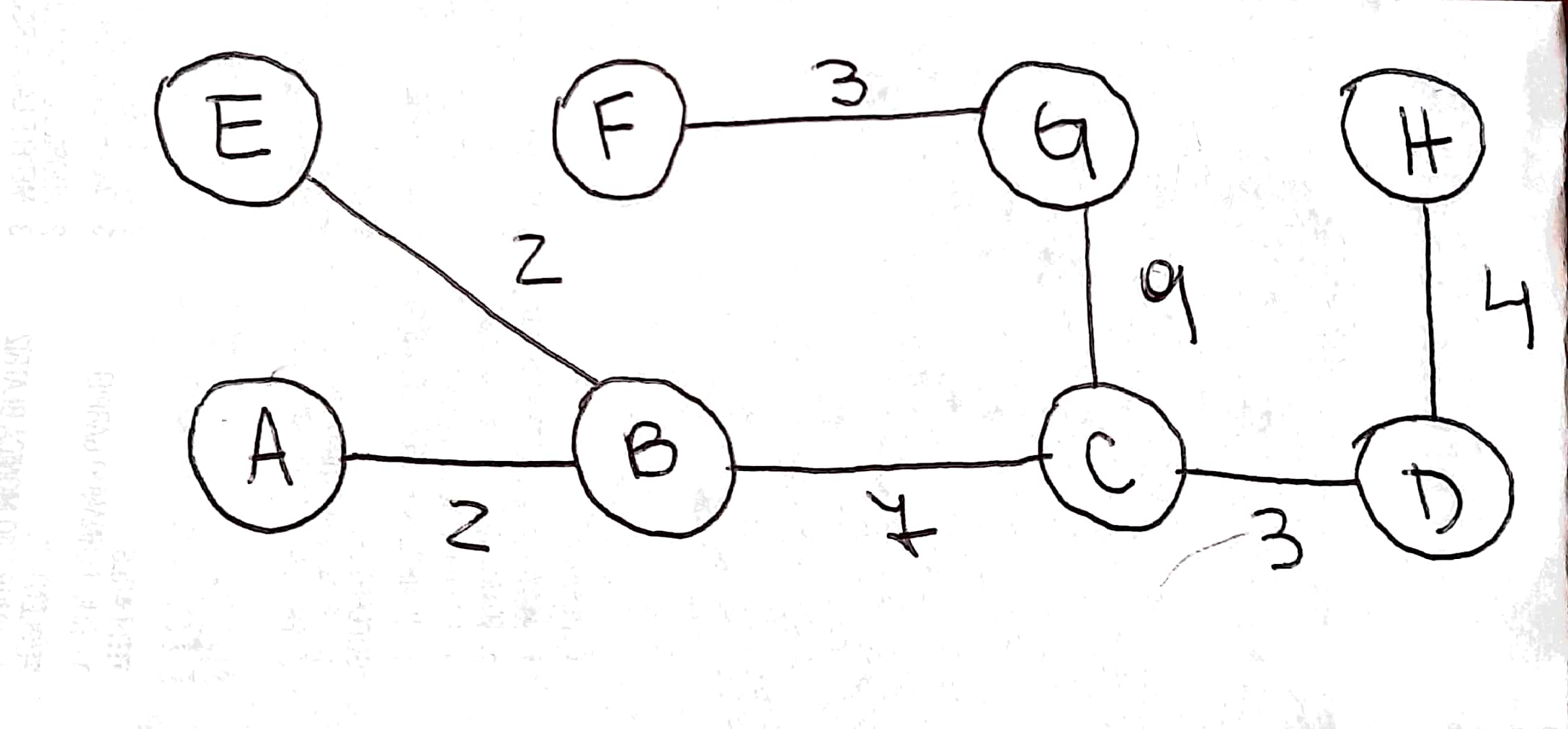
11) Ejecute el algoritmo de Prim en el siguiente grafo, partiendo del vértice A. ¿Cuál es la suma de los pesos de la primera, tercera y quinta arista seleccionadas según el algoritmo?



1. 9 (b) 10 (c) 11 (d) 12 (e) 13

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nro. de selección | Vértices | Costo (v,w) | Vért. Previo | Visitado |
| 0º | A | 0 | 0 | 1 |
| 1º | B | ~~∞~~ 2 | A | 1 |
| 3 º | C | ~~∞~~ 7 | B | 1 |
| 4º | D | ~~∞~~ 3 | C | 1 |
| 2º | E | ~~∞~~ ~~7~~ 2 | ~~A~~ B | 1 |
| 7º | F | ~~∞~~ ~~12~~ ~~11~~ 3 | ~~B~~ ~~E~~ G | 1 |
| 6º | G | ~~∞~~ 9 | C | 1 |
| 5º | H | ~~∞~~ ~~8~~ 4 | D | 1 |

Respuesta (e) 13

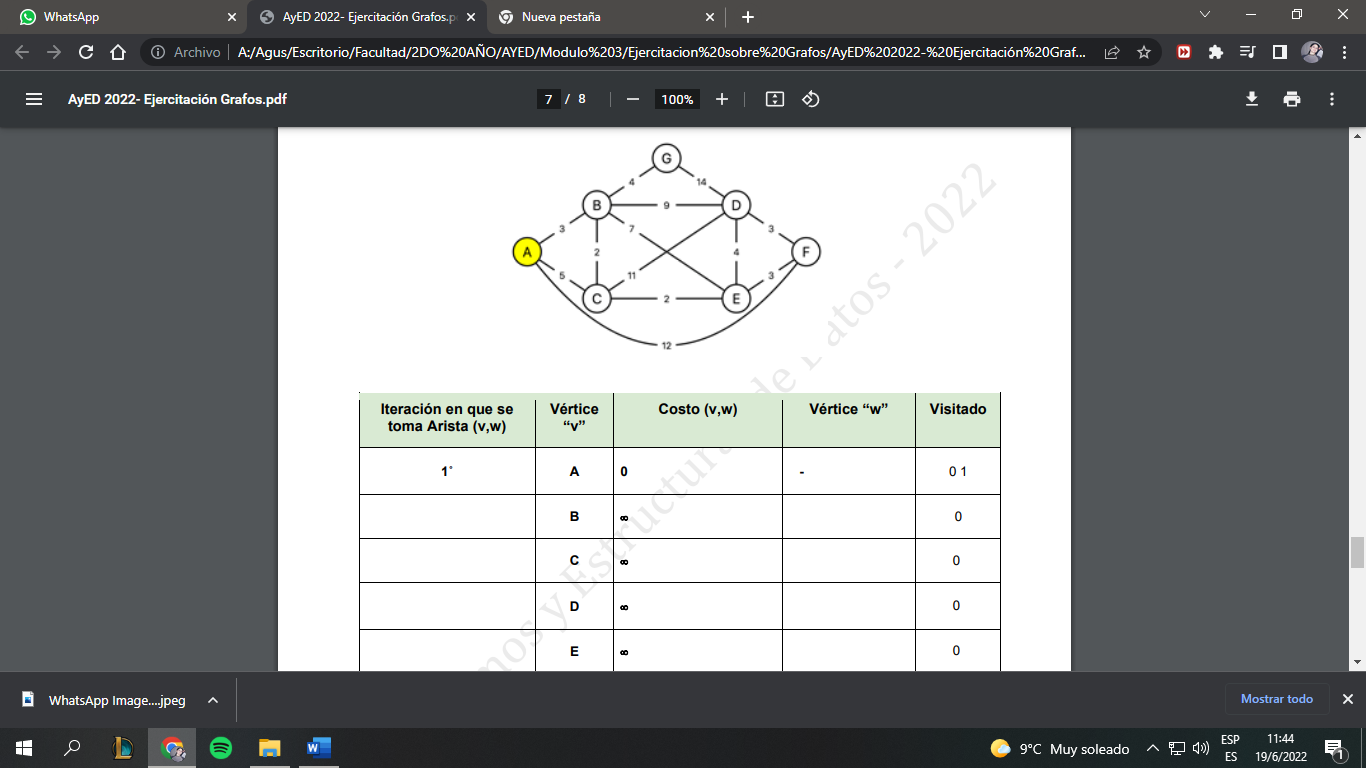


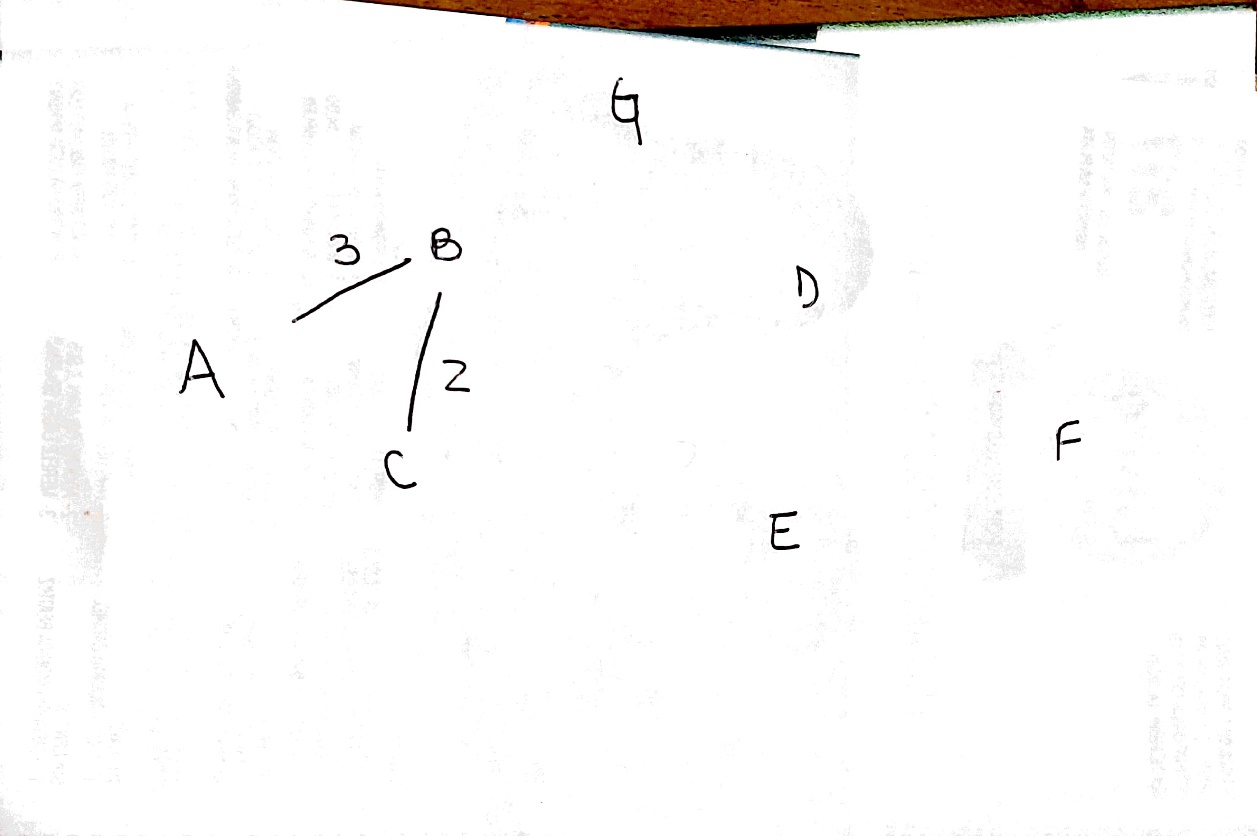
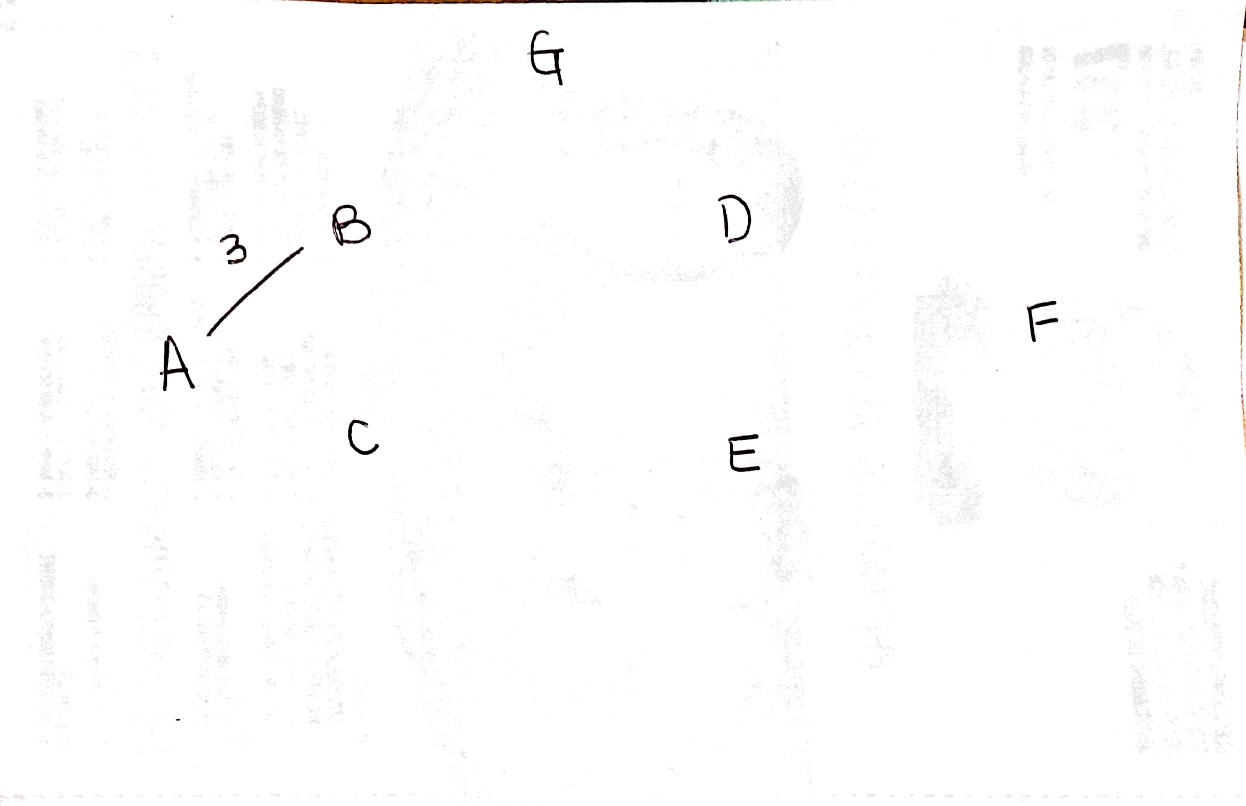
12) Obtener el árbol de expansión mínima utilizando el algoritmo de PRIM en el siguiente grafo comenzando del vértice A.

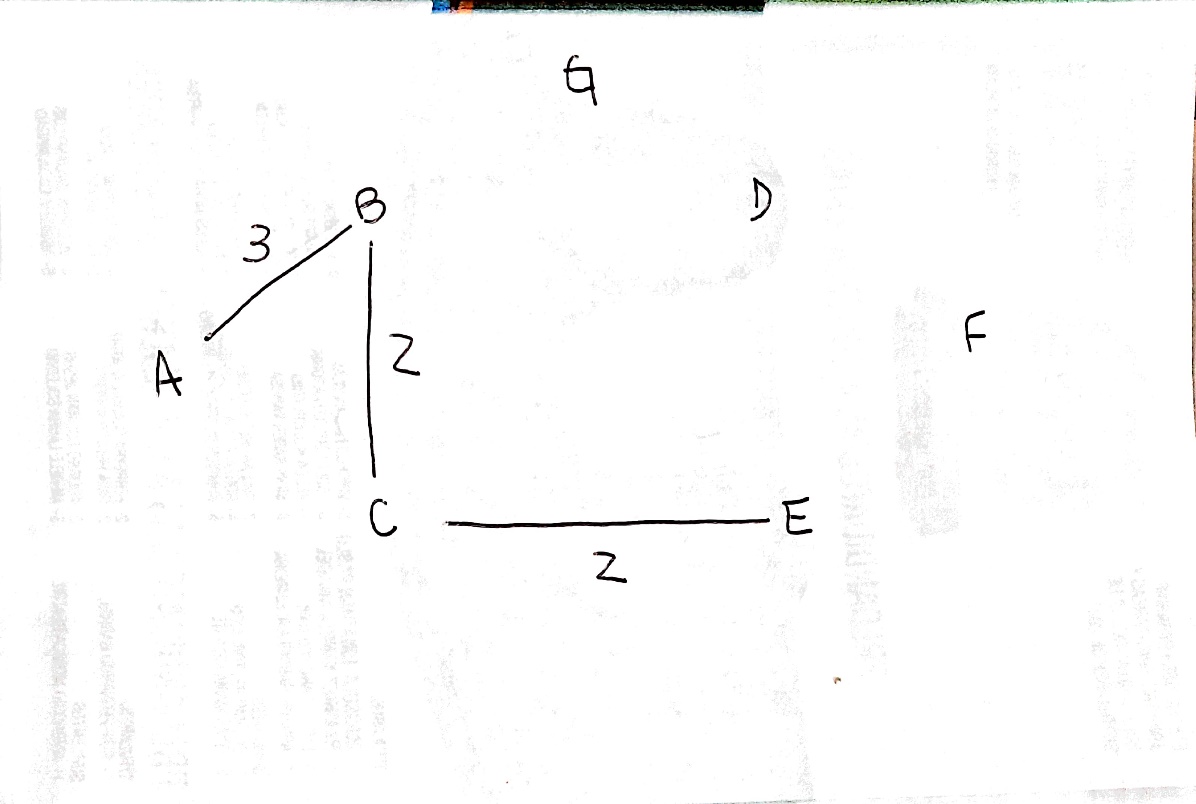
a.- Dibuje cómo evoluciona la construcción del árbol en cada paso.

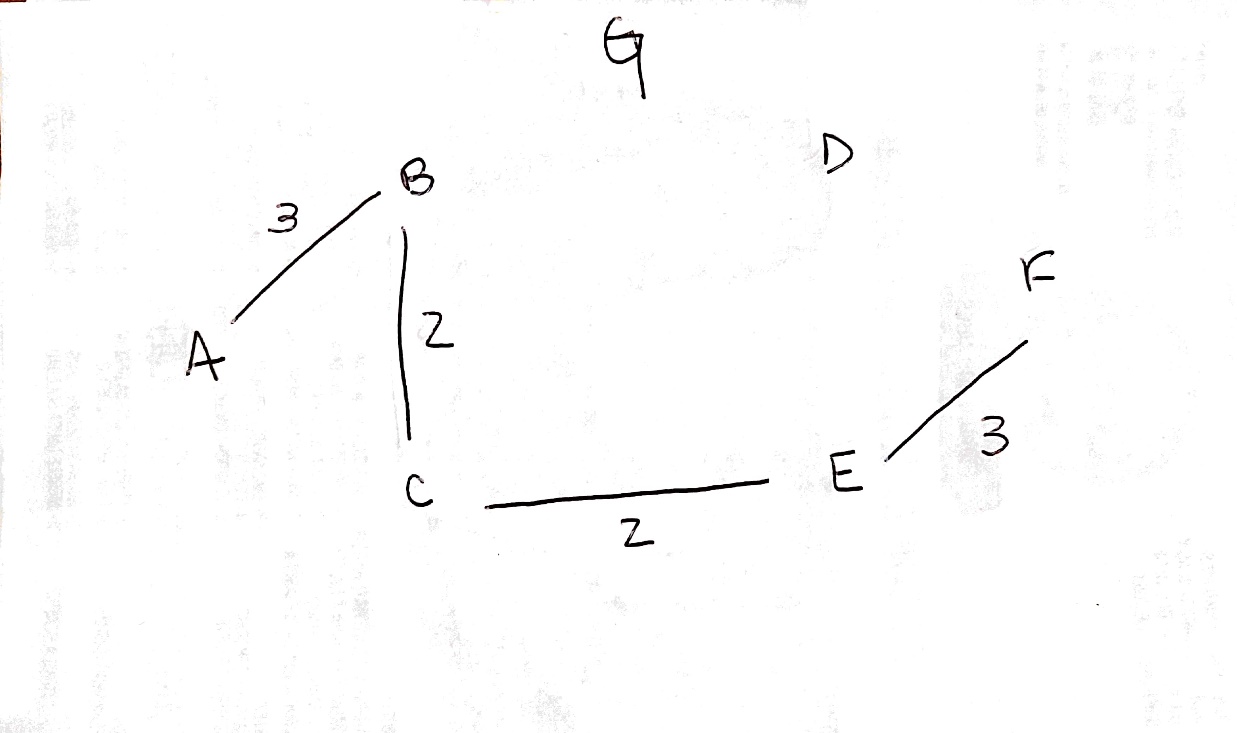
b.- Muestre la ejecución del algoritmo en la tabla que aparece más abajo.

c.- Exprese el orden de ejecución del algoritmo (en cuanto a su eficiencia). Justifique su respuesta.

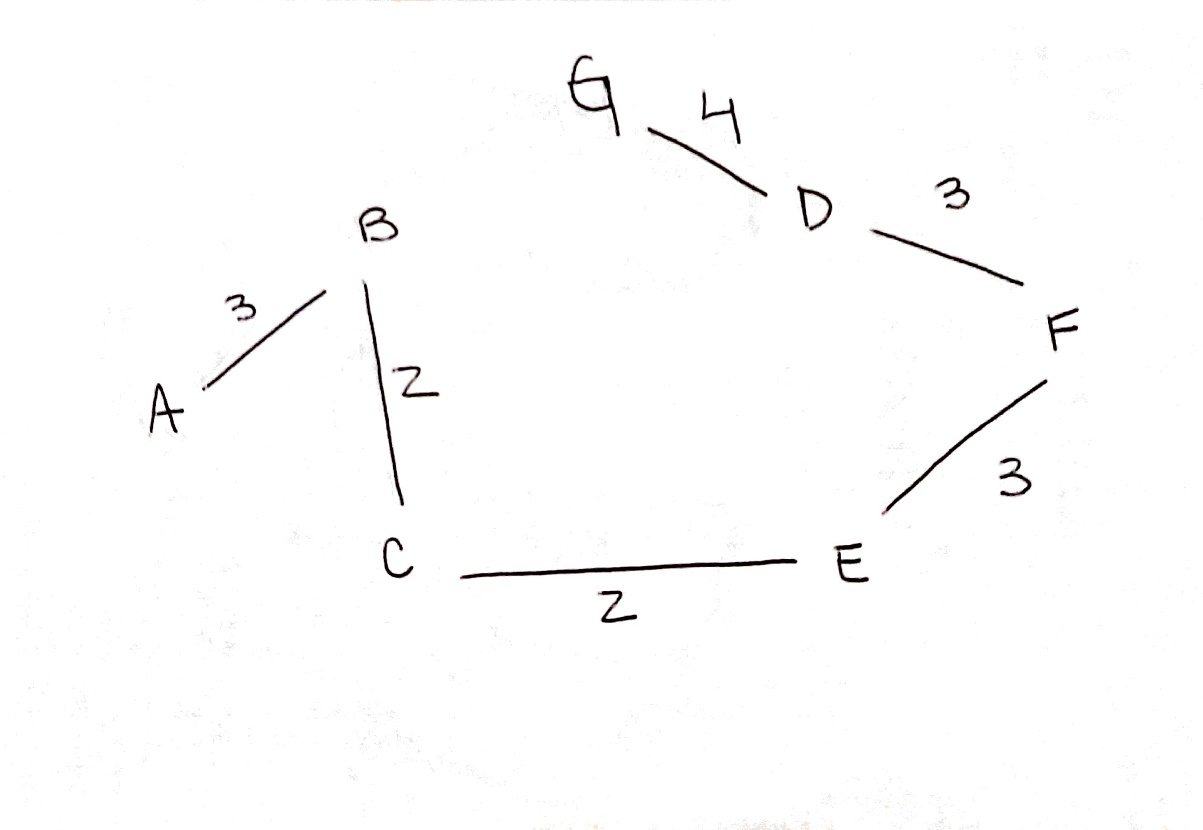










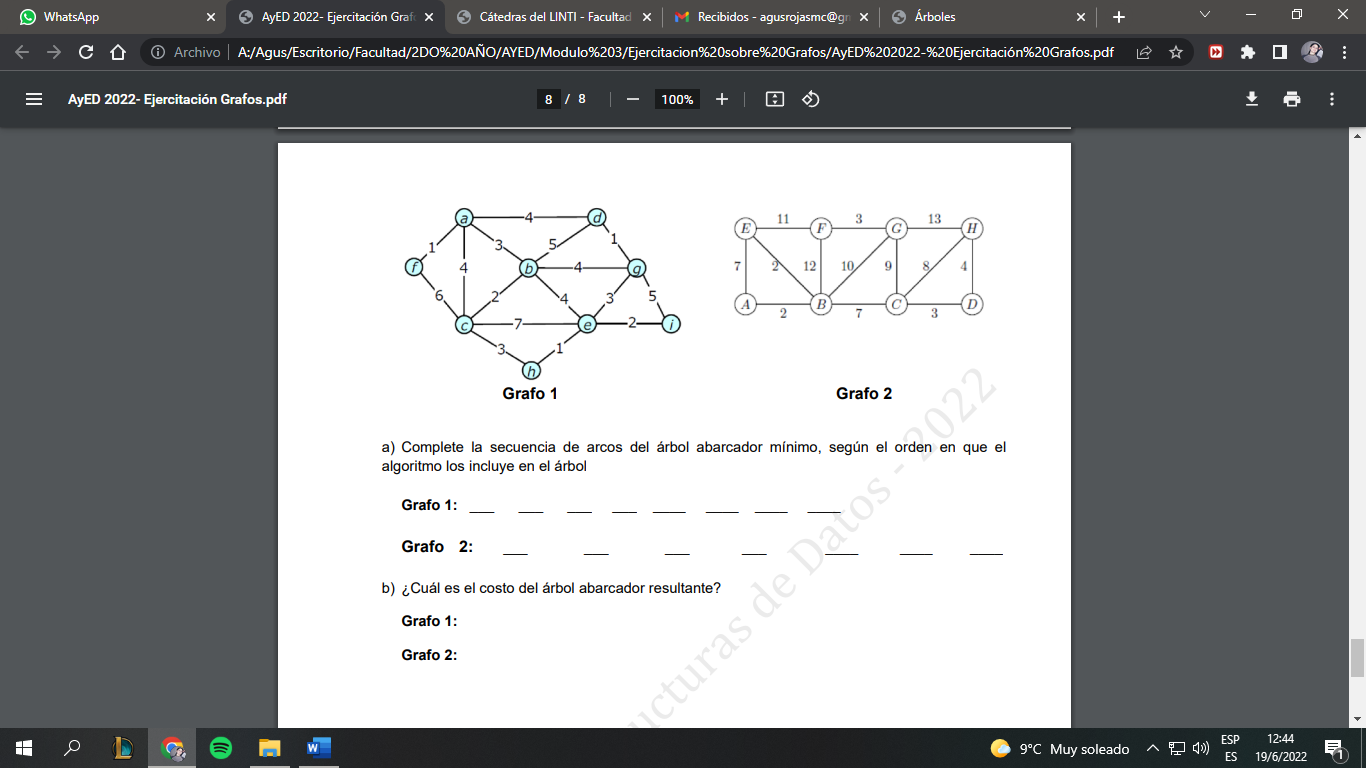


|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Iteración en que se toma Arista (v,w) | Vértice “v” | Costo (v,w) | Vért. Previo | Visitado |
| 1º | A | 0 | - | 1 |
| 2º | B | ~~∞~~ 3 | A | 1 |
| 3º | C | ~~∞~~ ~~5~~ 2 | ~~A~~ B | 1 |
| 6º | D | ~~∞~~ ~~11~~ ~~4~~ 3 | ~~B~~ ~~E~~ F | 1 |
| 4º | E | ~~∞~~ ~~7~~ 2 | ~~B~~ C | 1 |
| 5º | F | ~~∞~~ ~~12~~ 3 | ~~A~~ E | 1 |
| 7º | G | ~~∞~~ 4 | B | 1 |

1. Se hacen las mismas consideraciones que para el algoritmo de Dijkstra :

* Si se implementa con una tabla secuencial: El costo total del algoritmo es O(|V|^2 )
  + Porque hay que recorrer V veces los vértices para sacar el mínimo y las actualizaciones se realizan E veces
* Si se implementa con heap: El costo total del algoritmo es O(|E| log|V|)
  + Porque hay que hacer un deleteMin V veces que sería log|V| y las actualizaciones se realizan E veces

13) Obtener el árbol de expansión mínima utilizando el algoritmo de Kruskal en los siguientes grafos, dibujando cómo evoluciona la construcción del árbol en cada paso. Exprese el orden de ejecución del algoritmo (en cuanto a su eficiencia). Justifique su respuesta.



GRAFO 1

(a,f) 🡪 1

(d,g) 🡪 1

(h,e) 🡪 1

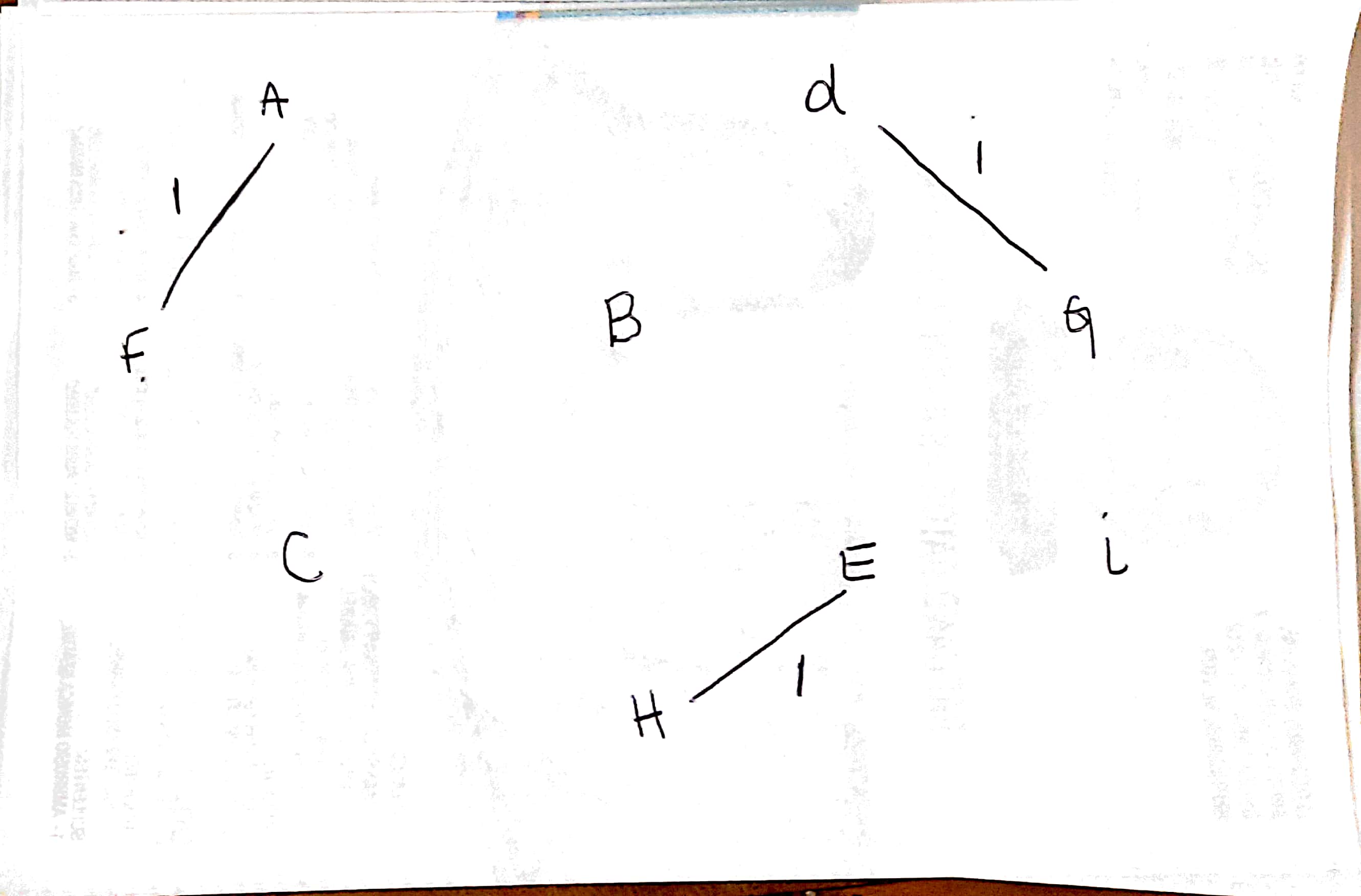
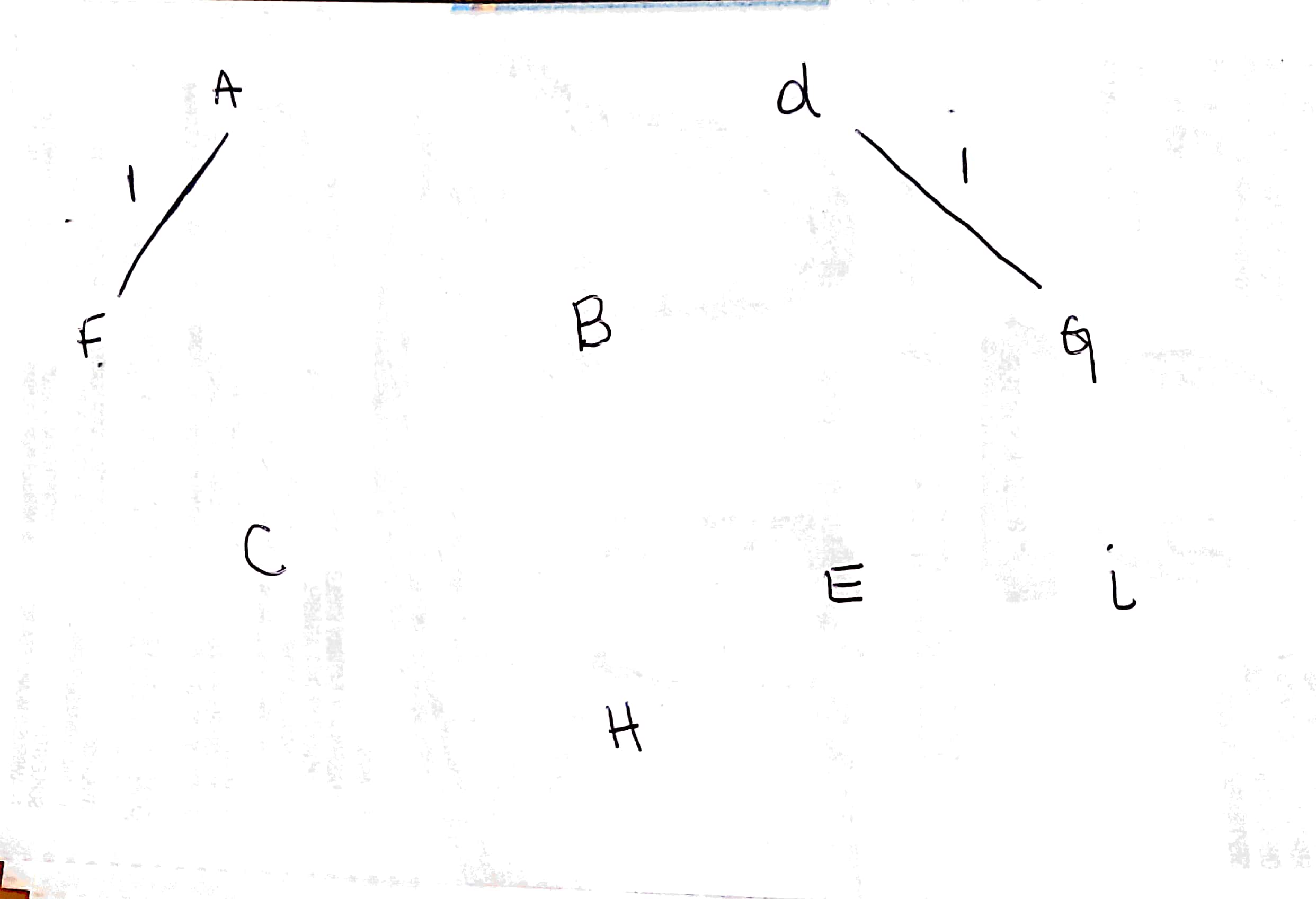
(c,b) 🡪 2

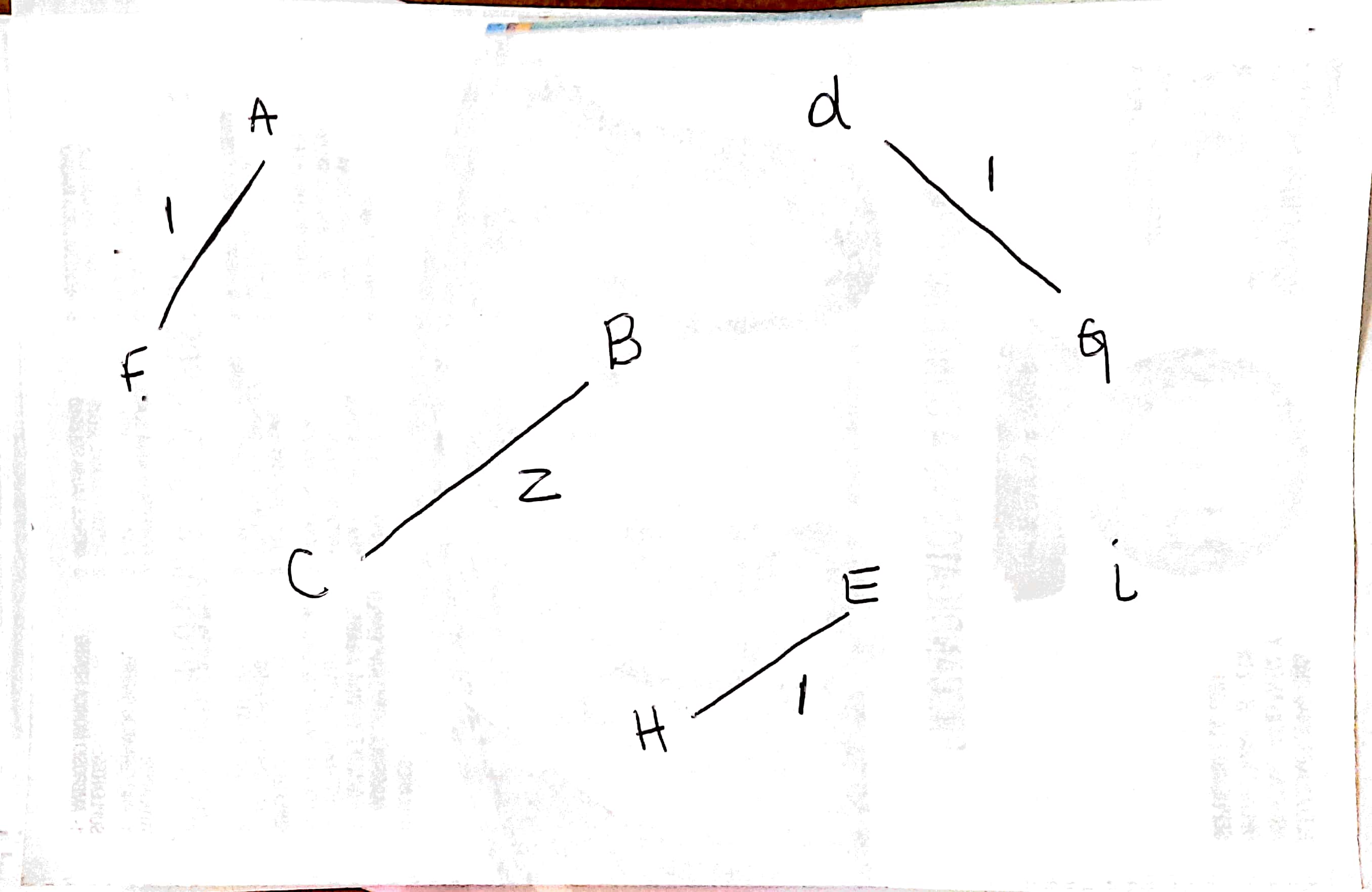
(e,i) 🡪 2

(a,b) 🡪 3

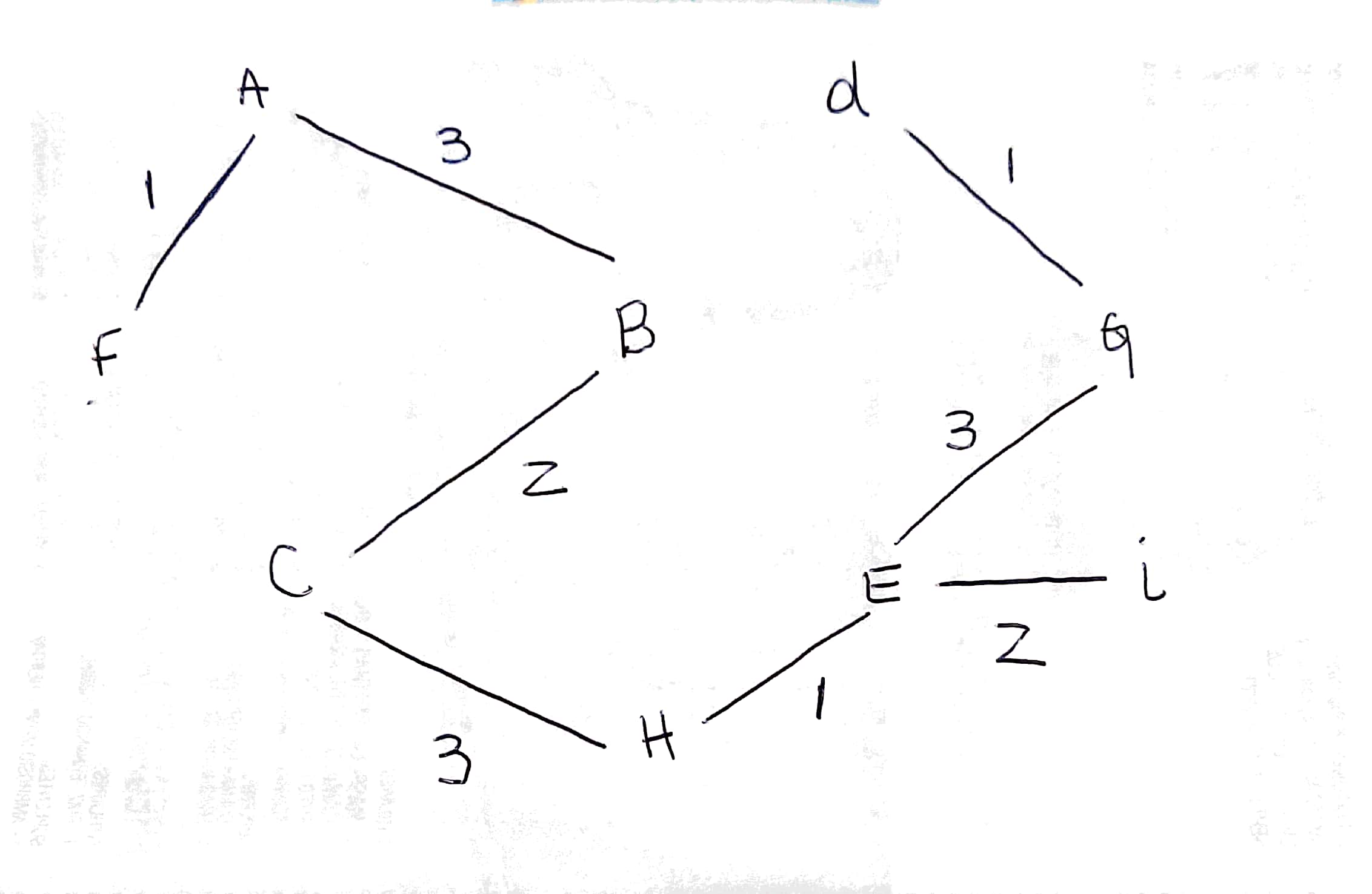
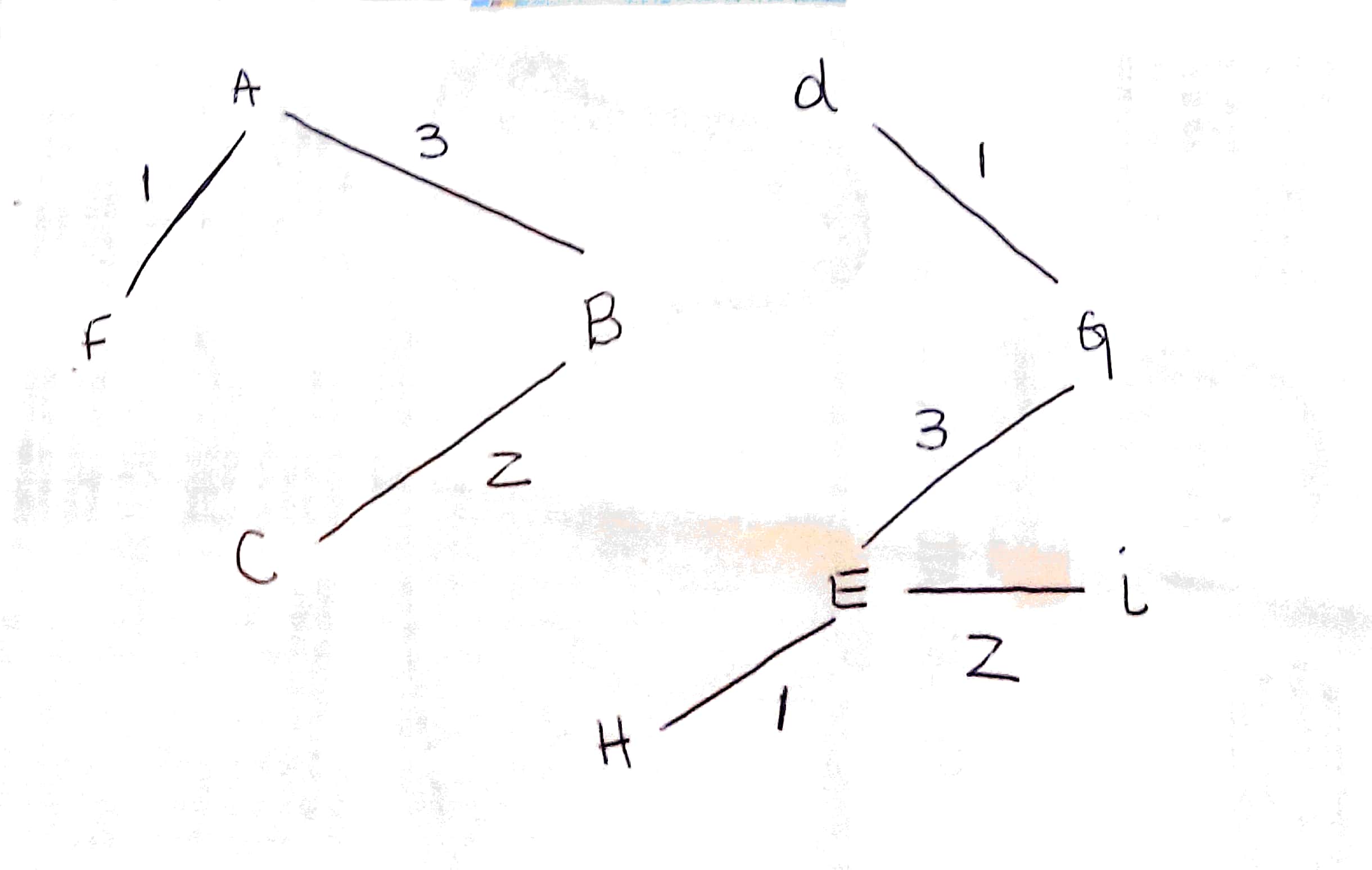
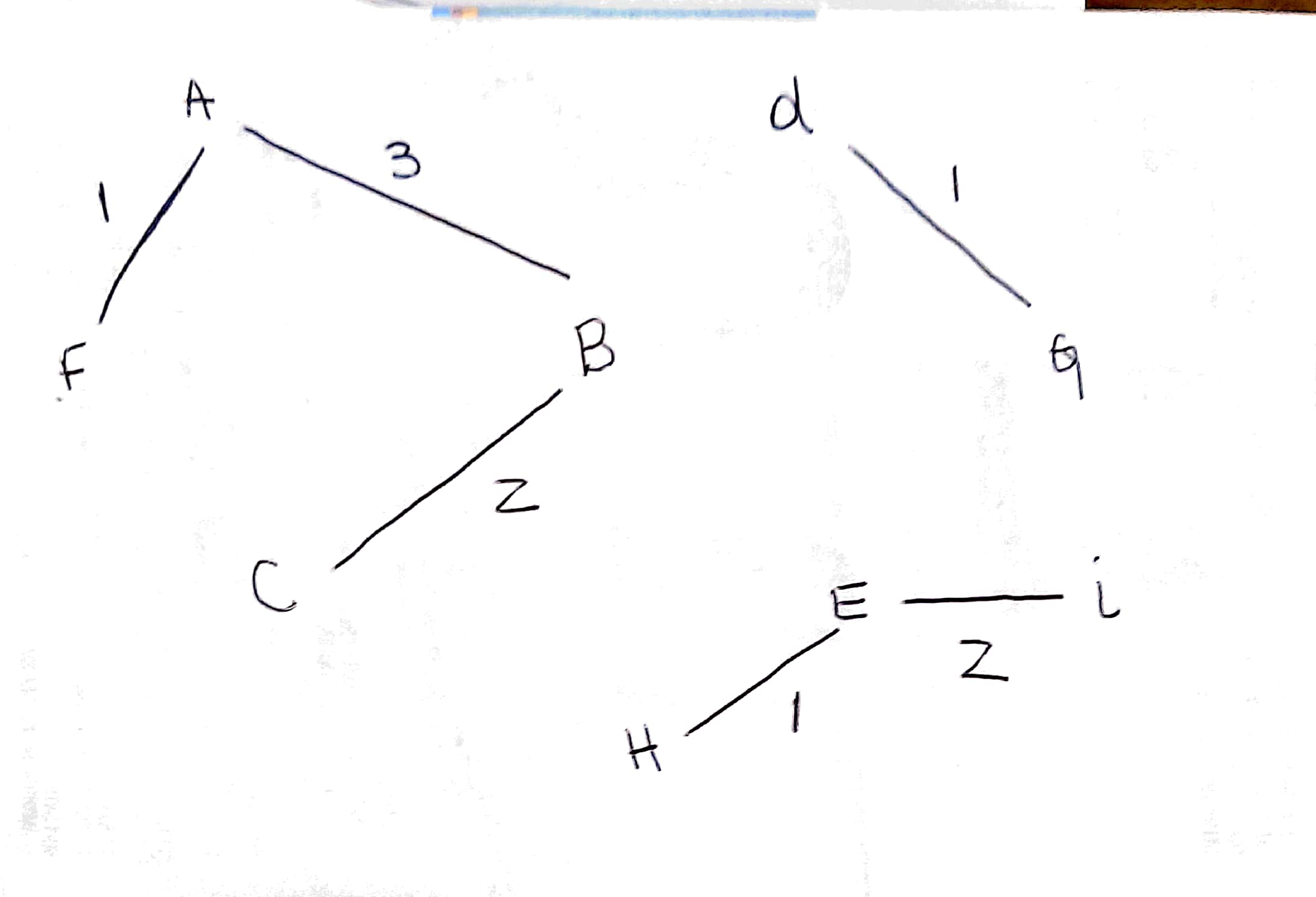
(e,g) 🡪 3

(c,h) 🡪 3









Grafo 2

(a,b) 🡪 2

(e,b) 🡪 2

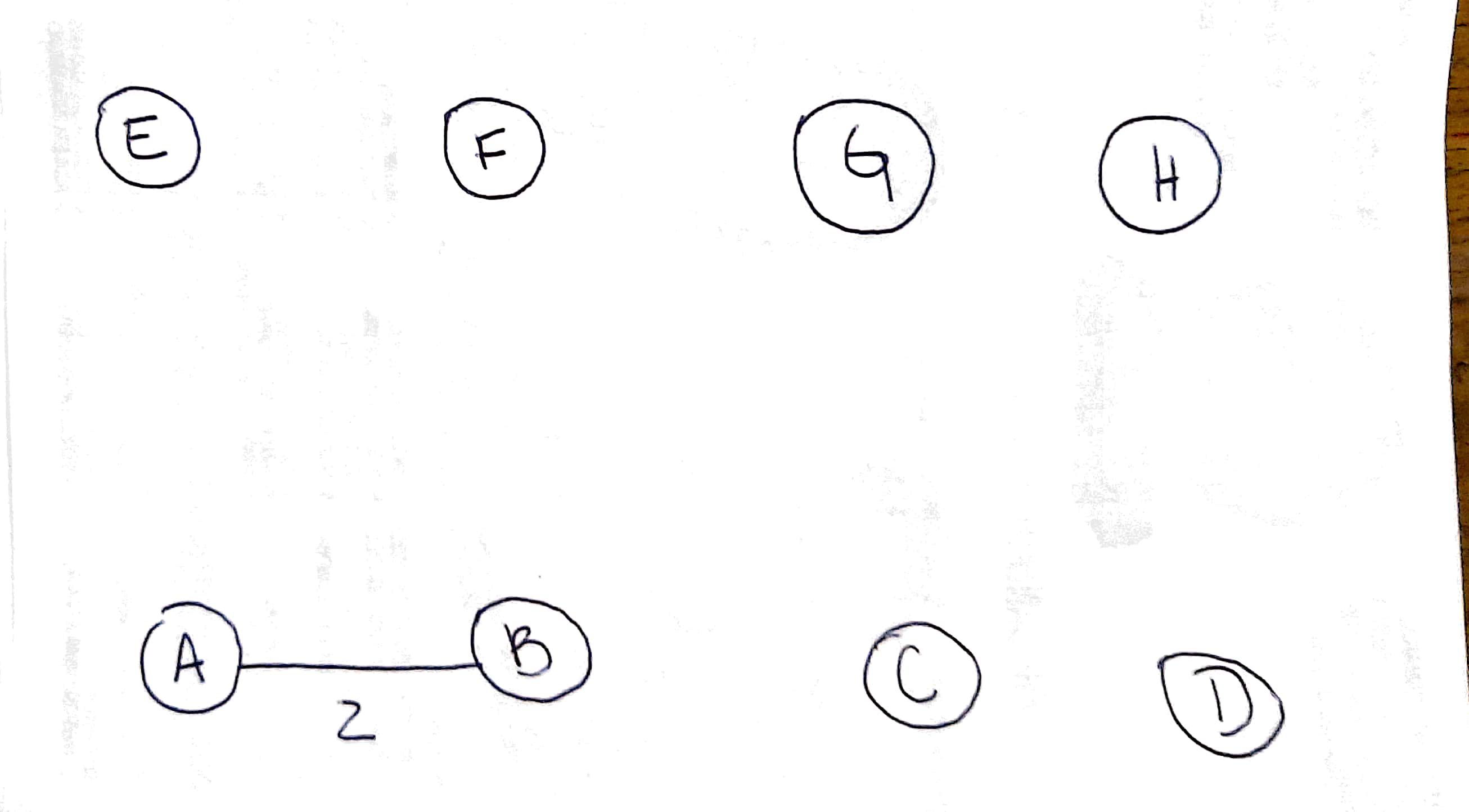
(f,g) 🡪 3

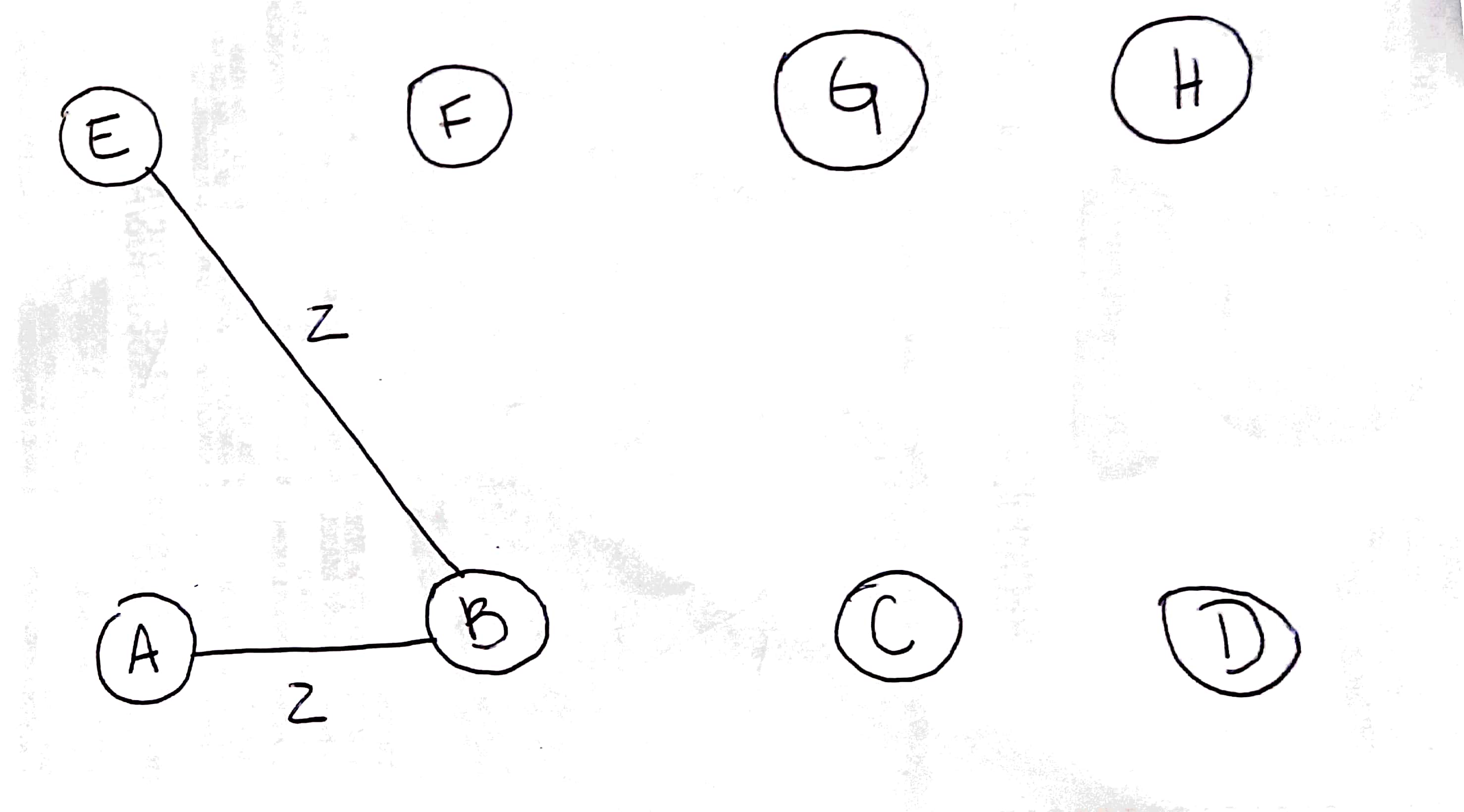
(c,d) 🡪 3

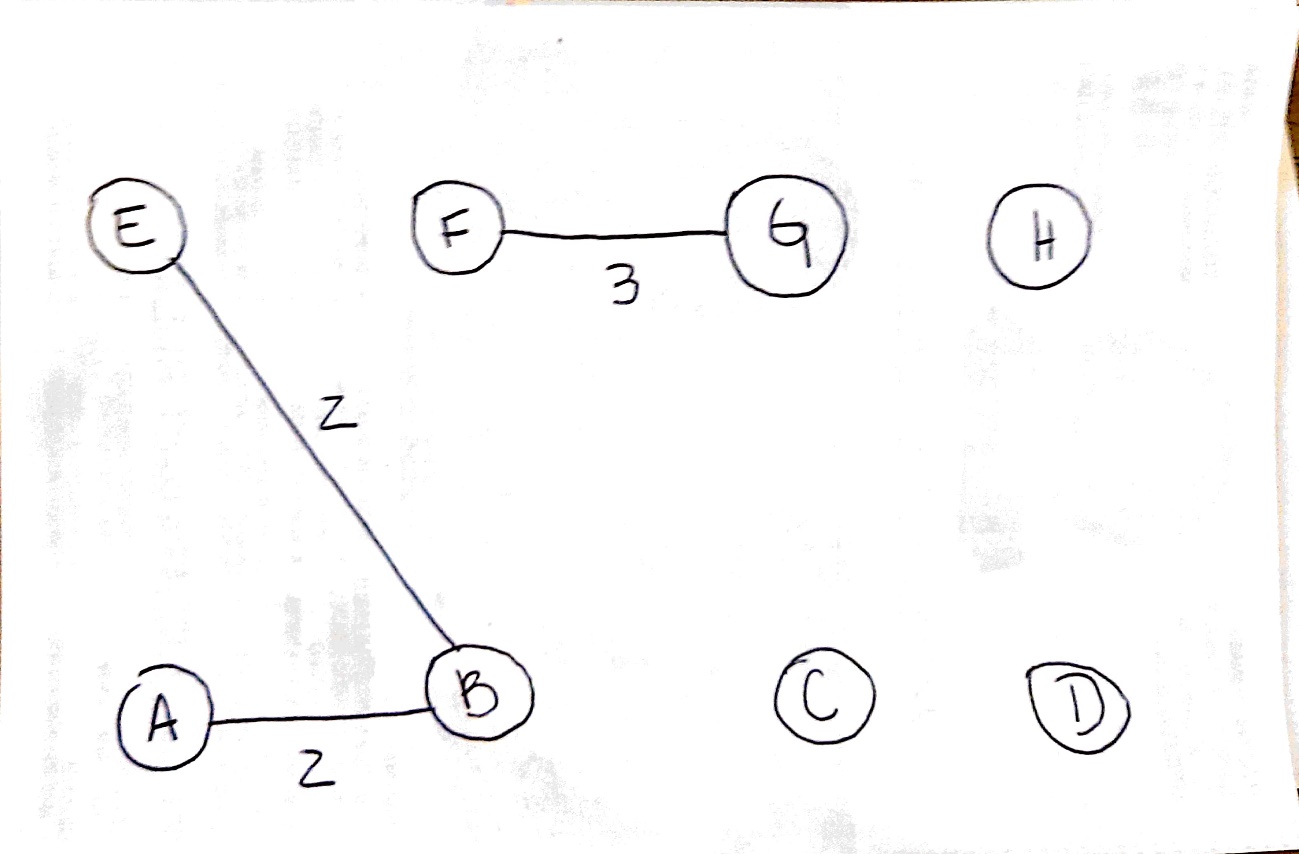
(h,d) 🡪 4

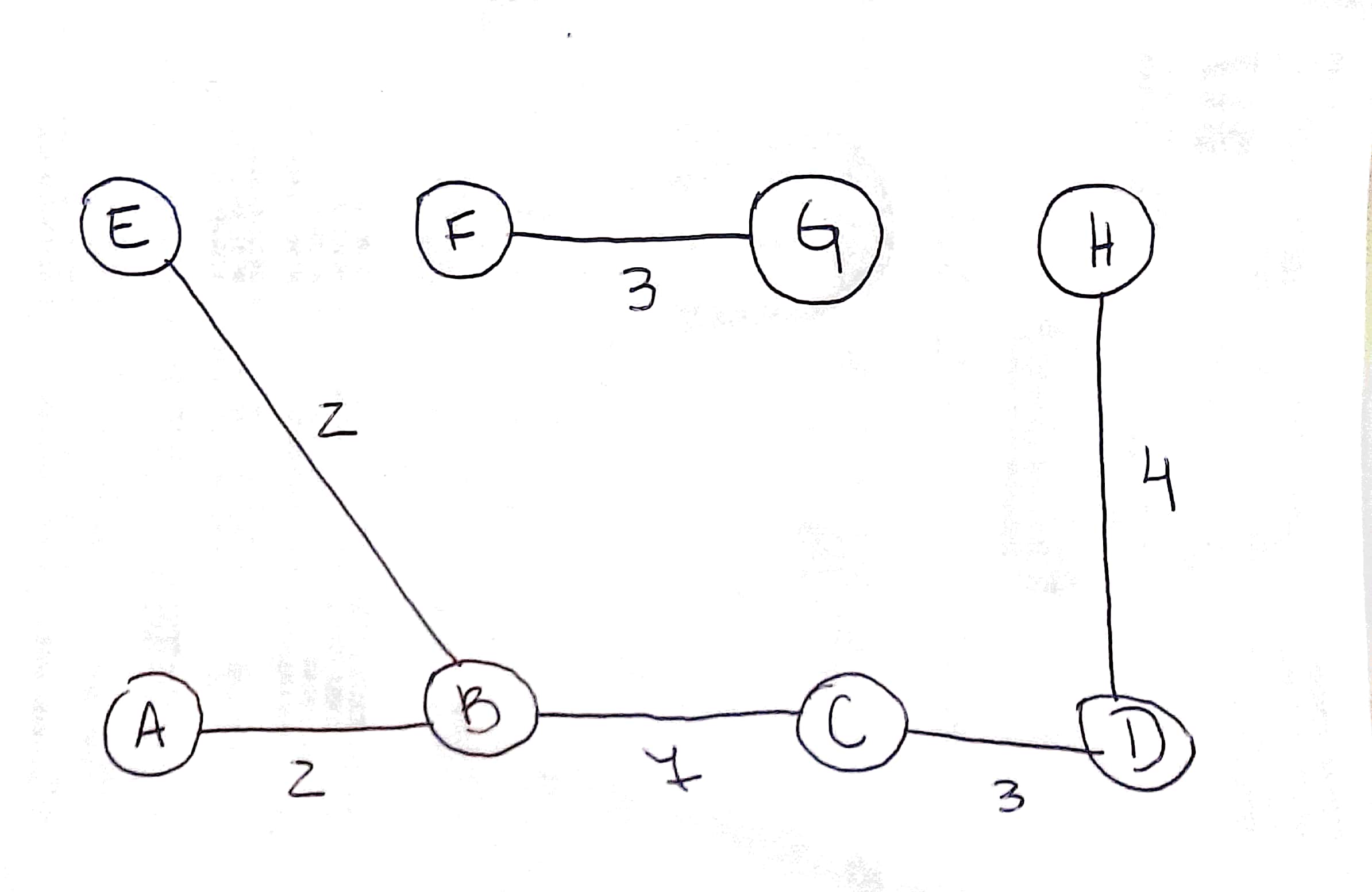
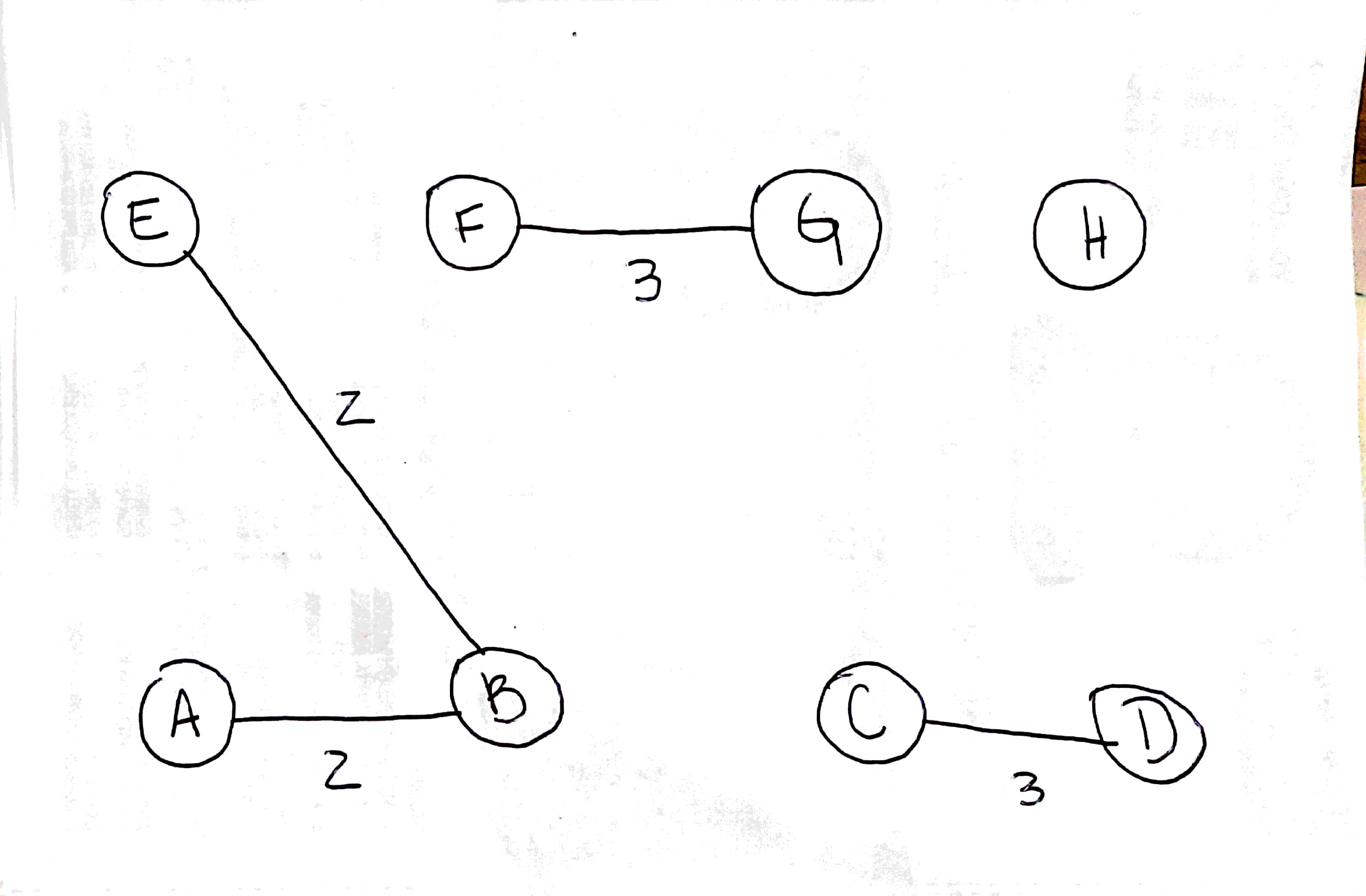
(b,c) 🡪 7

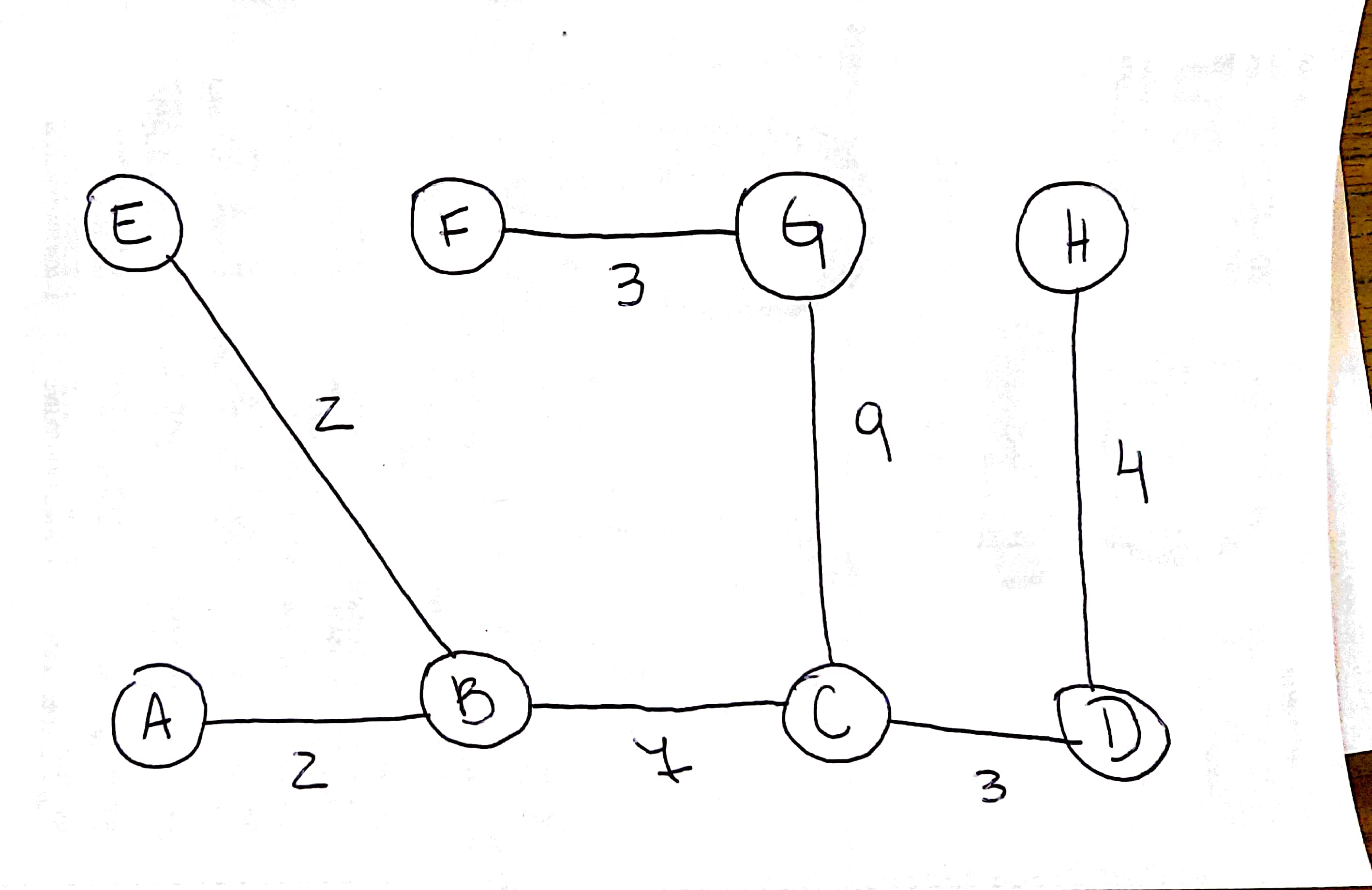
(g,c) 🡪 9











1. Complete la secuencia de arcos del árbol abarcador mínimo, según el orden en que el algoritmo los incluye en el árbol

Grafo 1:

1. (a,f) 🡪 1
2. (d,g) 🡪 1
3. (h,e) 🡪 1
4. (c,b) 🡪 2
5. (e,i) 🡪 2
6. (a,b) 🡪 3
7. (e,g) 🡪 3
8. (c,h) 🡪 3

Grafo 2:

1. (a,b) 🡪 2
2. (e,b) 🡪 2
3. (f,g) 🡪 3
4. (c,d) 🡪 3
5. (h,d) 🡪 4
6. (b,c) 🡪 7
7. (g,c) 🡪 9
8. ¿Cuál es el costo del árbol abarcador resultante?

Grafo 1: 16

Grafo 2: 30