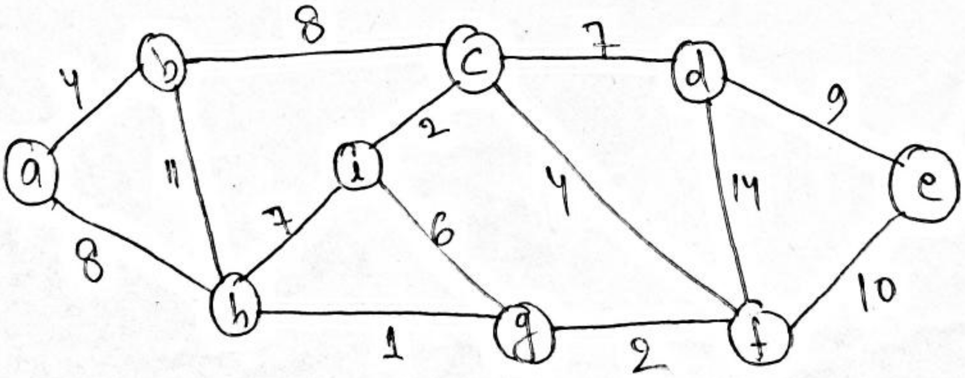
Prim's algorithm

- Prim's algorithm is also used for constructing Minimum Spanning Tree
- Prim's algorithm is also based on greedy approach.
- Prim's algorithm operates much like Dijkstra's algorithm for finding shortest paths in a graph.
- Prim's algorithm has the property that the edges in the set A always form a single tree. The tree starts from an arbitrary root vertex r and grows until the tree spans all the vertices in V.
- To apply Prim's algorithm, the given graph must be weighted, connected and undirected.

ex:→



key[v] : is the minimum weight of any edge connecting v to a vertex in the tree, by convention key[v] = ∞ if there is no such edge.

π[v] : denote the parent of v in the tree.

$A = \{(v, \pi[v]) : v \in V - \{r\} - Q\}$

MST-PRIM (G, w, r)

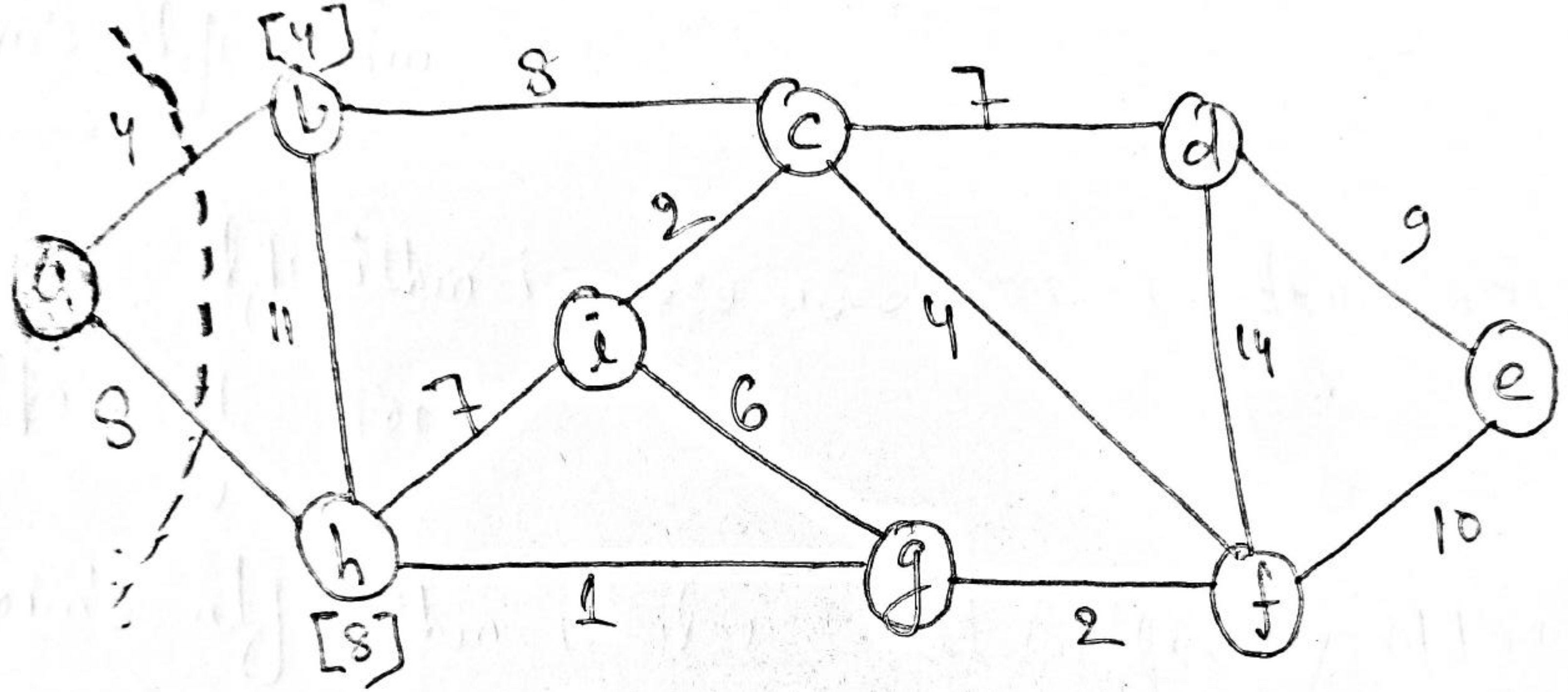
```
for each u ∈ V[G]
    do key[u] ← ∞
        π[u] ← NIL
key[r] ← 0
Q ← V[G]
while Q ≠ ϕ
    do u ← EXTRACT-MIN (Q)
        for each v ∈ Adj[u]
            do if v ∈ Q and w(u,v) < key(v)
                then π[v] ← u
                    key[v] ← w(u,v)
```

when the algorithm terminates, the min-Priority Q is empty, the minimum spanning tree A of G is thus

$$A = \{ (v, \pi[v]) : v \in V - \{r\} \}.$$

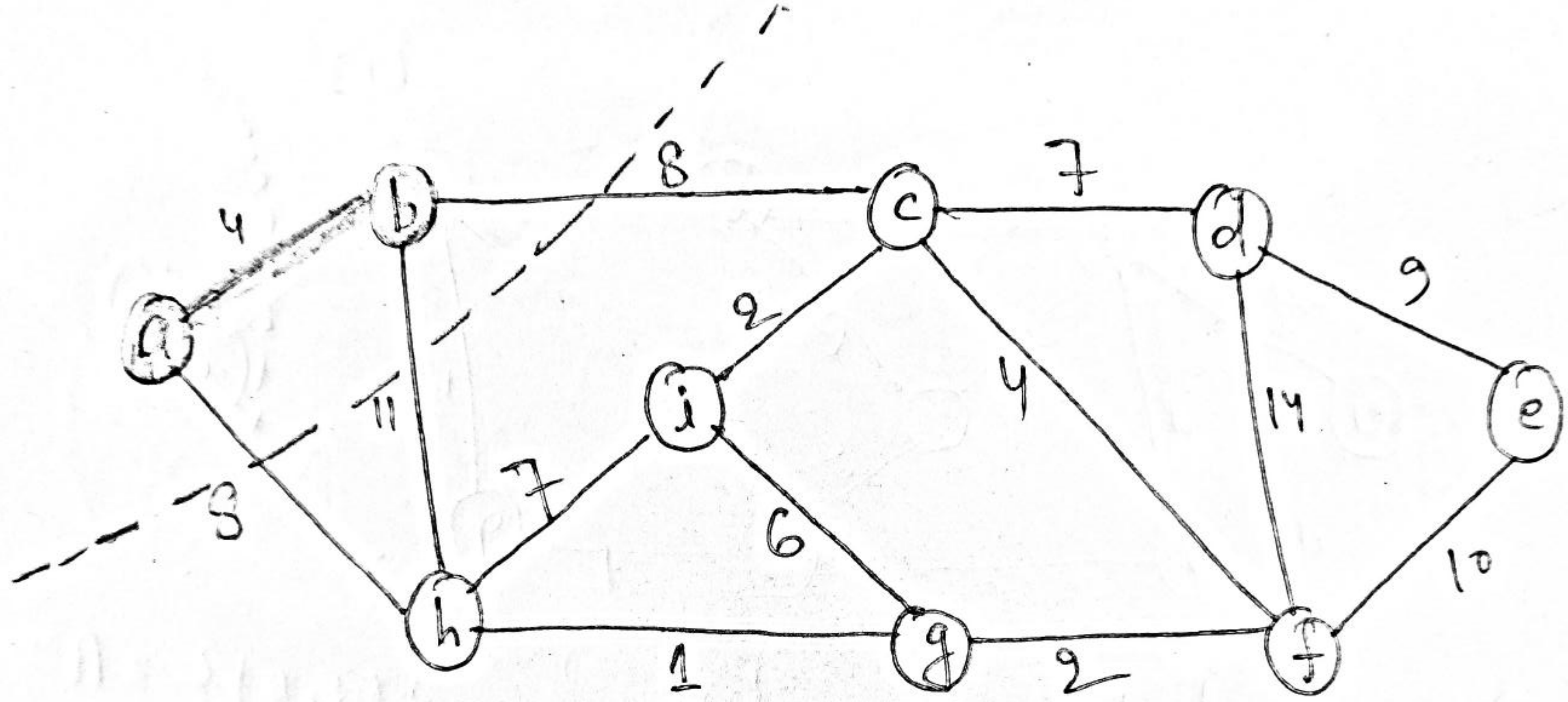
→ Lines 1-5 set the key of each vertex to ∞ (except for the root r, whose key is set to 0 so that it will be the first vertex processed), set the parent of each vertex to NIL, and initialize the min-Priority queue Q to contain all the vertices.

→ Line 7 identifies a vertex $u \in Q$ incident on a light edge crossing the cut $(V-Q, Q)$ (with the exception of the first iteration, in which $u = r$ due to line 4). Removing u from the set Q adds it the set $V-Q$ of vertices in the tree, thus adding $(u, \pi(u))$ to A.



select $b \in Q$ incident on a light edge crossing the cut
$(\{a\}, \{b, c, d, e, f, g, h, i\})$

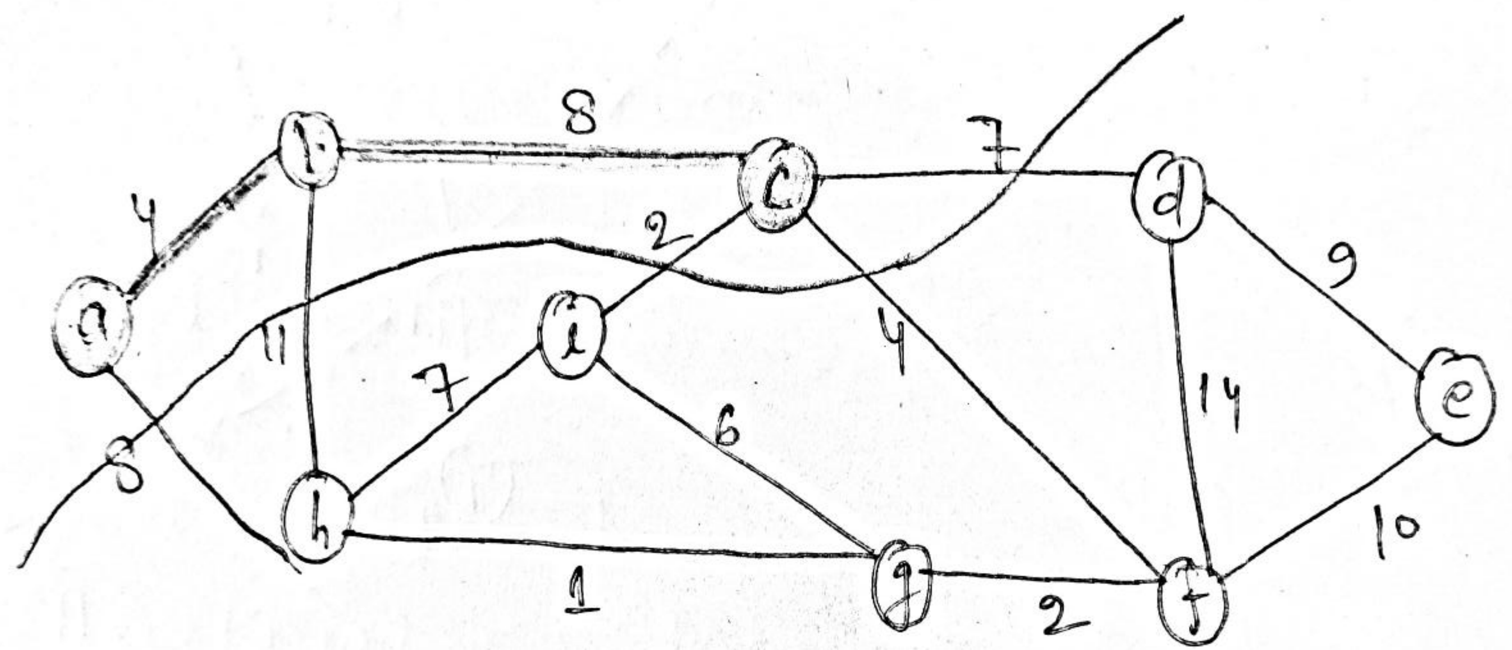
$A = \{(a, b)\}$



~~A = {[illegible]}~~

Select $c \in Q$ incident on a light edge crossing the cut $(\{a,b\}, \{c,d,e,f,g,h,i\})$

$A = \{(a,b), (b,c)\}$

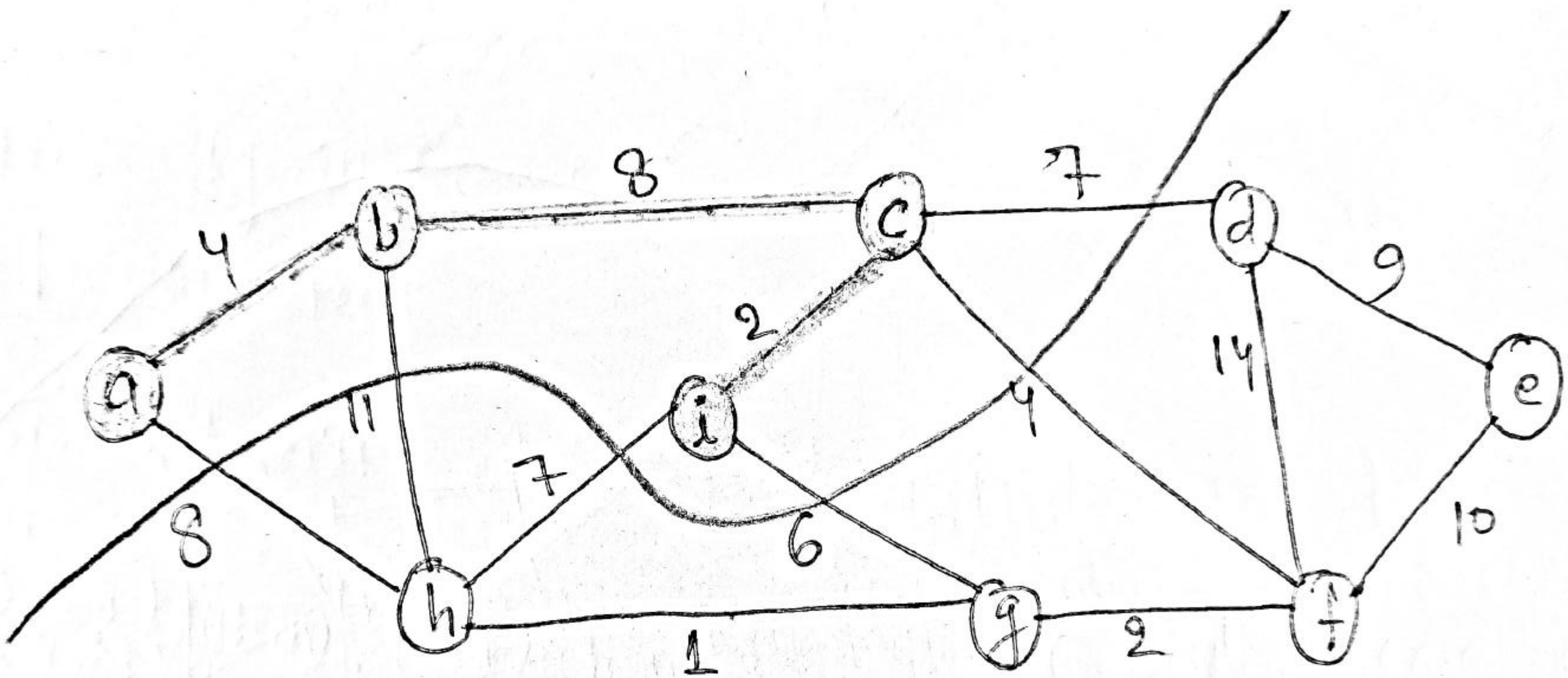


$A = \cancel{\{(a,b),(b,c),(c,i)\}}$

select $i \in Q$ incident on a light edge crossing the cut $(\{a,b,c\}, \{d,e,f,g,h,i\})$.

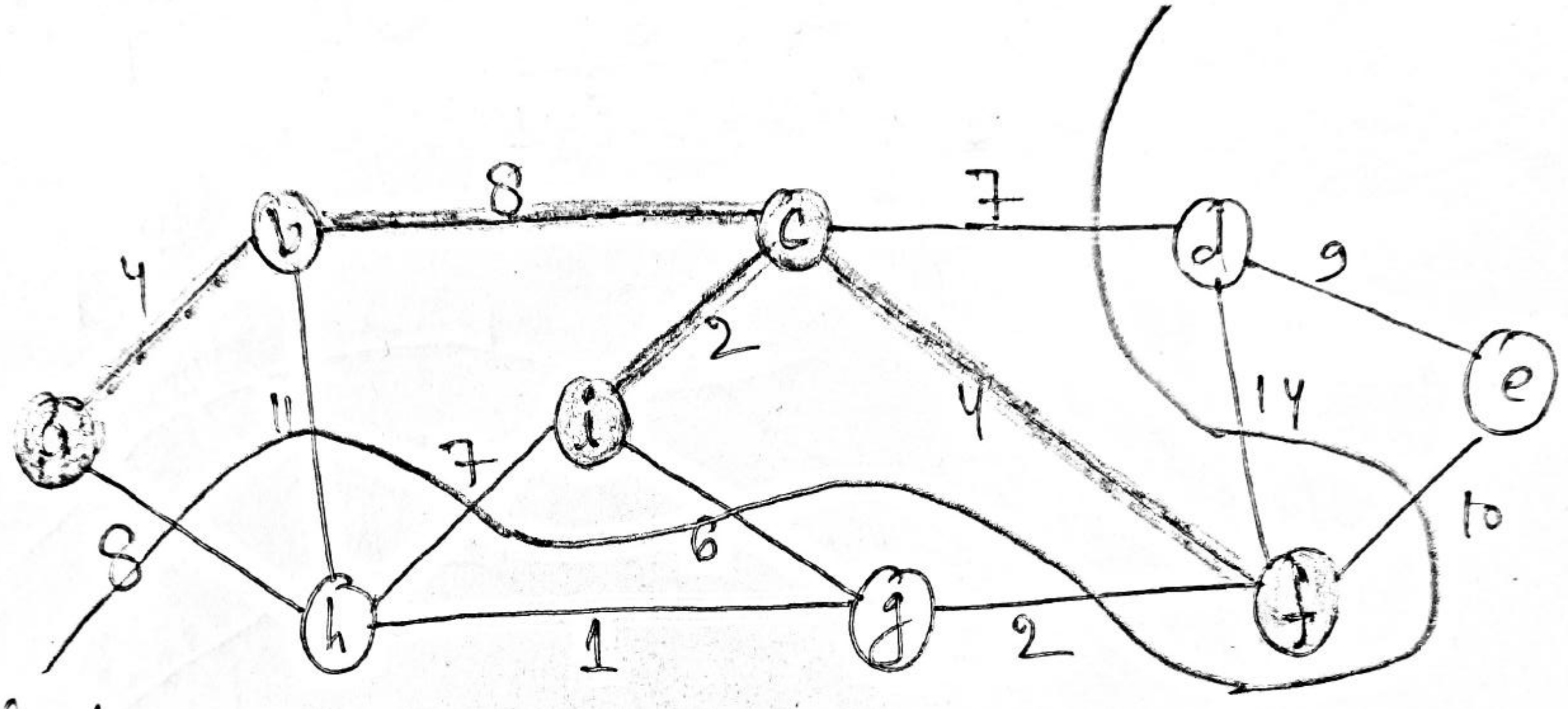
$A = \{(a,b),(b,c),(c,i)\}$

d)



$A = \{ (a,b), (b,c), (c,i), (c,f) \}$ $\Bigg\{$ select $f \in \mathbb{Q}$ incident on a light edge crossing the cut $(\{a,b,c,i\}, \{d,e,f,g,h\})$.

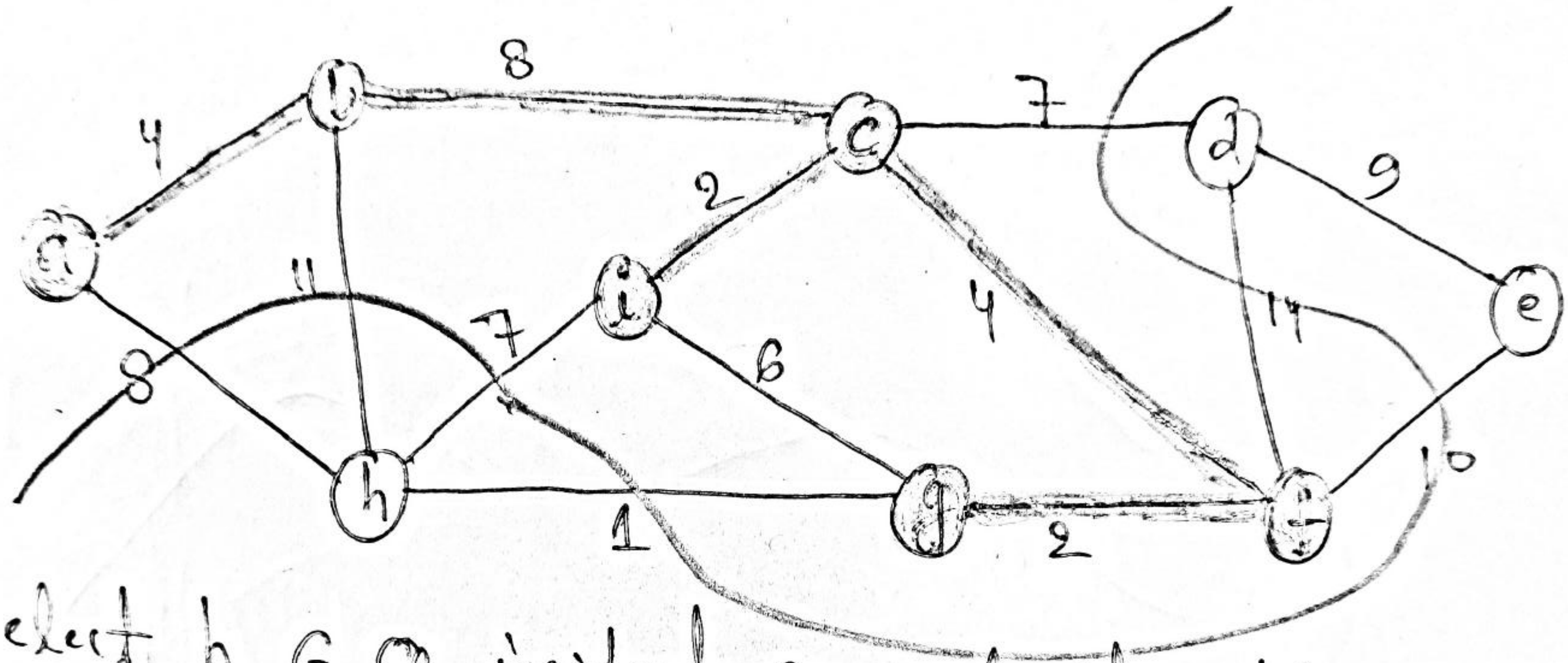
e)



Select $g \in Q$ incident on a light edge crossing the cut $(\{a, b, c, f, i\}, \{d, e, g, h\}.)$

$A = \{(a, b), (b, c), (c, i), (c, f), (f, g)\}.$

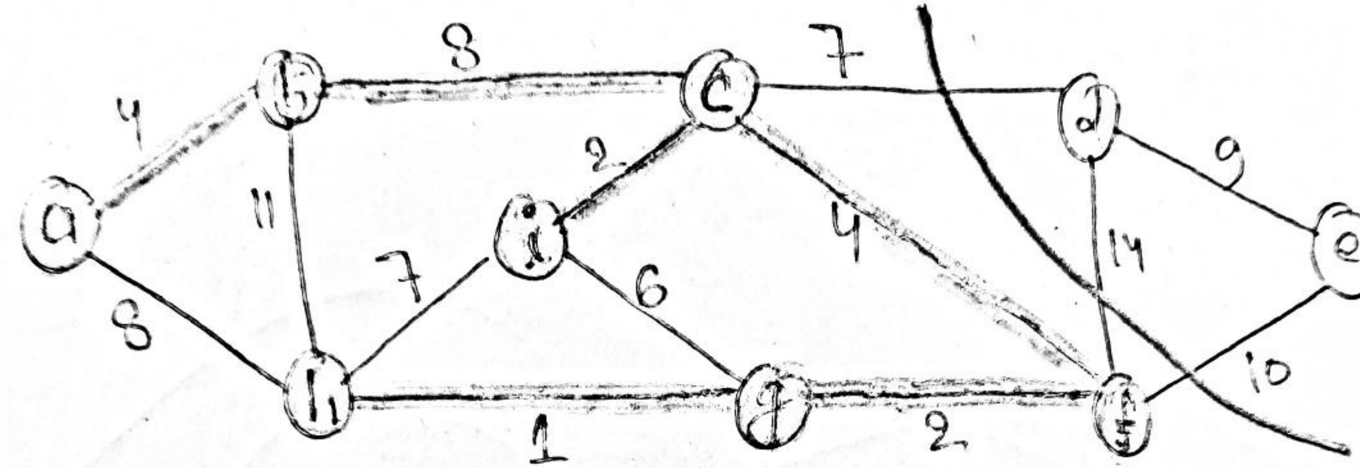
f)



select $h \in Q$ incident on a light edge crossing the cut $\{(a, b, c, i, f, g), \{d, e, h\}\}$

$A = \{(a,b), (b,c), (c,i), (c,f), (f,g), (g,h)\}$

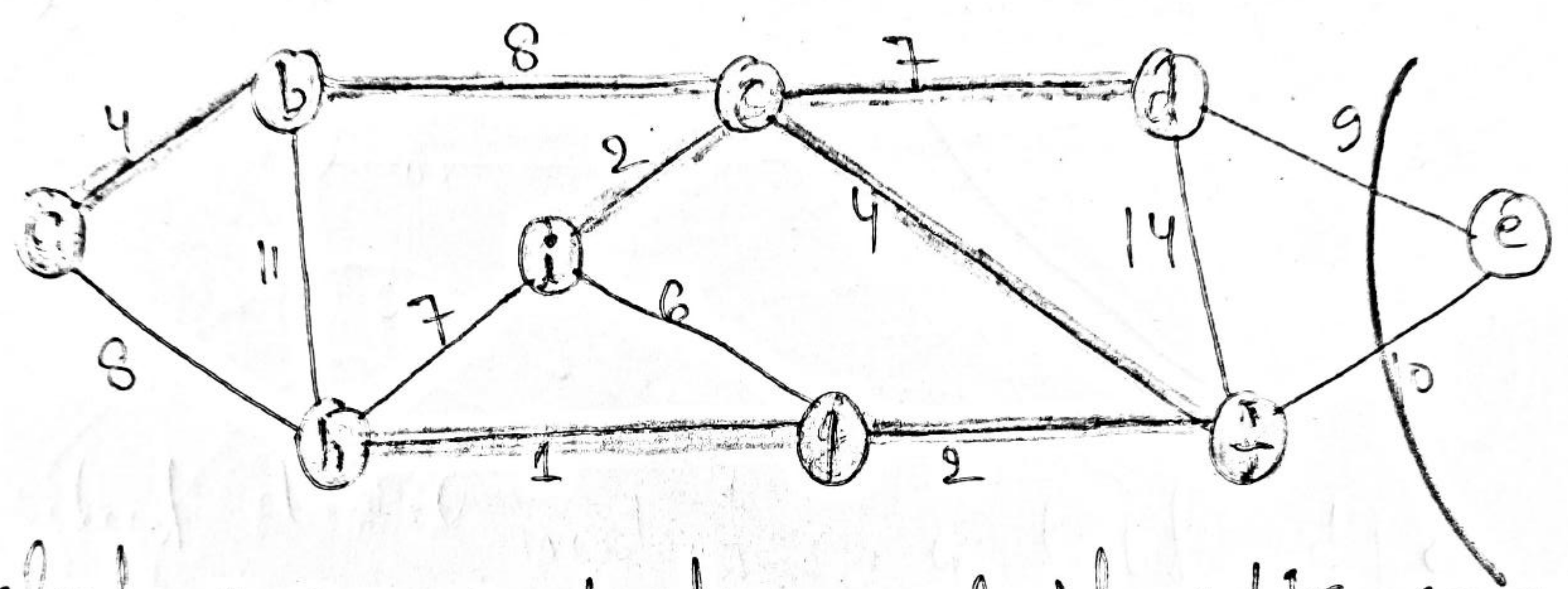
8)



Select d ∈ Q incident on a light edge crossing the cut $\{(a, b, c, f, g, h, i\}, \{d, e)\}$

$A = \{(a, b), (b, c), (c, i), (c, f), (f, g), (g, h), (c, d)\}$

h>



Select $e \in Q$ incident on a light edge crossing the cut $\{(a,b,c,d,f,g,h,i), \{e\}\}$

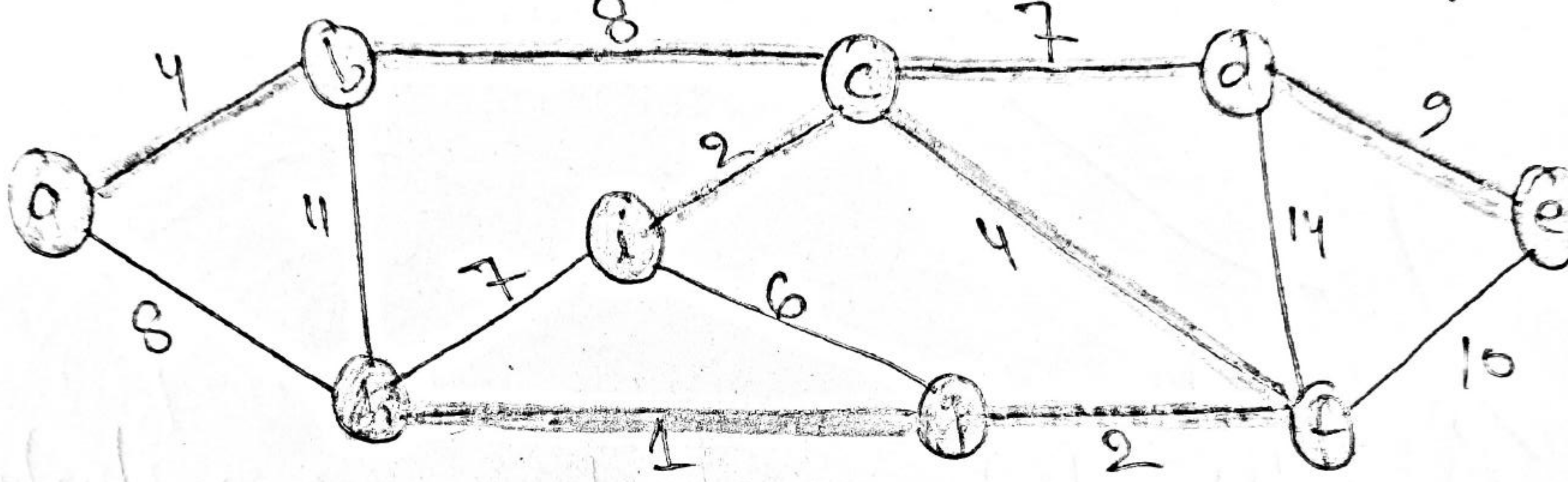
$A = \{(a,b), (b,c), (c,i), (c,f), (f,g), (g,h), (c,d), (d,e)\}$

[illegible]

Cost of minimum spanning tree

= [illegible]

Now. $V - Q = \{a, b, c, d, e, f, g, h, i\}$
and $Q = \{\ \}$.



Cost of Minimum Spanning Tree

$= 4 + 8 + 7 + 9 + 2 + 4 + 2 + 1$

$= 37$