Prim's algorithm

```
MST(G, w, r)
1. for each u ∈ G.V
2.       u.key = ∞
3.       u.π = NIL
4. r.key = 0
5  Q = G.V
6. while Q ≠ ∅
7.       u = EXTRACT-MIN(Q)
8.       for each vertex v ∈ G.Adj[u]
9.             if v ∈ Q and w(u,v) < v.key
10                  v.π = u
11.                 v.key = w(u,v)
```

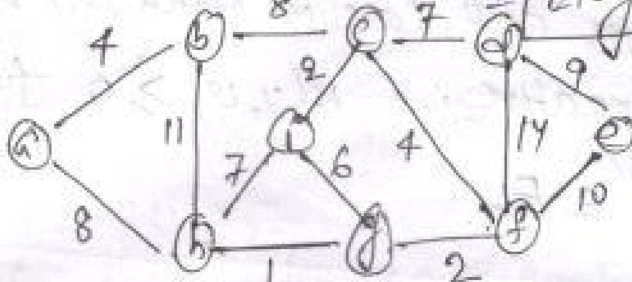
Time complexity : If we implement as a binary heap. We can use the BUILD-MIN-HEAP procedure to perfor line 1-5 in $O(v)$ time. The body of while loop execute $|V|$ and EXTRACT-MIN-HEAP operation takes of $O(\log v)$ time. So over all time to ~~extract~~ call Extract-MIN-HEAP is $O(v\log$

→ The for loop in line 8-11 executes $|E|$ time altogether, since the sum of the lengths of all adjacency is $2E$. Within the for loop we can implement the test for memb in Q in line 9 in constant time. The assignment of in line 11 update the min value in heap hence it will again $\log v$ time.

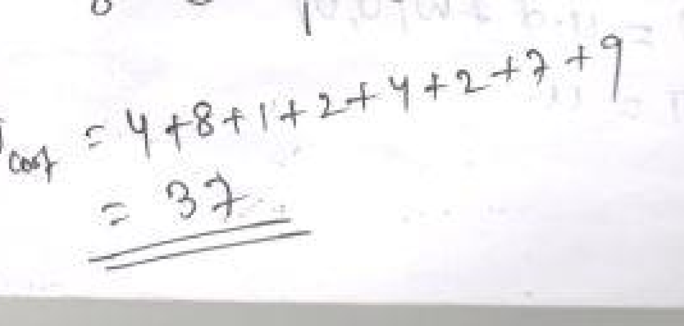
hence over all complexity $= V(\log V) + E\log V$

$= (V+E)\log V$

$\approx \boxed{E\log V}$ when $E >> V$



	vertex	key[v]	parent
x	a	0	NIL
x	b	~~∞~~ 4	~~NIL~~ a
x	c	~~∞~~ ~~8~~ 4	~~NIL~~ ~~b~~ f
x	d	~~∞~~ ~~14~~ 7	~~NIL~~ ~~f~~ c
	e	~~∞~~ ~~10~~ 9	~~NIL~~ ~~f~~ d
x	f	~~∞~~ 2	~~NIL~~ g
x	g	~~∞~~ 1	~~NIL~~ h
x	h	~~∞~~ ~~8~~ (mistake hai pay attention)	~~NIL~~ ~~a~~
x	i	~~∞~~ ~~7~~ ~~6~~ 2	~~NIL~~ ~~h~~ ~~g~~ c



$MST_{cost} = 4+8+1+2+4+2+7+9$

$= \underline{\underline{37}}$

DIJKSTRA ALGORITHM

→ Dijkstra's algorithm solves the single-source shortest-paths problem on a weighted, directed graph $G = (V, E)$ for each case in which all edge weights are non-negative i.e $w(u,v) \geq 0$ for each edge $(u,v) \in E$.

INITIALIZE-SINGLE-SOURCE(G,s)

1. for each vertex $v \in G.V$
2. $v.d = \infty$
3. $v.\pi = NIL$
4. $s.d = 0$

⇒ Relaxation is a process ~~to~~ of relaxing edge (u,v) consist of testing whether we can improve the shortest path to v found so far by going through u and, if so updating $v.d$ and $v.\pi$.

RELAX(u, v, w)

1. if $v.d > u.d + w(u,v)$
2. $v.d = u.d + w(u,v)$
3. $v.\pi = u$.

DIJKSTRA(G, W, S)

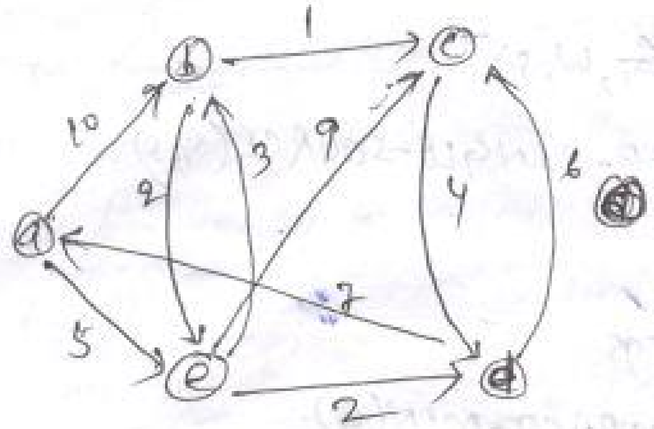
1. INITIALIZE-SINGLE-SOURCE(G, S)
2. $S = \emptyset$
3. $Q = G.V$
4. while $Q \neq \emptyset$
5. $u = $ EXTRACT-MIN(Q)
6. $S = S \cup \{u\}$.
7. for each vertex $v \in G.Adj[u]$
8. RELAX(u, v, w)

Analysis

~~Each Extract~~

→ Each EXTRACT-MIN(Q) operation takes $O(\log V)$ and the are V such operation. The time to build binary-min-heap is $O(V)$. Each DECREASE-KEY operation takes time $O(\log V)$ and there are at most |E| such operation. The total running time thereafter is $O((V+E)\log V)$

Ex



Vertex	v.d	v.π	
a	0	NIL	1
b	~~∞~~ ~~10~~ 8	~~NIL~~ ~~a~~ e	4
c	~~∞~~ ~~14~~ ~~13~~ 9	~~NIL~~ ~~e~~ ~~d~~ b	
d	~~∞~~ 7	~~NIL~~ e	3
e	~~∞~~ 5	~~NIL~~ a	2