

PRIM'S MINIMUM COST SPANNING TREE ALGORITHM

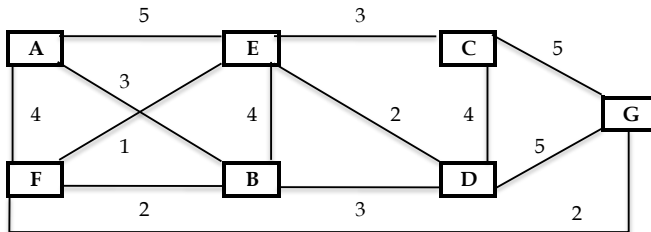
- 1 Uses BFS
- 2 Creates a tree selecting one [most beneficial] edge at a time & minimizes the overall cost of reaching to $|V|$.

Pre-Conditions:

- 1 Edge cost may be negative.
- 2 Should not contain a negative cost cycle.
- 3 Weighted undirected graph

Required Data Structures:

- 1 A Queue [an array Visited[] to keep track of already visited vertices].
Also a minHeap or Fibonacci Heap may also be used.
- 2 Two 1-D arrays of size = $|V|$. Array Length[] will store cost of edge.
Array Parent[] will indicate the parent vertex of the current vertex.

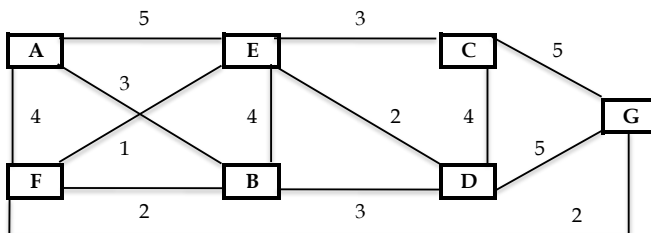


	A	B	C	D	E	F	G
A	0	3	NF	NF	5	4	NF
B	3	0	NF	3	4	2	NF
C	NF	NF	0	4	3	NF	5
D	NF	3	4	0	2	NF	5
E	5	4	3	2	0	3	NF
F	4	2	NF	NF	3	0	2
G	NF	NF	5	5	NF	2	0

Adjacency Matrix

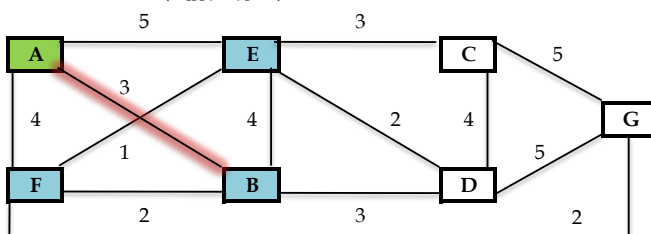
NF := Infinity

Initial State



	A	B	C	D	E	F	G
VISITED[]	F	F	F	F	F	F	F
PARENT[]	-	-	-	-	-	-	-
LENGTH[]	NF	NF	NF	NF	NF	NF	NF

Iteration-01: Call Prim(G[[] | V |], A)

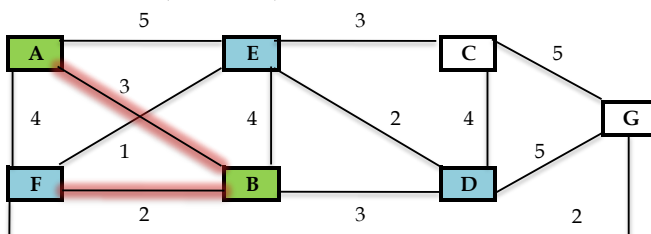


SRC:=	A						
A	0	3	NF	NF	5	4	NF
	A	B	C	D	E	F	G
VISITED[]	T	F	F	F	F	F	F
PARENT[]	-	A	-	-	A	A	-
LENGTH[]	0	3	NF	NF	5	4	NF

minDistV : B

among non-visited vertices

Iteration-02: Call Prim(G[[] | V |], B)

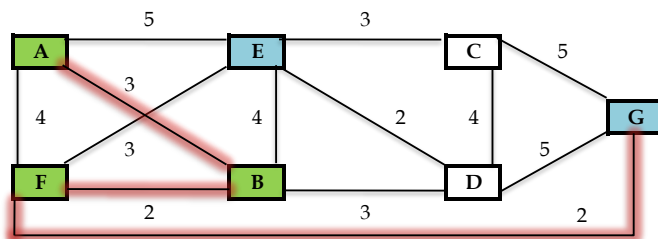


SRC:=	B						
B	3	0	NF	3	4	2	NF
	A	B	C	D	E	F	G
VISITED[]	T	T	F	F	F	F	F
PARENT[]	-	A	-	B	B	B	-
LENGTH[]	0	3	NF	3	4	2	NF

minDistV : F

among non-visited vertices

Iteration-03: Call Prim($G[|V|]$, F)



SRC:= F

F	4	2	NF	NF	3	0	2
	A	B	C	D	E	F	G

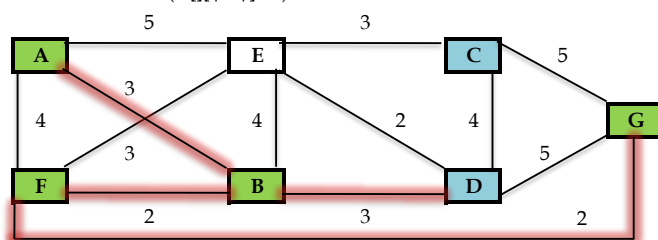
VISITED[]	T	T	F	F	F	T	F
-----------	---	---	---	---	---	---	---

PARENT[]	-	A	-	B	F	B	F
----------	---	---	---	---	---	---	---

LENGTH[]	0	3	NF	3	3	2	2
----------	---	---	----	---	---	---	---

minDistV : G among non-visited vertices

Iteration-04: Call Prim($G[|V|]$, G)



SRC:= G

G	NF	NF	5	5	NF	2	0
	A	B	C	D	E	F	G

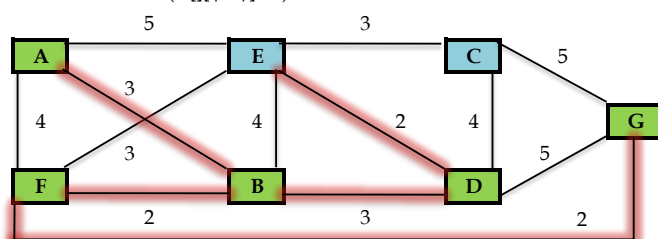
VISITED[]	T	T	F	F	F	T	T
-----------	---	---	---	---	---	---	---

PARENT[]	-	A	G	B	F	B	F
----------	---	---	---	---	---	---	---

LENGTH[]	0	3	5	3	3	2	2
----------	---	---	---	---	---	---	---

minDistV : D among non-visited vertices

Iteration-05: Call Prim($G[|V|]$, D)



SRC:= D

D	NF	3	4	0	2	NF	5
	A	B	C	D	E	F	G

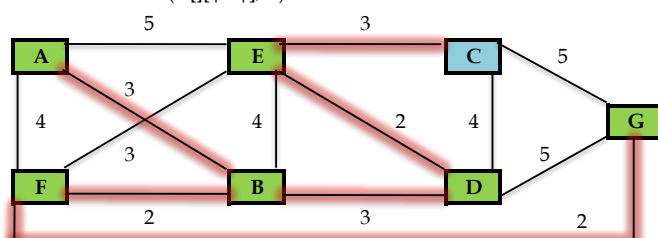
VISITED[]	T	T	F	T	F	T	T
-----------	---	---	---	---	---	---	---

PARENT[]	-	A	D	B	D	B	F
----------	---	---	---	---	---	---	---

LENGTH[]	0	3	4	3	2	2	2
----------	---	---	---	---	---	---	---

minDistV : E among non-visited vertices

Iteration-06: Call Prim($G[|V|]$, E)



SRC:= E

E	5	4	3	2	0	3	NF
	A	B	C	D	E	F	G

VISITED[]	T	T	F	T	T	T	T
-----------	---	---	---	---	---	---	---

PARENT[]	-	A	E	B	D	B	F
----------	---	---	---	---	---	---	---

LENGTH[]	0	3	3	3	2	2	2
----------	---	---	---	---	---	---	---

Prim's MST Terminates on adding $|V| - 1$ most beneficial edges to the Tree.

Thus, the Edges (with cost) of the minimum cost spanning tree [MST] are ...

Edge	Cost
A -- B	3
B -- F	2
F -- G	2
C -- E	3
B -- D	3
D -- E	2

Cost of MST := 15

By : Prof. D. A. Borikar

CSE, RCOEM, NAGPUR

AY - 2019-20