

# Bangladesh Army International University of Science and Technology



## Algorithms

Level-2 Term-II  
Department of CSE

Name: Aftab Uddin Alif

ID No: 1109026

# Prims algorithm

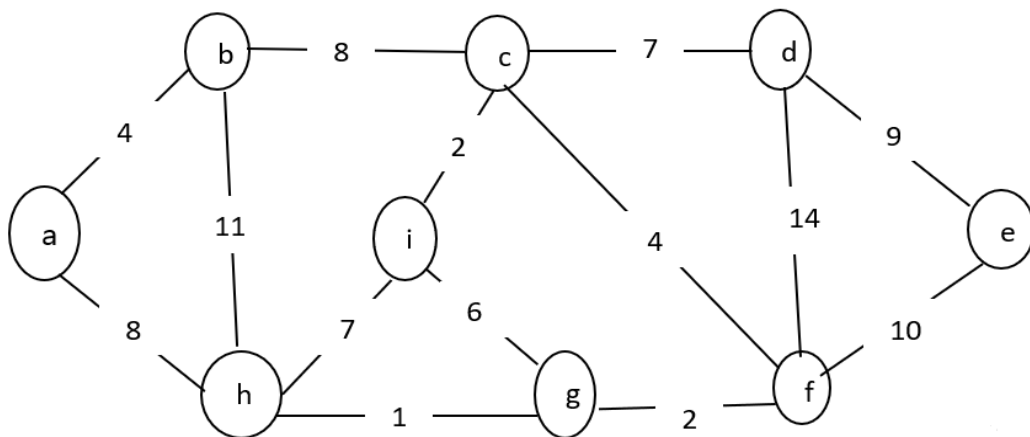
**নিয়মঃ** Prims algorithm এর ক্ষেত্রে প্রথমে starting vertices থেকে সবচেয়ে কম cost দিয়ে পরের vertices এ যেতে হবে। এর পর যতগুলো vertices visit করা আছে তার মধ্যে যে vertices দিয়ে সবচেয়ে কম cost এ অপর vertices এ যাওয়া যায় সে vertices কে select করতে হবে ও একাধিক vertices এর cost সমান হলে যে কোনটি নেওয়া যাবে, তবে কোনো loop হতে পারবে না। সবগুলো vertices visit করা শেষে visit করা edge এর cost গুলো যোগ করে total cost বের করতে হবে।

1 <sup>st</sup> pass	2 <sup>nd</sup> pass	3 <sup>rd</sup> pass	4 <sup>th</sup> pass
5 <sup>th</sup> pass		6 <sup>th</sup> pass	
7 <sup>th</sup> pass		8 <sup>th</sup> pass	
Total cost $4+8+1+2+4+2+7+9=37$			

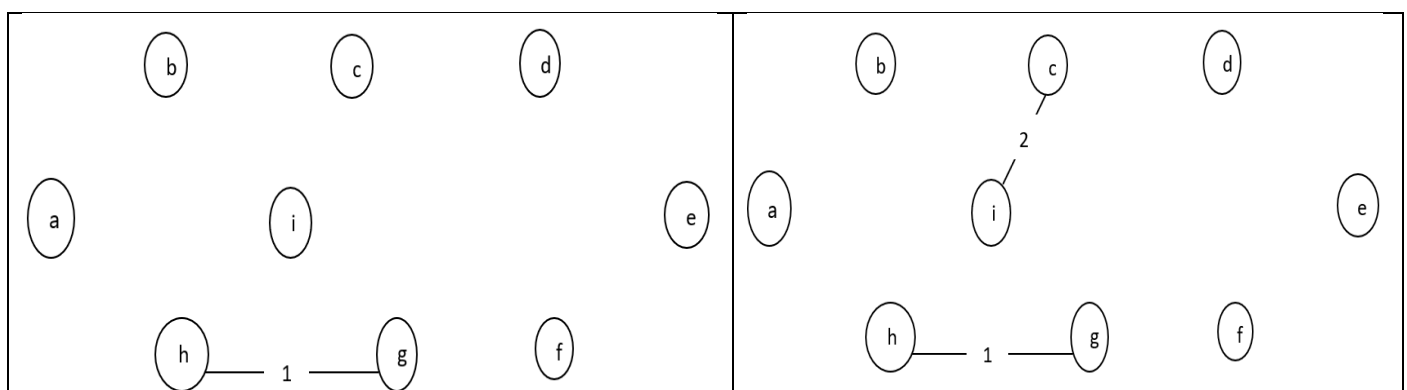
Algorithm	Practice
<b>MST-Prim(<math>G, w, r</math>)</b> 1. for each $u \in G.V$ 2. $u.key = \infty$ 3. $u.\pi = \text{Nil}$ 4. $r.key = 0$ 5. $Q = G.V$ 6. while $Q \neq \emptyset$ 7. $u = \text{EXTRACT-MIN}(Q)$ 8. for each $v \in G.\text{Adj}[u]$ 9. If $v \in Q$ and $w(u, v) < v.key$ 10. $v.\pi = u$ 11. $v.key = w(u, v)$	<p>Ans=13</p>

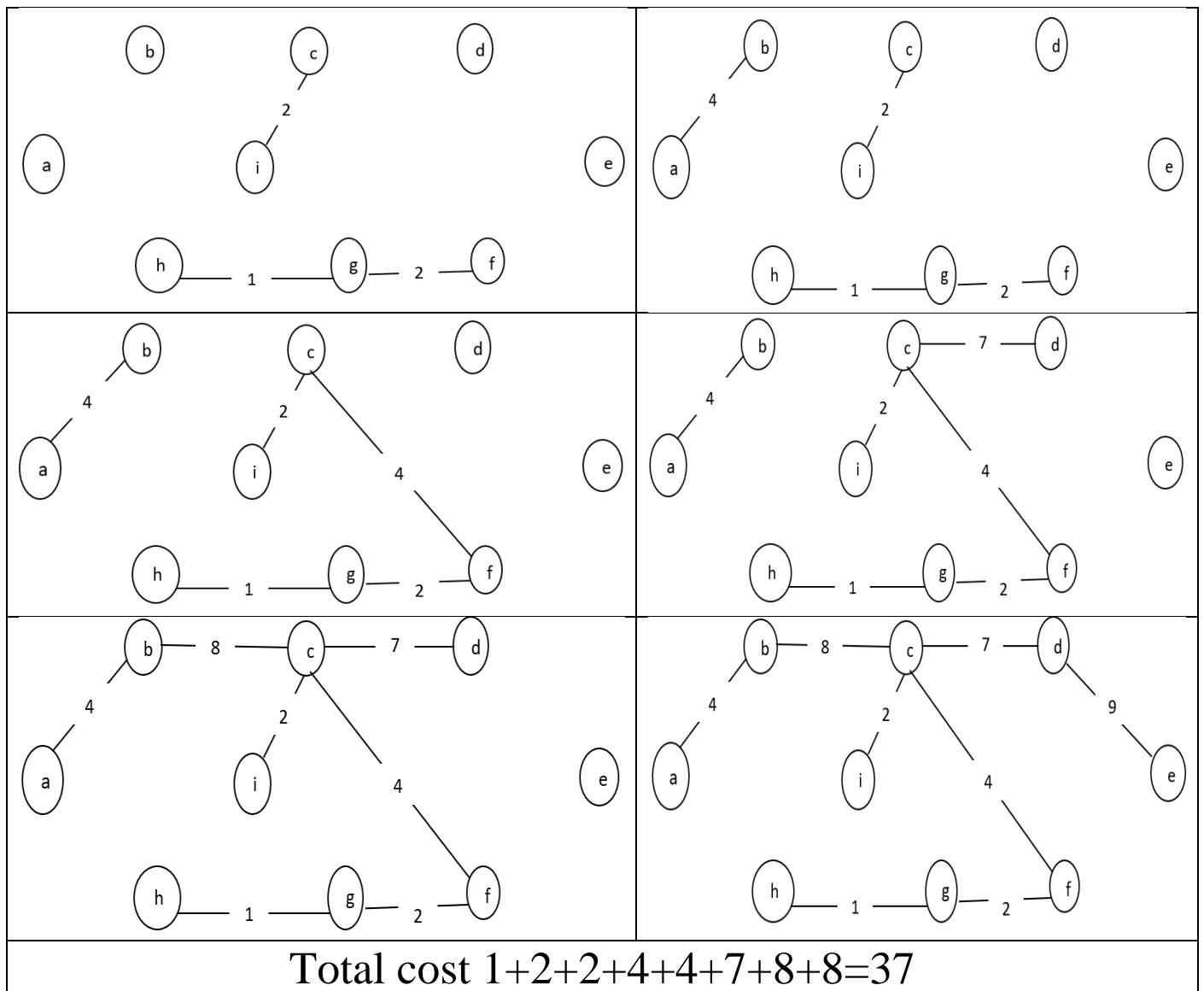
## Kuskal's algorithm

**নিয়মঃ** Kuskal's algorithm এর ক্ষেত্রে প্রথমে সবগুলো edage এর cost গুলো ছোটো থেকে বড় সাজিয়ে নিতে হবে। এর পর ছোটো cost থেকে vertices গুলো visit করতে হবে। কিন্তু কোনো cost নিলে যদি loop হয়ে যায় তাহলে তা বাদ দিতে হবে।



Cost	1	2	2	4	4	6	7	7	8	8	9	10	11	14
Edge	h-g	c-i	g-f	a-b	c-f	i-g	c-d	h-i	b-c	a-h	d-e	e-f	b-h	d-f





Algorithm	Practice
<p><b>MSYT-Kuskal(G,w)</b></p> <ol style="list-style-type: none"> <li>1. <math>A=\emptyset</math></li> <li>2. for each <math>u \in G.V</math></li> <li>3. <b>MAKE-SET</b> (v)</li> <li>4. Sort the edges of <math>G.E</math> into nondecreasing order by weight w.</li> <li>5. for each edge <math>(u,v) \in G.E</math>, taken in nondecreasing order by weight w.</li> <li>6. If <math>\text{FIND-SET}(u) \neq \text{FIND-SET}(v)</math></li> <li>7. <math>A=A \cup \{(u,v)\}</math></li> <li>8. <b>UNION</b>(u,v)</li> <li>9. Return A</li> </ol>	<div style="text-align: center;"> </div> <p><b>Ans=13</b></p>

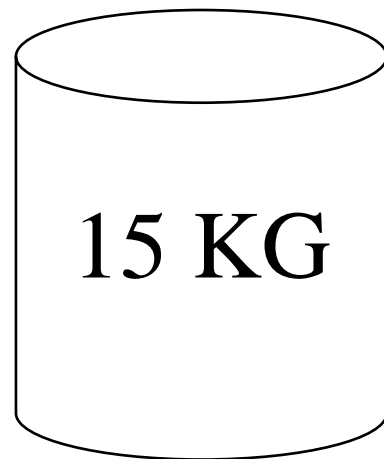
# 0/1 knapsnak Fractional

**নিয়মঃ** 0/1 Knapsnak এর ক্ষেত্রে আমাদের কে সবচেয়ে বড় Max কে নিয়ে কাজ করতে হবে তবে যদি Remaining এ থাকা Weight এর থেকে Max এর Weight বেশি হয় তবে Remaining এ যে Weight আছে তা নিয়ে কাজ করব এবং নিচের নিয়মে Profit বের করব।

Objects:	1	2	3	4	5	6	7
Profit:	12	5	16	7	9	11	6
Weight:	3	1	4	2	9	4	3
Max(P/w):	4	5	4	3.5	1	2.75	2

When Weight>Remaining then,  
 Weight= Remaining  
 Profit=Profit\*(Previous Remaining weight/Weight)

Objects	Profit	Weight	Remaining
2	5	1	15-1=14
1	12	3	14-3=11
3	16	4	11-4=7
4	7	2	7-2=5
6	11	4	5-4=1
7	2	1	1-1=0
Total	53Taka	15KG	0



## Practice

Object	1	2	3	4	5	6	7	Total Kg=15
Profit	10	5	15	7	6	18	3	Ans=55.33
Weight	2	3	5	7	1	4	1	

Object	1	2	3	4	5	6	7	Total Kg=15
Profit	5	10	15	7	8	4	9	Ans=51
Weight	1	3	5	4	1	3	2	

# Dijkstra algorithm:

**নিয়মঃ** Dijkstra algorithm এর ক্ষেত্রে সবগুলো Vertices visit করতে হবে। এটি Directed/undirected graph দুইটার জন্যই হতে পারে।

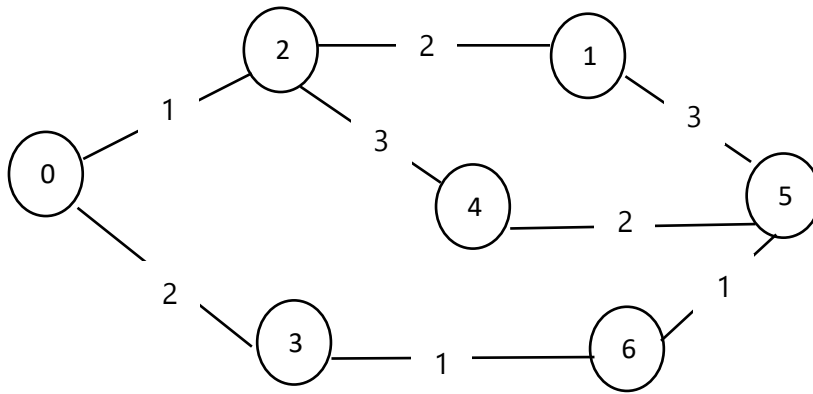
**শর্তঃ**

১। কোনো Vertices একবার visit করলে তা আর visit করা যাবে না।

২। বর্তমান distance পূর্বের distance এর ছোটো হলে বর্তমান distance নিতে হবে। এক্ষেত্রে  $If[d(u)+c(u,v) \leq d(v)]$  then  $d(v)=d(u)+c(u,v)$  সূত্র প্রয়োগ করে distance বের করবো।

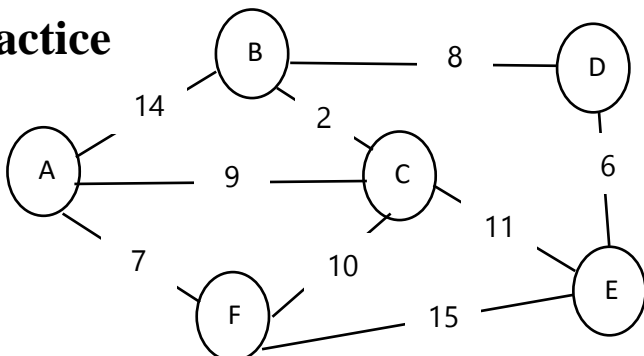
৩। Visit এর জন্য সবসময় সবচেয়ে ছোটো Distance কে select করতে হবে এক্ষেত্রে একের অধিক Vertices এর distance same হলে vertices এর ক্রমানুসারে যেতে হবে।

৪। Negative distance থাকতে পারবে না।



If $[d(u)+c(u,v) \leq d(v)]$ then $d(v)=d(u)+c(u,v)$ formula apply									
Visit	0	1	2	3	4	5	6		
0	0	$\infty$	$\infty$	$\infty$	$\infty$	$\infty$	$\infty$	$u=0, v=2$ $d(0)+c(0,2)=0+1=1 < \infty$ $d(2)=1$	$u=3, v=6$ $d(3)+c(3,6)=2+1=3 < \infty$ $d(6)=3$
2		$\infty$	1/0	2/0	$\infty$	$\infty$	$\infty$	$u=0, v=3$ $d(0)+c(0,3)=0+2=2 < \infty$ $d(3)=2$	$u=1, v=5$ $d(1)+c(1,5)=3+3=6 < \infty$ $d(5)=6$
3		3/2		2/0	4/2	$\infty$	$\infty$	$u=2, v=1$ $d(2)+c(2,1)=1+2=3 < \infty$ $d(1)=3$	$u=6, v=5$ $d(6)+c(6,5)=3+1=4 < 6$ $d(5)=4$
1		3/2			4/2	$\infty$	3/3	$u=2, v=4$ $d(2)+c(2,4)=1+3=4 < \infty$ $d(4)=4$	$u=4, v=5$ $d(4)+c(4,5)=4+2=6 > 4$
6					4/2	6/1	3/3		
4					4/2	4/6			
5						4/6			
Distance & Path									
Distance(1)=3 Path: 1->2->0					Distance(4)=4 Path: 4->2->0				
Distance(2)=1 Path: 2->0					Distance(5)=4 Path: 5->6->3->0				
Distance(3)=2 Path: 3->0					Distance(6)=3 Path: 6->3->0				

## Practice



Answer:

Distance(B)=11 Path B->C->A

Distance(C)=9 Path C->A

Distance(D)=19 Path D->B->C->A

Distance(E)=20 Path E->C->A

Distance(F)=7 Path F->A

# Bellman Ford

**নিয়মঃ** Bellman Ford এর ক্ষেত্রে iteration এর সংখ্যা হবে  $(n-1)$  সংখ্যক যেখানে  $n = \text{vertices}$  সংখ্যা।  
তবে যদি পরপর দুই বার iteration এর result same হয় তাহলে আর iteration করতে হবে না।

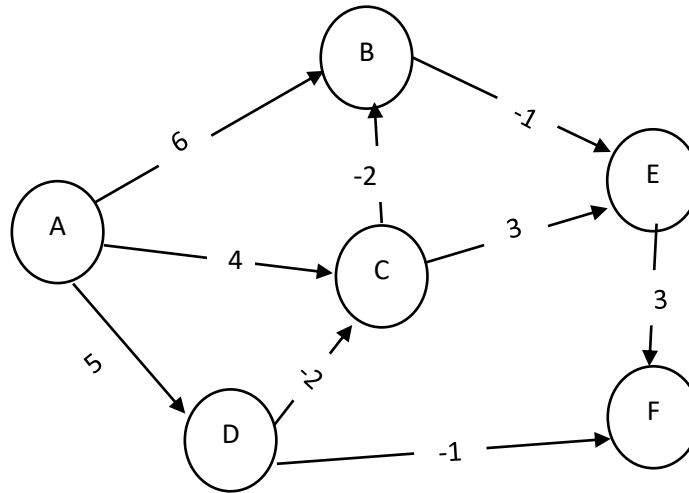
শর্তঃ

১। Graph এ কোনো loop থাকতে পারবে না।

২। If  $[d(u) + c(u,v) < d(v)]$  then  $d(v) = d(u) + c(u,v)$  এই সূত্র প্রয়োগ করে distance বের করবো।

৩। Directional graph হতে হবে ও negative cost থাকতে পারবে।

৪। প্রতিবার সব Vertices visit করতে হবে।



Here  $n=6$  so, iteration  $= (6-1)=5$

1 <sup>st</sup> iteration						
	A	B	C	D	E	F
A	0	$\infty$	$\infty$	$\infty$	$\infty$	$\infty$
B	0	6	4	5	$\infty$	$\infty$
C	0	6	4	5	5	$\infty$
D	0	2	4	5	5	$\infty$
E	0	2	3	5	5	4
F	0	2	3	5	5	4

2 <sup>nd</sup> iteration						
	A	B	C	D	E	F
A	0	2	3	5	5	4
B	0	2	3	5	5	4
C	0	2	3	5	1	4
D	0	1	3	5	1	4
E	0	1	3	5	1	4
F	0	1	3	5	1	4

3 <sup>rd</sup> iteration						
	A	B	C	D	E	F
A	0	1	3	5	1	4
B	0	1	3	5	1	4
C	0	1	3	5	0	4
D	0	1	3	5	0	4
E	0	1	3	5	0	4
F	0	1	3	5	0	3
	0	1	3	5	0	3

$u=A, v=B$   
 $d(A)+c(A,B)=0+6=6>1$   $d(B)=1$   
 $u=A, v=C$   
 $d(A)+c(A,C)=0+4=4>3$   $d(C)=3$   
 $u=A, v=D$   
 $d(A)+d(A,D)=0+5=5=5$   $d(D)=5$   
 $u=B, v=E$   
 $d(B)+c(B,E)=1+(-1)=0<1$   $d(E)=0$   
 $u=C, v=B$   
 $d(C)+c(C,B)=3+(-2)=1=1$   $d(B)=1$

$u=C, v=E$   
 $d(C)+c(C,E)=3+3=6>0$   $d(E)=0$   
 $u=D, v=C$   
 $d(D)+c(D,C)=5+(-2)=3=3$   $d(C)=3$   
 $u=D, v=F$   
 $d(D)+c(D,F)=5+(-1)=4=4$   $d(F)=4$   
 $u=E, v=F$   
 $d(E)+c(E,F)=0+3=3<4$   $d(F)=3$

4 <sup>th</sup> iteration						
	A	B	C	D	E	F
A	0	1	3	5	0	3
B	0	1	3	5	0	3
C	0	1	3	5	0	3
D	0	1	3	5	0	3
E	0	1	3	5	0	3
F	0	1	3	5	0	3
	0	1	3	5	0	3

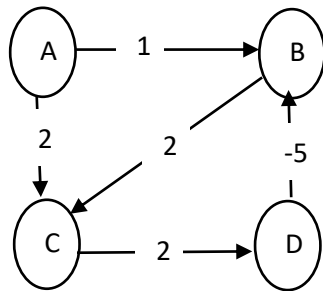
$u=A, v=B$   
 $d(A)+c(A,B)=0+6=6>1$   $d(B)=1$   
 $u=A, v=C$   
 $d(A)+c(A,C)=0+4=4>3$   $d(C)=3$   
 $u=A, v=D$   
 $d(A)+d(A,D)=0+5=5=5$   $d(D)=5$   
 $u=B, v=E$   
 $d(B)+c(B,E)=1+(-1)=0=0$   $d(E)=0$   
 $u=C, v=B$   
 $d(C)+c(C,B)=3+(-2)=1=1$   $d(B)=1$

$u=C, v=E$   
 $d(C)+c(C,E)=3+3=6>0$   $d(E)=0$   
 $u=D, v=C$   
 $d(D)+c(D,C)=5+(-2)=3=3$   $d(C)=3$   
 $u=D, v=F$   
 $d(D)+c(D,F)=5+(-1)=4>3$   $d(F)=3$   
 $u=E, v=F$   
 $d(E)+c(E,F)=0+3=3=3$   $d(F)=3$

এখানে যেহেতু 3<sup>rd</sup> & 4<sup>th</sup> iteration same তাই আর 5<sup>th</sup> iteratin করা লাগবে না।

Distance(A)=0 Distance(B)=1 Distance(C)=3	Distance(D)=5 Distance(E)=0 Distance(F)=3
---	---

**Drawback:** কেনো হতে Loop পারবে না।



Here  $n=4$  so, iteration  $=(4-1)=3$

1 <sup>st</sup> iteration				
	A	B	C	D
A	0	$\infty$	$\infty$	$\infty$
B	0	1	2	$\infty$
C	0	1	2	$\infty$
D	0	1	2	4
	0	-1	2	5

$u=A, v=B$   
 $d(A)+c(A,B)=0+1=1<\infty$   $d(B)=1$   
 $u=A, v=C$   
 $d(A)+c(A,C)=0+2=2<\infty$   $d(C)=2$   
 $u=B, v=C$   
 $d(B)+c(B,C)=1+2=3>2$   $d(C)=2$   
 $u=C, v=D$   
 $d(C)+c(C,D)=2+2=4<\infty$   $d(D)=4$   
 $u=D, v=B$   
 $d(D)+c(D,B)=4+(-5)=-1<1$   $d(B)=-1$

2 <sup>nd</sup> iteration				
	A	B	C	D
A	0	-1	2	5
B	0	-1	2	5
C	0	-1	1	5
D	0	-1	1	3
	0	-2	1	3

$u=A, v=B$   
 $d(A)+c(A,B)=0+1=1>-1$   $d(B)=-1$   
 $u=A, v=C$   
 $d(A)+c(A,C)=0+2=2=2$   $d(C)=2$   
 $u=B, v=D$   
 $d(B)+c(B,C)=-1+2=1<2$   $d(C)=1$   
 $u=C, v=D$   
 $d(C)+c(C,D)=1+2=3<5$   $d(D)=3$   
 $u=D, v=B$   
 $d(D)+c(D,B)=3+(-5)=-2<0$   $d(B)=-2$

3 <sup>rd</sup> iteration				
	A	B	C	D
A	0	-2	1	3
B	0	-2	1	3
C	0	-2	0	3
D	0	-2	0	2
	0	-3	0	2

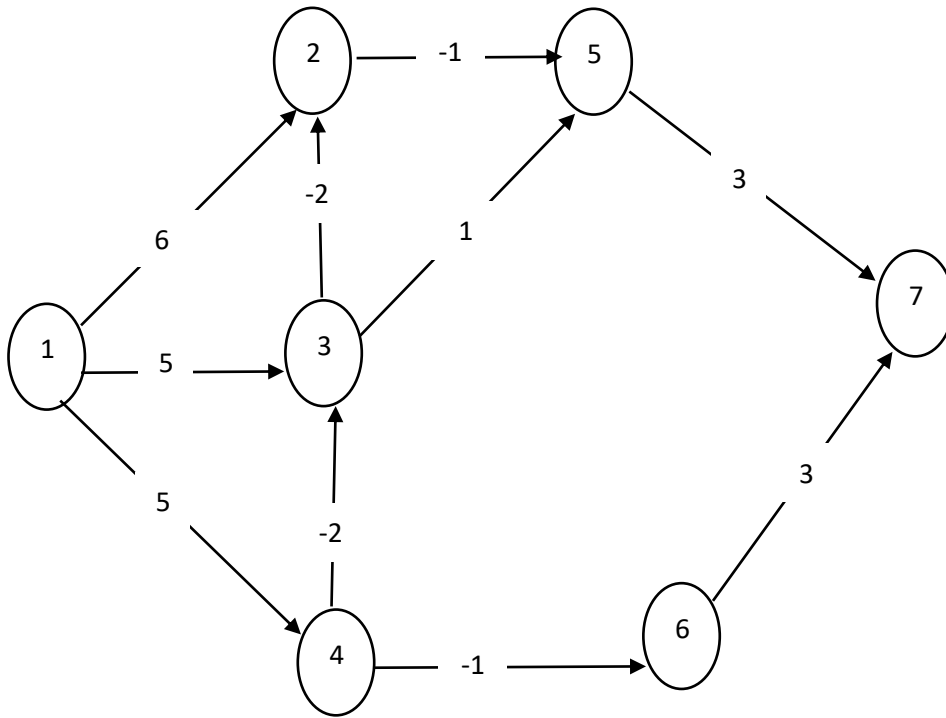
$u=A, v=B$   
 $d(A)+c(A,B)=0+1=1>-2$   $d(B)=-2$   
 $u=A, v=C$   
 $d(A)+c(A,C)=0+4=4>1$   $d(C)=1$   
 $u=B, v=D$   
 $d(B)+c(B,C)=-2+2=0<1$   $d(C)=0$   
 $u=C, v=D$   
 $d(C)+c(C,D)=0+2=2<3$   $d(D)=2$   
 $u=D, v=B$   
 $d(D)+c(D,B)=2+(-5)=-3<-2$   $d(B)=-3$



4 <sup>th</sup> iteration					
	A	B	C	D	u=A,v=B d(A)+c(A,B)=0+1=1>-3 d(B)=-3 u=A,v=C d(A)+c(A,C)=0+4=4>0 d(C)=0 u=B,v=D d(B)+c(B,C)=-3+2=-1 d(C)=-1 u=C,v=D d(C)+c(C,D)=-1+2=1<2 d(D)=1 u=D,v=B d(D)+c(D,B)=1+(-5)=-4<-3 d(B)=-4
A	0	-3	0	2	
B	0	-3	0	2	
C	0	-3	-1	2	
D	0	-3	-1	1	
	0	-4	-1	1	

এখানে দেখা যাচ্ছে 3<sup>rd</sup> iteration এর পরও value change হচ্ছে কিন্তু নিয়ম অনুসারে 3<sup>rd</sup> iteration এর পর value change হতে পারবে না তাই graph এ loop থাকলে Bellman Ford use করা যায় না

## Practice:



## Floyd warshall algorithm

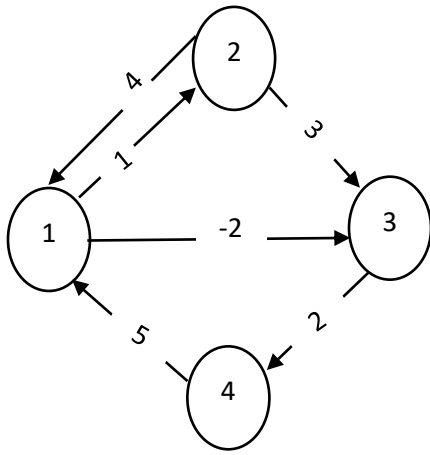
**নিয়মঃ** Floyd warshall algorithm positive & negative eadge দুইটার জন্যই কাজ করে কিন্তু কোনো negative cycle এর ক্ষেত্রে Floyd warshall algorithm use করা যায় না। এটি directed & undirected graph দুইটার জন্যই কাজ করে।

সর্তঃ

১। প্রতিবার একটি vertices দিয়ে সবগুলো vertices visit করতে হবে।

২। শুরুতে একটি distance matrix তৈরি করতে হবে।

৩। এরপর যতগুলো vertices আছে ততগুলো distance matrix তৈরি করতে হবে এবং যে vertices নিয়ে কাজ করবো শুরুতে তার row, column & diagonal value নিয়ে নিতে হবে। এক্ষেত্রে এই সূত্র  $D^k[i,j]=\min\{D^{k-1}[i,j],D^{k-1}[i,k]+D^{k-1}[k,j]\}$  ব্যবহার করতে হবে।



D0	1	2	3	4
1	0	1	-2	$\infty$
2	4	0	3	$\infty$
3	$\infty$	$\infty$	0	2
4	5	$\infty$	$\infty$	0

D1	1	2	3	4	$D^1[2,3]=\min\{D^0[2,3], D^0[2,1]+D^0[1,3]\}$ $=\min\{3, 4+(-2)\}=\min\{3, 1\}=2$ $D^1[2,4]=\min\{D^0[2,4], D^0[2,1]+D^0[1,4]\}$ $=\min\{\infty, 4+\infty\}=\min\{\infty, \infty\}=\infty$ $D^1[3,2]=\min\{D^0[3,2], D^0[3,1]+D^0[1,2]\}$ $=\min\{\infty, \infty+1\}=\min\{\infty, \infty\}=\infty$	$D^1[3,4]=\min\{D^0[3,4], D^0[3,1]+D^0[1,4]\}$ $=\min\{2, \infty+\infty\}=\min\{2, \infty\}=2$ $D^1[4,2]=\min\{D^0[4,2], D^0[4,1]+D^0[1,2]\}$ $=\min\{\infty, 5+1\}=\min\{\infty, 6\}=6$ $D^1[4,3]=\min\{D^0[4,3], D^0[4,1]+D^0[1,3]\}$ $=\min\{\infty, 5+(-2)\}=\min\{\infty, 3\}=3$
1	0	1	-2	$\infty$		
2	4	0	2	$\infty$		
3	$\infty$	$\infty$	0	2		
4	5	6	3	0		

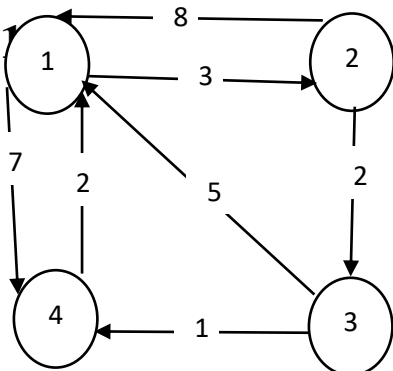
D2	1	2	3	4	$D^2[1,3]=\min\{D^1[1,3], D^1[1,2]+D^1[2,3]\}$ $=\min\{-2, 1+2\}=\min\{-2, 3\}=-2$ $D^2[1,4]=\min\{D^1[1,4], D^1[1,2]+D^1[2,4]\}$ $=\min\{\infty, 1+\infty\}=\min\{\infty, \infty\}=\infty$ $D^2[3,1]=\min\{D^1[3,1], D^1[3,2]+D^1[2,1]\}$ $=\min\{\infty, \infty+4\}=\min\{\infty, \infty\}=\infty$	$D^2[3,4]=\min\{D^1[3,4], D^1[3,2]+D^1[2,4]\}$ $=\min\{2, \infty+\infty\}=\min\{2, \infty\}=2$ $D^2[4,1]=\min\{D^1[4,1], D^1[4,2]+D^1[2,1]\}$ $=\min\{5, 6+4\}=\min\{5, 10\}=5$ $D^2[4,3]=\min\{D^1[4,3], D^1[4,2]+D^1[2,3]\}$ $=\min\{3, 6+2\}=\min\{3, 8\}=3$
1	0	1	-2	$\infty$		
2	4	0	2	$\infty$		
3	$\infty$	$\infty$	0	2		
4	5	6	3	0		

D2	1	2	3	4	$D^3[1,2]=\min\{D^2[1,2], D^2[1,3]+D^2[3,2]\}$ $=\min\{1, -2+\infty\}=\min\{1, \infty\}=1$ $D^3[1,4]=\min\{D^2[1,4], D^2[1,3]+D^2[3,4]\}$ $=\min\{\infty, -2+2\}=\min\{\infty, 0\}=0$ $D^3[2,1]=\min\{D^2[2,1], D^2[2,3]+D^2[3,1]\}$ $=\min\{4, 2+\infty\}=\min\{4, \infty\}=4$	$D^3[2,4]=\min\{D^2[2,4], D^2[2,3]+D^2[3,4]\}$ $=\min\{\infty, 2+2\}=\min\{\infty, 4\}=4$ $D^3[4,1]=\min\{D^2[4,1], D^2[4,3]+D^2[3,1]\}$ $=\min\{5, 3+\infty\}=\min\{5, \infty\}=5$ $D^3[4,2]=\min\{D^2[4,2], D^2[4,3]+D^2[3,2]\}$ $=\min\{6, 3+\infty\}=\min\{6, \infty\}=6$
1	0	1	-2	0		
2	4	0	2	4		
3	$\infty$	$\infty$	0	2		
4	5	6	3	0		

D2	1	2	3	4	$D^4[1,2]=\min\{D^3[1,2], D^3[1,4]+D^3[4,2]\}$ $=\min\{1, 0+6\}=\min\{1, 6\}=1$ $D^4[1,3]=\min\{D^3[1,3], D^3[1,4]+D^3[4,3]\}$ $=\min\{-2, 0+0\}=\min\{-2, 0\}=-2$ $D^4[2,1]=\min\{D^3[2,1], D^3[2,4]+D^3[4,1]\}$ $=\min\{4, 4+5\}=\min\{4, 9\}=4$	$D^4[2,3]=\min\{D^3[2,3], D^3[2,4]+D^3[4,3]\}$ $=\min\{2, 4+3\}=\min\{2, 7\}=2$ $D^4[3,1]=\min\{D^3[3,1], D^3[3,4]+D^3[4,1]\}$ $=\min\{\infty, 2+5\}=\min\{\infty, 7\}=7$ $D^4[3,2]=\min\{D^3[3,2], D^3[3,4]+D^3[4,2]\}$ $=\min\{\infty, 2+6\}=\min\{\infty, 8\}=8$
1	0	1	-2	0		
2	4	0	2	4		
3	7	8	0	2		
4	5	6	3	0		

Which is required answer of all pair shortest path.

### Practice:



Answer				
D4	1	2	3	4
1	0	3	5	6
2	5	0	2	3
3	3	6	0	1
4	2	5	7	0

# Longest common subsequence

**নিয়মঃ** Longest common subsequence এর ক্ষেত্রে প্রথমে X এবং Y কে একটি টেবিলে সারি (শুরুতে 0) এবং কলামে (শুরুতে 0) লিখতে হবে এবং প্রথম সারি ও কলাম 0 হবে। এরপর যদি এর মান কোনাকুনি মিলে যায় তাহলে তাদের কর্নার এর মানের সাথে ১ যোগ করে লিখতে হবে আর যদি না মিলে তাহলে তাদের সামনের/উপরের মানের মধ্যে যেটি বড় সেটি লিখতে হবে।

P,Q same conner rule		
	0	P
0	0	0
P	0	0+1=1

P,Q not same conner rule		
	0	P
0	0	0
Q	0	0

X= P R E S I D E N T

Y= P R O V I D E N C E

	0	P	R	O	V	I	D	E	N	C	E	Check
0	0	0	0	0	0	0	0	0	0	0	0	PRIDEN=6
P	0	1	1	1	1	1	1	1	1	1	1	
R	0	1	2	2	2	2	2	2	2	2	2	
E	0	1	2	2	2	2	2	3	3	3	3	
S	0	1	2	2	2	2	2	3	3	3	3	
I	0	1	2	2	2	3	3	3	3	3	3	
D	0	1	2	2	2	3	4	4	4	4	4	
E	0	1	2	2	2	3	4	5	5	5	5	
N	0	1	2	2	2	3	4	5	6	6	6	
T	0	1	2	2	2	3	4	5	6	6	6	

Short Technique			
X= P R E S I D E N T	6	X= P R E S I D E N T	3
Y= P R O V I D E N C E		Y= P R O V I D E N C E	
X= P R E S I D E N T	5	X= P R E S I D E N T	2
Y= P R O V I D E N C E		Y= P R O V I D E N C E	
X= P R E S I D E N T	4	X= P R E S I D E N T	1
Y= P R O V I D E N C E		Y= P R O V I D E N C E	
X= P R E S I D E N T	4	X= P R E S I D E N T	1
Y= P R O V I D E N C E		Y= P R O V I D E N C E	
X= P R E S I D E N T	4	X= P R E S I D E N T	1
Y= P R O V I D E N C E		Y= P R O V I D E N C E	

## Practice

X=	B D C A B A	Ans=4
Y=	A B C B D A B	

# 0/1 Knapsack

**নিয়মঃ** 0/1 Knapsack এর এক্ষেত্রে কোনো একটি object নিলে ওইটা হয় পুরোটা নিতে হবে নাহয় নিতে পারব না। কিন্তু কোনো Fraction নেওয়া যাবে না। এক্ষেত্রে একটি টেবিল তৈরি করতে হবে যার row বরাবর weight এবং coloumn বরাবর item লিখতে হবে এবং শুরুতে এর মান করে দিতে হবে। এরপর যে item নিয়ে কাজ করব তার weight এর আগ পর্যন্ত সব value লিখে ফেলতে হবে এবং বাকি value গুলো তার আগের row এর প্রথম value থেকে ক্রমানুসারে ঐ item এর profit এর সাথে যোগ করে লিখতে হবে। তবে আগের value বড় হলে সেটা নিতে হবে।

অথবা,  $B[i,w]=\max(B[i-1,w],B[i-1,w-w[i]+v[i])$  এই rule দিয়েও করা যাবে।

Items/objects: 1   2   3   4                      max weight=5  
 Weight:            3   2   5   4                      n=4  
 Value/Profit:    4   3   6   5

	0	1	2	3	4	5	$B[4,4]=\max(B[4-1,4],B[4-1,4-w[4]+v[4])$ $=\max(B[3,4],B[3,4-4]+5)$ $=\max(B[3,4],B[3,0]+5)$ $=\max(4,0+5)=\max(4,5)=5$
0	0	0	0	0	0	0	
1	0	0	0	4	4	4	
2	0	0	3	4	4	7	$B[4,5]=\max(B[4-1,5],B[4-1,5-w[4]+v[4])$ $=\max(B[3,5],B[3,5-4]+5)$ $=\max(B[3,5],B[3,1]+5)$ $=\max(7,0+5)=\max(7,5)=7$
3	0	0	3	4	4	7	
4	0	0	3	4	4	7	
Profit=4+7=7							
Item: 1   2   3   4	7-3=4						
1   1   0   0	4-4=0						

## Practice:

Items/objects: 1   2   3   4                      max weight=8	Ans=8
Weight:            2   3   4   5                      n=4	
Value/Profit:    1   2   4   5	