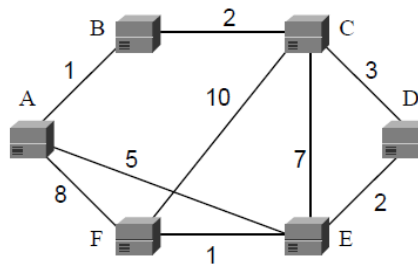


5 Strato di rete (Parte 2)

Esercizio 5.31

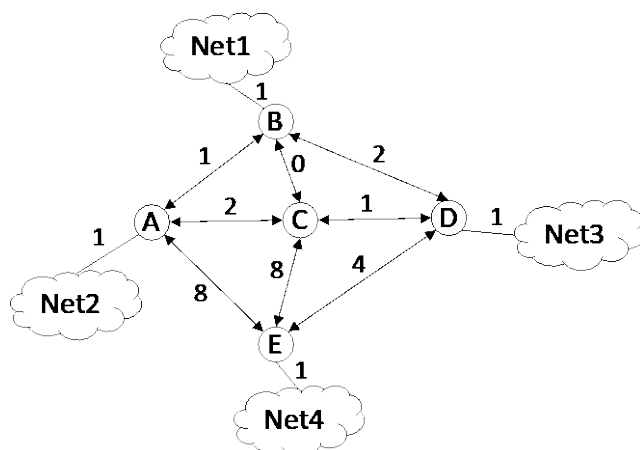
Costruire, utilizzando l'algoritmo di *Bellman-Ford*, la tabella di instradamento del nodo F della rete seguente a 6 nodi, che indica le distanze tra nodi mutuamente connessi, per i primi quattro passi di aggiornamento.



Nodo F – Step 1			Nodo F – Step 2			Nodo F – Step 3			Nodo F – Step 4		
Dest	Cost	Next hop	Dest	Cost	Next hop	Dest	Cost	Next hop	Dest	Cost	Next hop
A	8	A	A	6	E	A	6	E			
B	INF	–	B	9	A	B	7	E			
C	10	C	C	8	E	C	6	E			
D	INF	–	D	3	E	D	3	E			
E	1	E	E	1	E	E	1	E			
F	0	–	F	0	–	F	0	–			

<i>Nodo F – Step 1</i>			<i>Nodo F – Step 2</i>			<i>Nodo F – Step 3</i>			<i>Nodo F – Step 4</i>		
<i>Dest</i>	<i>Cost</i>	<i>Next hop</i>	<i>Dest</i>	<i>Cost</i>	<i>Next hop</i>	<i>Dest</i>	<i>Cost</i>	<i>Next hop</i>	<i>Dest</i>	<i>Cost</i>	<i>Next hop</i>
<i>A</i>	<i>8</i>	<i>A</i>	<i>A</i>	<i>6</i>	<i>E</i>	<i>A</i>	<i>6</i>	<i>E</i>	<i>A</i>	<i>6</i>	<i>E</i>
<i>B</i>	<i>inf</i>	<i>-</i>	<i>B</i>	<i>9</i>	<i>A</i>	<i>B</i>	<i>7</i>	<i>E</i>	<i>B</i>	<i>7</i>	<i>E</i>
<i>C</i>	<i>10</i>	<i>C</i>	<i>C</i>	<i>8</i>	<i>E</i>	<i>C</i>	<i>6</i>	<i>E</i>	<i>C</i>	<i>6</i>	<i>E</i>
<i>D</i>	<i>inf</i>	<i>-</i>	<i>D</i>	<i>3</i>	<i>E</i>	<i>D</i>	<i>3</i>	<i>E</i>	<i>D</i>	<i>3</i>	<i>E</i>
<i>E</i>	<i>1</i>	<i>E</i>	<i>E</i>	<i>1</i>	<i>E</i>	<i>E</i>	<i>1</i>	<i>-</i>	<i>E</i>	<i>1</i>	<i>-</i>
<i>F</i>	<i>0</i>	<i>-</i>	<i>F</i>	<i>0</i>	<i>-</i>	<i>F</i>	<i>0</i>	<i>-</i>	<i>F</i>	<i>0</i>	<i>-</i>

Esercizio 5.32 (E)



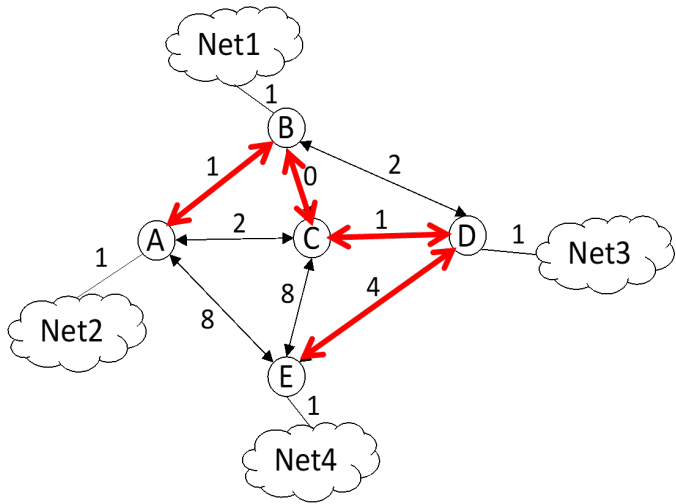
Nella rete in figura è rappresentato il grafo di una rete in cui sono presenti dei *router* (A, B, C, D, E) e 4 reti (Net1, Net2, Net3, Net4). I costi di attraversamento sono indicati accanto ad ogni *link*, i *link* sono bidirezionali e simmetrici. Si chiede di:

- Calcolare mediante l'algoritmo di *Bellman-Ford* l'albero dei cammini minimi con sorgente A e destinazioni tutti gli altri *router* (si omettano le reti nel grafo). Indicare:
 - nella tabella sottostante, la tabella di routing del nodo A ad ogni step in cui il nodo viene analizzato;
 - nella figura sopra, l'albero trovato
- Sulla base dell'albero dei cammini calcolato al punto precedente, indicare i *Distance vector* (DV) relativi alle reti Net1, Net2, Net3 e Net4, inviati dal *router* A ai propri vicini nella modalità *Split Horizon* con *Poisonous Reverse*. Per ogni DV inviato indicare chiaramente il destinatario del DV e le reti raggiungibili comunicate con i rispettivi costi.

Nodo A – Step 1			Nodo A – Step 2			Nodo A – Step 3			Nodo A – Step 4		
Dest	Cost	Next hop	Dest	Cost	Next hop	Dest	Cost	Next hop	Dest	Cost	Next hop

a)

Nodo A – Step 1			Nodo A – Step 2			Nodo A – Step 3			Nodo A – Step 4		
Dest	Cost	Next hop	Dest	Cost	Next hop	Dest	Cost	Next hop	Dest	Cost	Next hop
A	0	-	A	0	-	A	0	-	A	0	-
B	1	B	B	1	B	B	1	B	B	1	B
C	2	C	C	1	B	C	1	B	C	1	B
D	inf	-	D	3	B	D	2	B	D	2	B
E	8	E	E	8	E	E	7	B	E	6	B

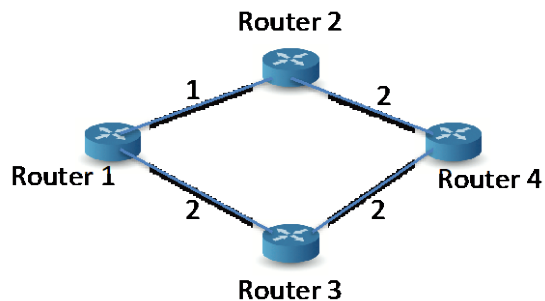


b)

	A=>B	A=>C	A=>E
Net 1	inf	2	2
Net 2	1	1	1
Net 3	inf	3	3
Net 4	inf	7	7

Esercizio 5.33

Scrivere il contenuto di tutti i pacchetti di *distance vector* inviati dal Router 1 agli altri router nei due casi: (i) *distance vector* base, (ii) *distance vector* con *Split Horizon* con *Poisonous Reverse* e Hop-limit=16. La figura riporta la topologia di rete e la tabella di *routing* del router 1.



Destinazione	Next Hop	Costo
Net 1	Router 2	4
Net 2	Router 3	3
Net 3	Router 2	2
Net 4	Router 3	3
Net 5	Local	Local

Distance vector base:

Il router 1 invia lo stesso messaggio a tutti i router collegati:

Net 1:4, Net2:3, Net3:2, Net4:3, Net5:local

Distance vector con *Split Horizon*

Il router 1 invia messaggi diversi a Router 2 e router 3. Le destinazioni che il router 1 raggiunge usando come next hop il destinatario del DV sono indicate con costo uguale a *hop limit* nei *distance vector* inviati.

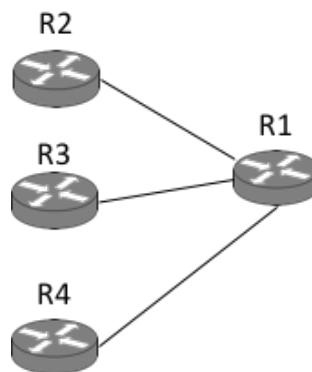
A router 2: Net 1: 16, Net 2: 3, Net 3: 16, Net 4: 3, Net 5: local

A router 3: Net 1: 4, Net 2: 16, Net 3: 2, Net 4: 16, Net 5: local

Esercizio 5.34 (I)

La figura riporta la topologia di rete e la tabella di *routing* di R1. Scrivere il contenuto di tutti i pacchetti di *distance vector* inviati dal router R1 agli altri router nei due casi: (a) *distance vector* base, (b) *distance vector* con *Split Horizon* con *Poisonous Reverse* e Hop-limit=16.

Network	Int	Next-hop
1.1.1.0/24	1	R4
1.1.2.0/24	3	R2
1.1.3.0/24	2	R2
1.1.4.0/24	4	R3
1.1.5.0/24	3	R3
0.0.0.0	3	R4



(a)

verso tutti

1.1.1.0/24 1
1.1.2.0/24 3
1.1.3.0/24 2
1.1.4.0/24 4
1.1.5.0/24 3

(b)

verso R2

1.1.1.0/24 1
1.1.2.0/24 16
1.1.3.0/24 16
1.1.4.0/24 4
1.1.5.0/24 3

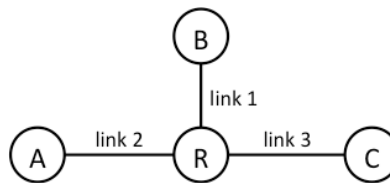
verso R3

1.1.1.0/24 1
1.1.2.0/24 3
1.1.3.0/24 2
1.1.4.0/24 16
1.1.5.0/24 16

verso R4

1.1.1.0/24 16
1.1.2.0/24 3
1.1.3.0/24 2
1.1.4.0/24 4
1.1.5.0/24 3

Esercizio 5.35



In figura sono rappresentati i router A, B, C e R, i costi di attraversamento di ogni *link* sono tutti uguali e pari a 2. Le informazioni di raggiungibilità del router R sono le seguenti:

Destination	Cost	Next Hop
Net A	8	B
Net B	6	B
Net C	5	A
Net D	2	C

Il router R riceve dal *link* 1 il seguente DV: (NetA,4), (NetB,7), (NetC,4) e successivamente dal *link* 2 il seguente DV: (NetB,2), (NetC,3), (NetE,7).

Si indichino:

- Le informazioni di raggiungibilità di R dopo il primo DV
- Le informazioni di raggiungibilità di R dopo il secondo DV
- A valle del secondo DV, il DV inviato da R nella modalità *Split Horizon* con *Poisonous Reverse*.

Dopo primo DV

Net	NH	Cost
NetA	B	6
NetB	B	9
NetC	A	5
NetD	C	2

Dopo secondo DV

Net	NH	Cost
NetA	B	6
NetB	A	4
NetC	A	5
NetD	C	2
NetE	A	9

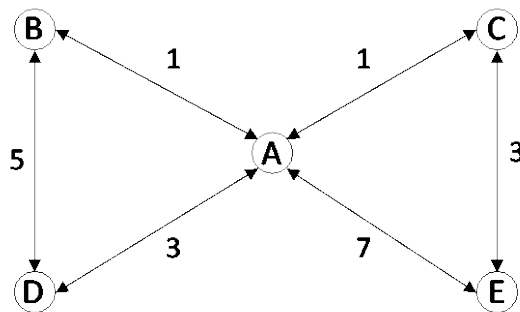
DV inviato ad A: (NetA,6),(NetB,inf),(NetC,inf),(NetD,2),(NetE,inf)

DV inviato ad B: (NetA,inf),(NetB,4),(NetC,5),(NetD,2),(NetE,9)

DV inviato ad C: (NetA,6),(NetB,4),(NetC,5),(NetD,inf),(NetE,9)

Esercizio 5.36 (E)

Sia data la rete in figura.



Sono indicati i nodi (A,B,C,D,E) ed il costo di attraversamento di ogni collegamento. Nella rete è attivo un algoritmo di *routing* secondo cammini minimi arrivato a convergenza. Ipotizzando che gli stessi nodi siano le destinazioni da raggiungere, si chiede di:

- Indicare i *Distance vector* (no *Split Horizon*) inviati dal nodo A (attenzione: contenuto e destinatario del DV)
- Indicare i *Distance vector* inviati dal nodo A in caso di *Split Horizon*, senza *Poisonous Reverse* (attenzione: contenuto e destinatario del DV)
- Nel caso in cui il nodo A riceva dal nodo C il seguente DV: (B,2), (D,1), (E,5), (F,3), riempire le tabelle di instradamento del nodo A sottostanti, una è prima della ricezione del DV, l'altra subito dopo.

PRIMA		
Dest.	Costo	Next-Hop

DOPO		
Dest.	Costo	Next-Hop

a)

Tutti i DV verso B, C, D, E sono identici e sono del tipo: (A,0); (B,1); (C,1); (D,3); (E,4)

b)

DV verso B: A0; C1; D3; E4

DV verso C: A0; B1; D3

DV verso D: A0; B1; C1; E4

DV verso E: A0; B1; C1; D3; E4

c)

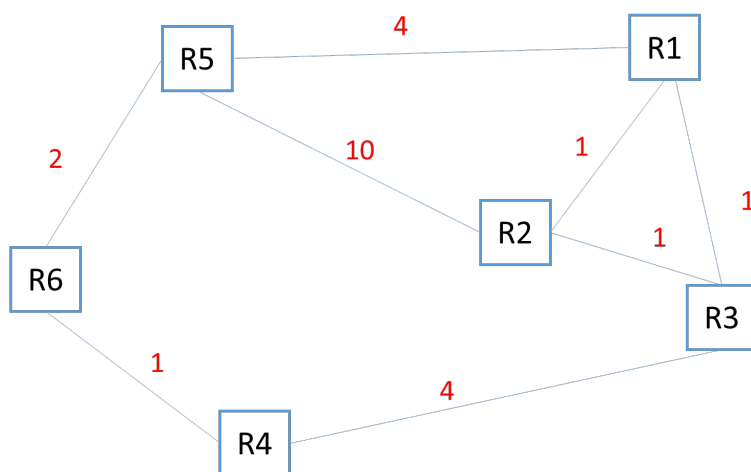
A riceve dal nodo C il seguente DV: (B,2), (D,1), (E,5), (F,3).

PRIMA		
Dest.	Costo	Next-Hop
A	0	Dir
B	1	B
C	1	C
D	3	D
E	4	C

DOPO		
Dest.	Costo	Next-Hop
A	0	dir
B	1	B
C	1	C
D	2	C
E	6	C
F	4	C

Esercizio 5.37

Si consideri il grafo in figura, che rappresenta una rete costituita da 6 router ed i costi dei relativi collegamenti. Si calcolino su tale grafo i cammini minimi da R1 a tutti gli altri router usando l'algoritmo di *Dijkstra*.

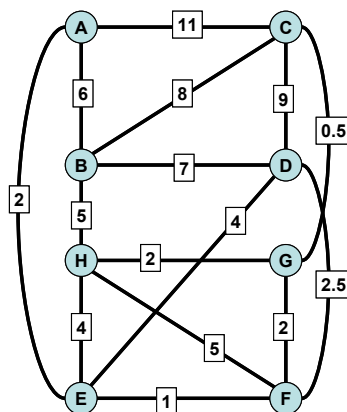


Cammini minimi da R1

Passo	M=Nodi nel MST	R1	R2	R3	R4	R5	R6
0	R1	(0, -)	(1, R1)	(1, R1)	(inf, -)	(4, R1)	(inf, -)
1	R1, R2			(1, R1)	(inf, -)	(4, R1)	(inf, -)
2	R1, R2, R3				(5, R3)	(4, R1)	(inf, -)
3	R1, R2, R3, R5				(5, R3)		(6, R5)
4	R1, R2, R3, R5, R4						(6, R4)
5	R1, R2, R3, R5, R4, R6						

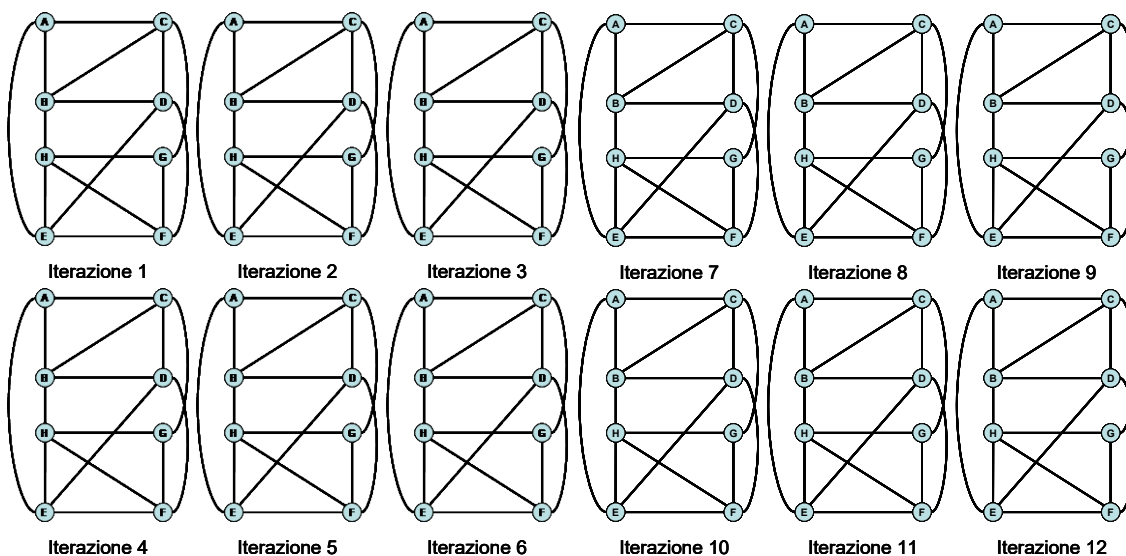
Esercizio 5.38

Data la rete rappresentata in figura (in cui su ogni link è riportato il costo) si trovi l'albero dei cammini minimi del nodo B mediante l'algoritmo di *Dijkstra*. Utilizzando i grafi seguenti, ad ogni passo si segnino sul grafo relativo il ramo ed il nodo aggiunti. Inoltre si riporti nella tabella seguente ad ogni passo e per ogni nodo x il vettore: (D_x, p_x) , dove p_x è il nodo precedente di x nel percorso e D_x è la distanza al passo corrente del nodo x dal nodo radice.



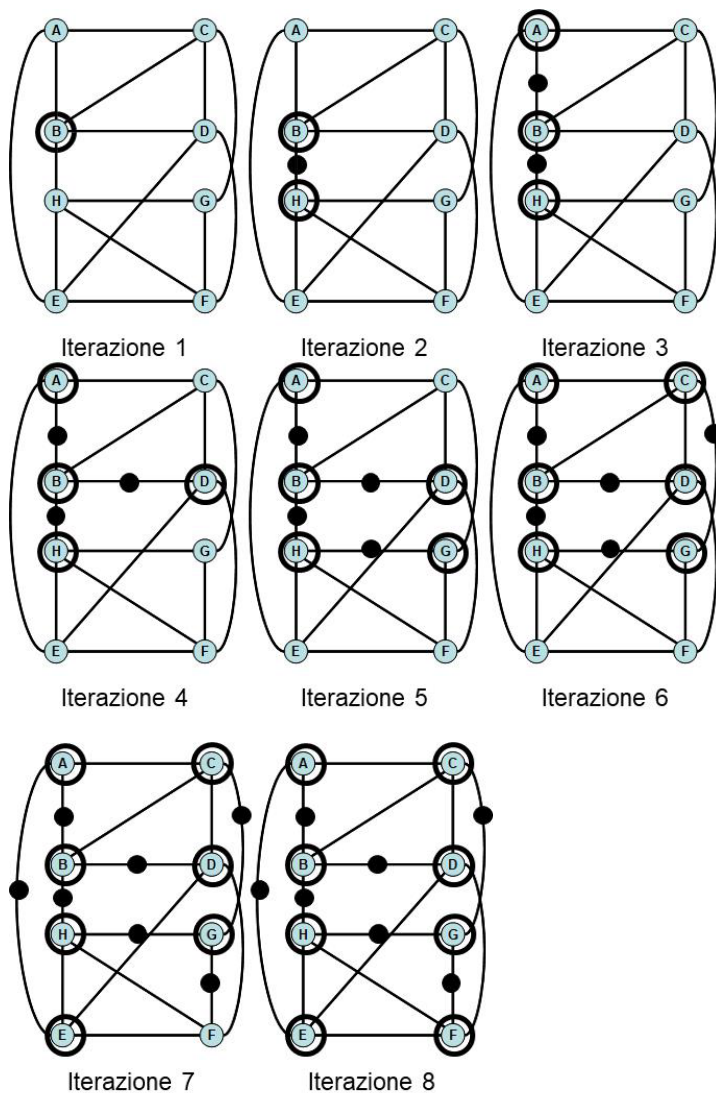
Passo	M	A		C		D		E		F		G		H	
		D_A	p_A	D_C	p_C	D_D	p_D	D_E	p_E	D_F	p_F	D_G	p_G	D_H	p_H
0															
1															
2															
3															
4															
5															
6															
7															

Nota: in caso di confronto tra nodi di uguale costo, scegliere secondo l'ordine alfabetico



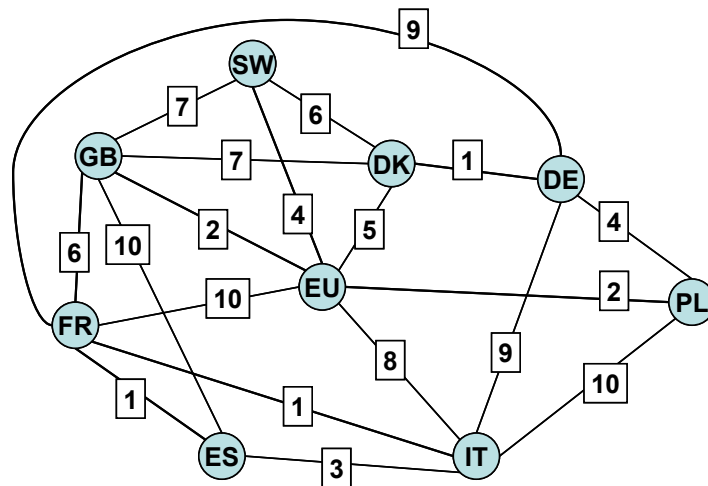
Passo	M	A		C		D		E		F		G		H	
		D_A	p_A	D_C	p_C	D_D	p_D	D_E	p_E	D_F	p_F	D_G	p_G	D_H	p_H
0	B	6	B	8	B	7	B							5	B
1	B,H	6	B	8	B	7	B	9	H	10	H	7	H		
2	A,B,H			8	B	7	B	8	A	10	H	7	H		
3	A,B,D,H			8	B			8	A	9.5	D	7	H		
4	A,B,D,G,H			7.5	G			8	A	9	G				
5	A,B,C,D,G,H							8	A	9	G				
6	A,B,C,D,E,G,H									9	G*				
7	A,B,C,D,E,F,G,H														

* Si poteva alternativamente considerare E come predecessore



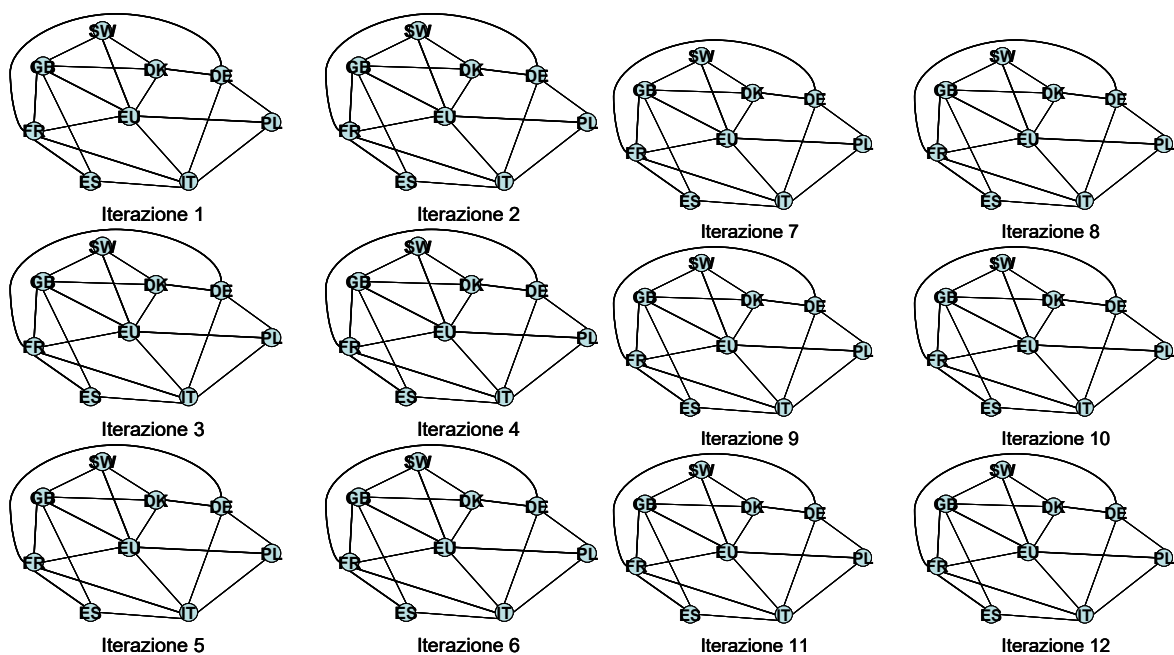
Esercizio 5.39

Data la rete rappresentata in figura si trovi l'albero dei cammini minimi del nodo ES mediante l'algoritmo di *Dijkstra*. Utilizzando i grafi seguenti, ad ogni passo si segnino sul grafo relativo il ramo ed il nodo aggiunti. Inoltre si riporti nella tabella seguente ad ogni passo e per ogni nodo x il vettore: (D_x, p_x) , dove p_x è il nodo precedente di x nel percorsi.



Passo	M	FR		GB		SW		EU		DK		DE		PL		IT	
		D_{FR}	p_{FR}	D_{GB}	p_{GB}	D_{SW}	p_{SW}	C	p_{EU}	D_{DK}	p_{DK}	D_{DE}	p_{DE}	D_{PL}	p_{PL}	D_{IT}	p_{IT}
0																	
1																	
2																	
3																	
4																	
5																	
6																	
7																	
8																	

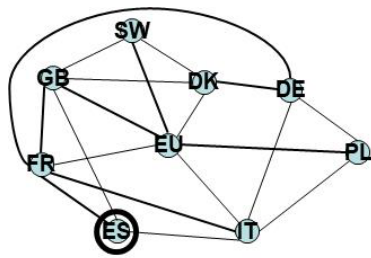
Nota: in caso di confronto tra nodi di uguale costo, scegliere secondo l'ordine alfabetico



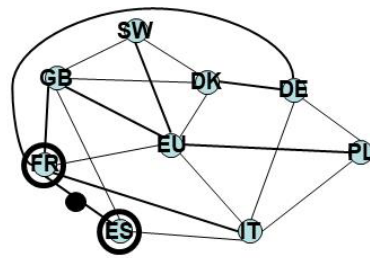
Passo	M	FR		GB		SW		EU		DK		DE		PL		IT	
		D _{FR}	P _{FR}	D _{GB}	P _{GB}	D _{SW}	P _{SW}	C	P _{EU}	D _{DK}	P _{DK}	D _{DE}	P _{DE}	D _{PL}	P _{PL}	D _{IT}	P _{IT}
0	ES	1	ES	10	ES											3	ES
1	ES,FR			7	FR			11	FR			10	FR			2	FR
2	ES,FR,IT			7	FR			10	IT			10	FR	12	IT		
3	ES,FR,IT,GB					14	GB	9	GB	14	GB	10	FR	12	IT		
4	ES,FR,IT,GB,EU					13	EU			14	EU*	10	FR	11	EU		
5	ES,FR,IT,GB,EU, DE					13	EU			11	DE		11	EU			
6	ES,FR,IT,GB,EU, DE,DK					13	EU						11	EU**			
7	ES,FR,IT,GB,EU,DE,DK,PL					13	EU										
8	ES,FR,IT,GB,EU,DE,DK,PL,SW																

* Si poteva alternativamente mantenere GB come predecessore

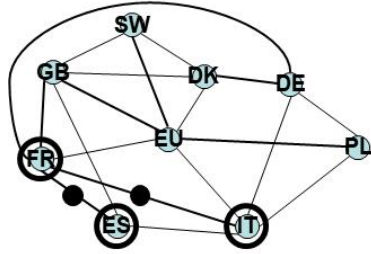
** Si poteva alternativamente scegliere prima PL



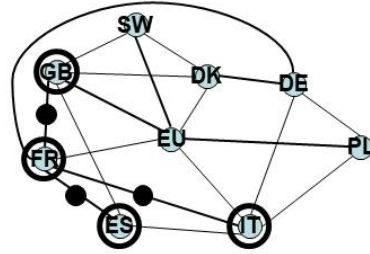
Iterazione 1



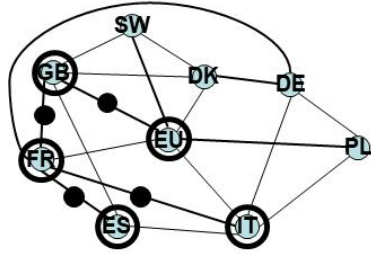
Iterazione 2



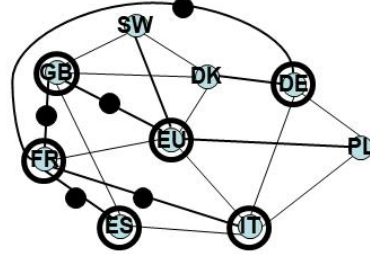
Iterazione 3



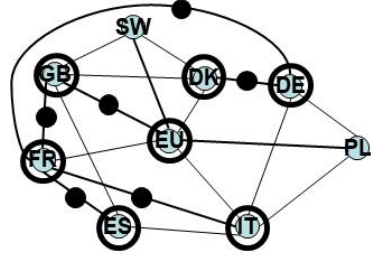
Iterazione 4



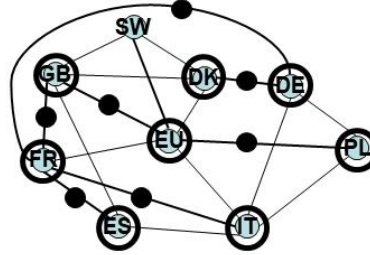
Iterazione 5



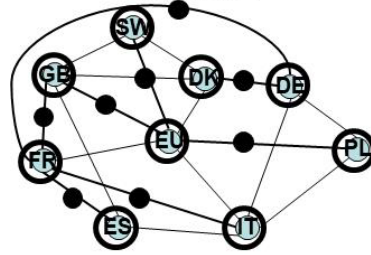
Iterazione 6



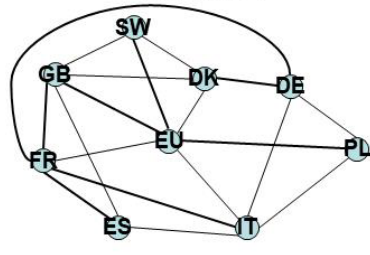
Iterazione 7



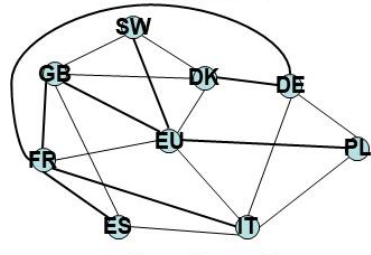
Iterazione 8



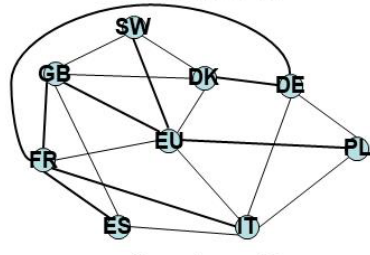
Iterazione 9



Iterazione 10



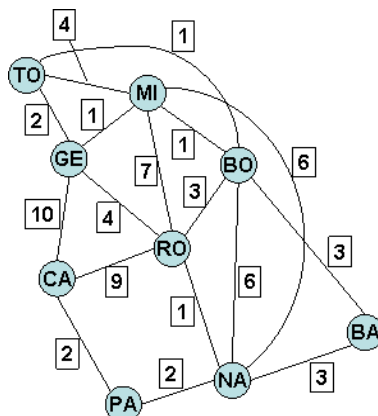
Iterazione 11



Iterazione 12

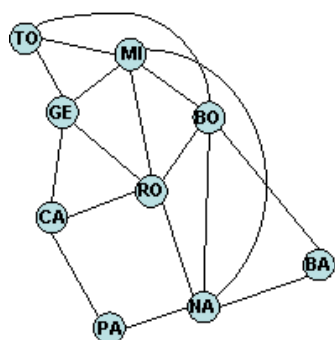
Esercizio 5.40

Data la rete rappresenta in figura si trovi l'albero dei cammini minimi del nodo MI mediante l'algoritmo di *Dijkstra*. Utilizzando i grafi seguenti, ad ogni passo si segnino sul grafo relativo il ramo ed il nodo aggiunti. Inoltre si riporti nella tabella seguente ad ogni passo e per ogni nodo x il vettore: (D_x, p_x) , dove p_x è il precedente di x nel percorso.

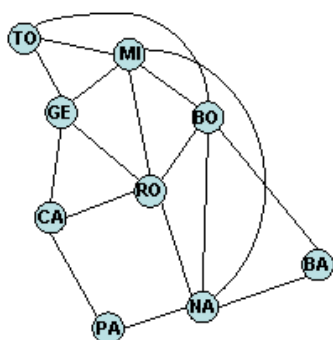


Passo	M		TO		GE		BO		RO		CA		NA		BA		PA	
			D_{TO}	p_{TO}	D_{GE}	p_{GE}	D_{BO}	p_{BO}	D_{RO}	p_{RO}	D_{CA}	p_{CA}	D_{NA}	p_{NA}	D_{BA}	p_{BA}	D_{PA}	p_{PA}
0																		
1																		
2																		
3																		
4																		
5																		
6																		
7																		
8																		

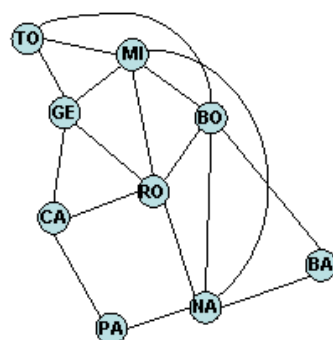
Nota: in caso di confronto tra nodi di uguale costo, scegliere secondo l'ordine alfabetico



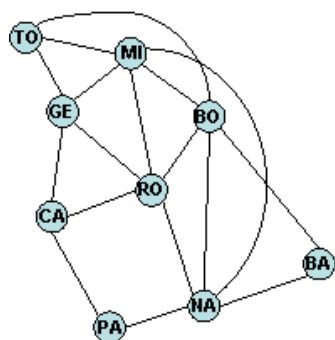
Iterazione 1



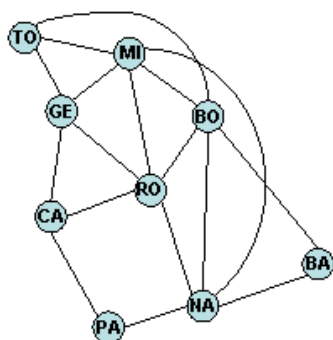
Iterazione 2



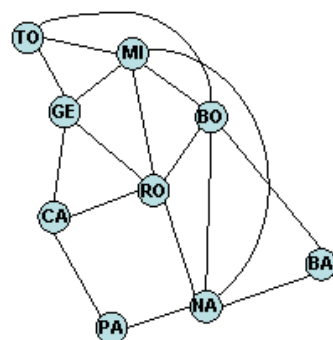
Iterazione 3



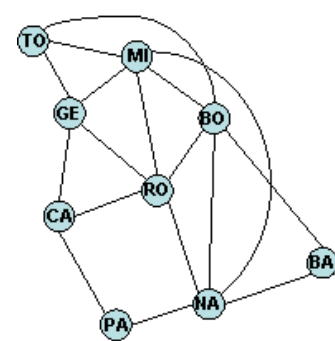
Iterazione 4



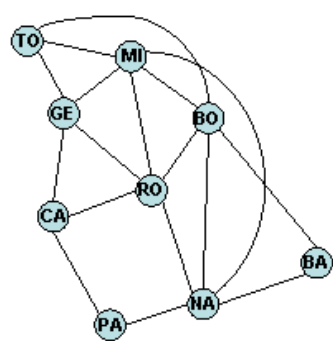
Iterazione 5



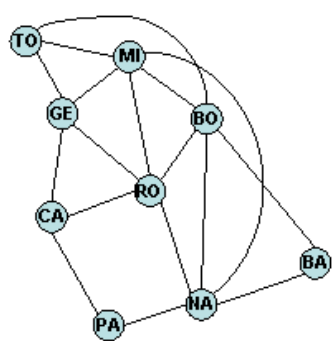
Iterazione 6



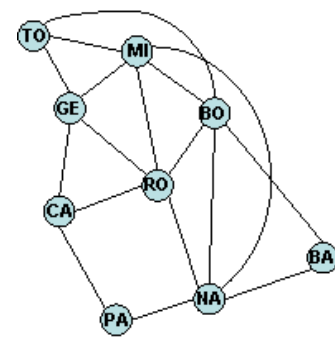
Iterazione 7



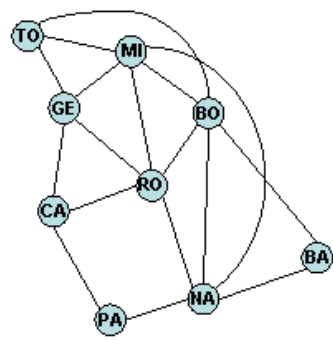
Iterazione 8



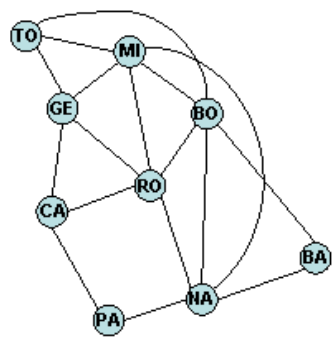
Iterazione 9



Iterazione 10

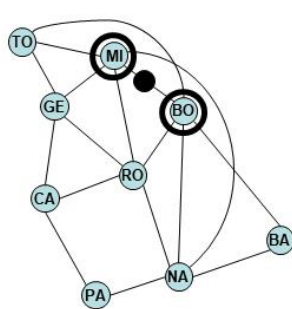


Iterazione 11

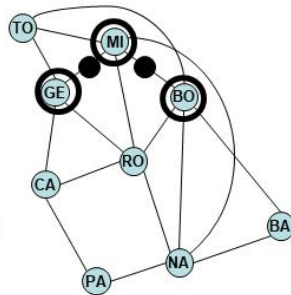


Iterazione 12

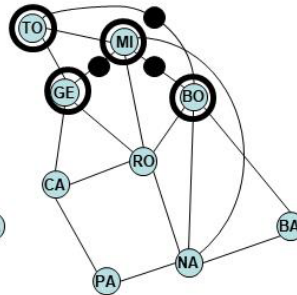
Passo	M	TO		GE		BO		RO		CA		NA		BA		PA	
		D _{TO}	P _{TO}	D _{GE}	P _{GE}	D _{BO}	P _{BO}	D _{RO}	P _{RO}	D _{CA}	P _{CA}	D _{NA}	P _{NA}	D _{BA}	P _{BA}	D _{PA}	P _{PA}
0	MI	4	MI	1	MI	1	MI	7	MI			6	MI				
1	MI,BO	2	BO	1	MI			4	BO			6	MI	4	BO		
2	MI,BO,GE	2	BO					4	BO	11	GE	6	MI	4	BO		
3	MI,BO,GE,TO							4	BO	11	GE	6	MI	4	BO		
4	MI,BO,GE,TO,BA							4	BO	11	GE	6	MI				
5	MI,BO,GE,TO,BA,RO									11	GE	5	RO				
6	MI,BO,GE,TO,BA,RO,NA									11	GE					7	NA
7	MI,BO,GE,TO,BA,RO,NA,PA									9	PA						
8	MI,BO,GE,TO,BA,RO,NA,PA,CA																



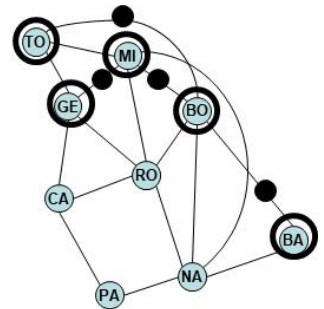
Iterazione 1



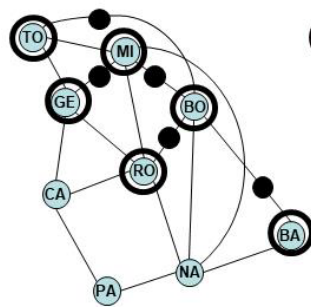
Iterazione 2



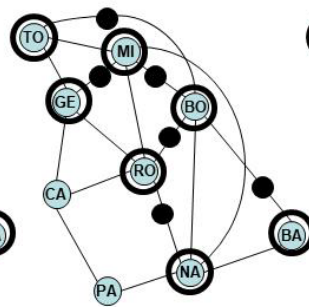
Iterazione 3



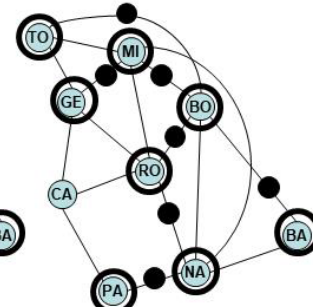
Iterazione 4



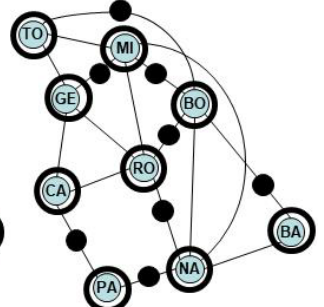
Iterazione 5



Iterazione 6



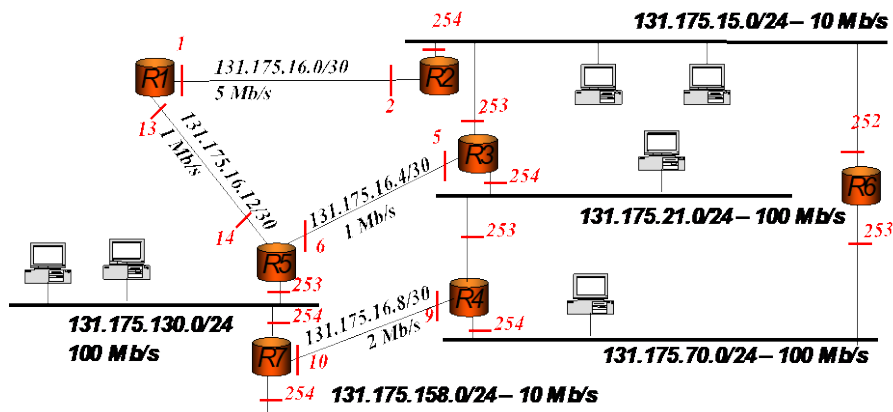
Iterazione 7



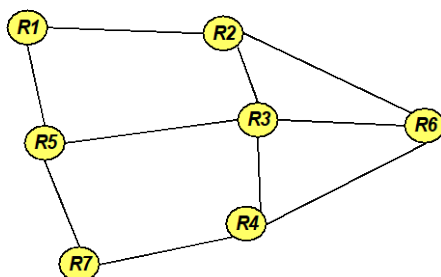
Iterazione 8

Esercizio 5.41

Si consideri la rete in figura. Si rappresenti, mediante un grafo, la rete per il calcolo dei cammini minimi (solo i nodi e gli archi – no reti). Si calcoli il cammino minimo tra R1 e tutti gli altri nodi mediante l'algoritmo di *Dijkstra* supponendo che ciascun arco abbia peso unitario. Si ripeta il calcolo assegnando a ciascun arco un peso pari a $100/C$ dove C è la velocità del *link* in Mbit/s.



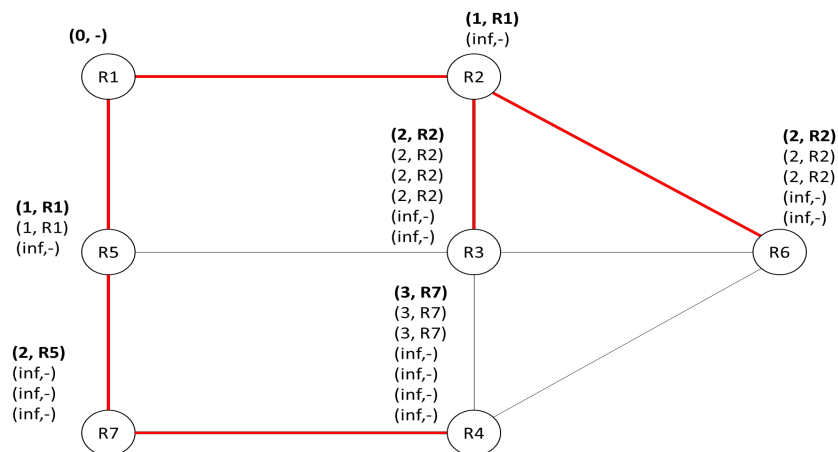
1-Il grafo che rappresenta la rete sopra è il seguente, rappresentando, come richiesto nel testo, solo router e collegamenti.



La tabella seguente mostra l'evoluzione dell'algoritmo di *Dijkstra*. Lo *Step 0* si riferisce all'inizializzazione. Ad ogni passo (*step*) sono indicati i router inclusi nell'albero dei cammini minimi (o Minimum Spanning Tree, MST) nella colonna MST.

Passo	MST	R1	R2	R3	R4	R5	R6	R7
0	R1	(0, -)	(1, R1)	(inf, -)	(inf, -)	(1, R1)	(inf, -)	(inf, -)
1	R1, R2			(2, R2)	(inf, -)	(1, R1)	(2, R2)	(inf, -)
2	R1, R2, R5			(2, R2)	(inf, -)		(2, R2)	(2, R5)
3	R1, R2, R5, R7			(2, R2)	(3, R7)		(2, R2)	
4	R1, R2, R5, R7, R6			(2, R2)	(3, R7)			
5	R1, R2, R5, R7, R6, R3				(3, R7)			
6	R1, R2, R5, R7, R6, R3, R4							

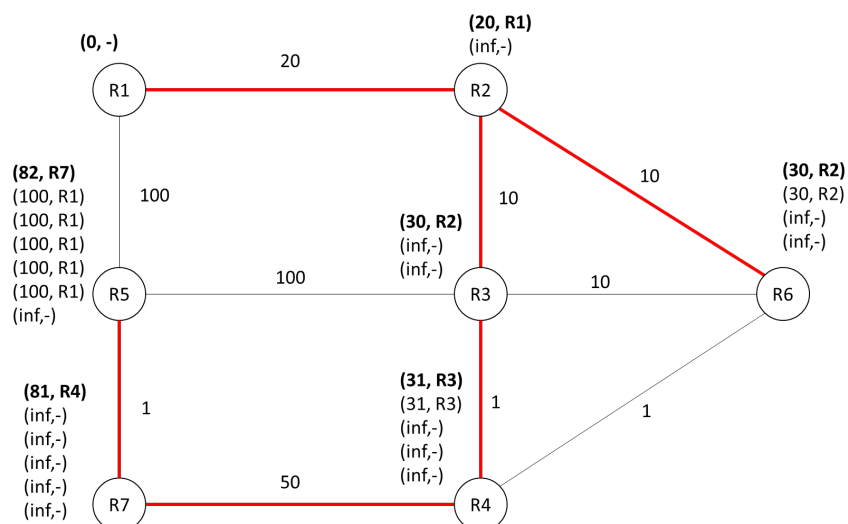
L'albero dei cammini minimi sarà il seguente:



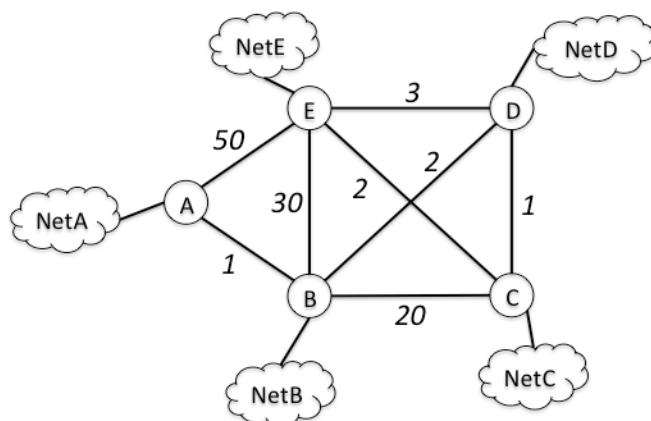
2-Applicando l'algoritmo di *Dijkstra* con la nuova metrica $100/C$, si ottiene il seguente albero dei cammini minimi:

Passo	MST	R1	R2	R3	R4	R5	R6	R7
0	R1	(0, -)	(20, R1)	(inf, -)	(inf, -)	(100, R1)	(inf, -)	(inf, -)
1	R1, R2			(30, R2)	(inf, -)	(100, R1)	(30, R2)	(inf, -)
2	R1, R2, R3				(31, R3)	(100, R1)	(30, R2)	(inf, -)
3	R1, R2, R3, R6				(31, R3)	(100, R1)		(inf, -)
4	R1, R2, R3, R6, R4					(100, R1)		(81, R4)
5	R1, R2, R3, R6, R4, R7					(82, R7)		
6	R1, R2, R3, R6, R4, R7, R5							

La figura seguente mostra l'albero dei cammini minimi.



Esercizio 5.42

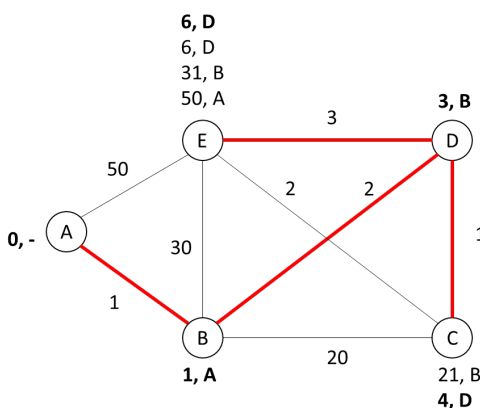


Sia data la rete in figura in cui sono indicati 5 *router* (X) e 5 reti afferenti Net(X). I costi dei collegamenti (bidirezionali) sono indicati accanto ai rispettivi *link*, si assumano pari ad 1 i costi dei collegamenti tra *router* e reti. Si chiede di:

- Calcolare l'albero dei cammini minimi con destinazione in A (considerando solo i *router*). Si utilizzi l'algoritmo più efficiente, rendendo comprensibili i passaggi svolti.
- Considerando l'instradamento calcolato al punto a), indicare i *Distance vector* inviati da A. A chi vengono inviati?
- Indicare i *Distance vector* con *Split-Horizon* (non *Poisonous-Reverse*) inviati da A. A chi vengono inviati?

a) Il grafo di rete presenta archi con pesi solo positivi. Possiamo usare l'algoritmo di *Dijkstra*.

Passo	MST	A	B	C	D	E
0	A	(0, -)	(1, A)	(inf, -)	(inf, -)	(50, A)
1	A, B			(21, B)	(3, B)	(31, B)
2	A, B, D			(4, D)		(6, D)
3	A, B, D, C					(6, D)
4	A, B, D, C, E					



- Il *router* A invia i propri *distance vector* ai *router* vicini B ed E. Il contenuto dei DV è il seguente:

NetA,1; NetB,2; NetC,5; NetD,4; NetE,7.

- c) Il *router* A non include nei DV le destinazioni che il *router* stesso raggiunge usando come next hop il destinatario del DV. Il *router* A usa il *router* B come next hop per tutte le destinazioni (tranne la rete NetA), quindi i contenuti dei due DV inviati a B ed E sono:

NetA,1; NetB,2; NetC,5; NetD,4; NetE,7. Inviato ad E

NetA,1. Inviato a B

Esercizio 5.43 (I)

Si supponga che i quattro *router* in figura usino il protocollo RIP versione 2. In figura è anche riportata la tabella di *routing* per il *router* R1. Si assuma che il costo dei collegamenti tra R1 e tutti gli altri *router* sia uguale a 1.

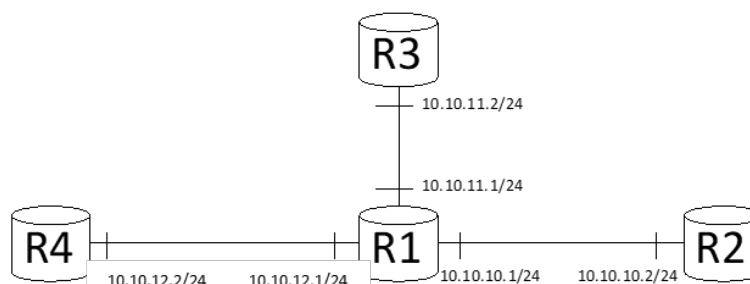


Tabella di *routing* di R1

Destinazione	Next Hop	Costo
131.175.124.0/24	10.10.10.2	3
131.180.0.0/16	10.10.11.2	6
131.175.16.0/24	10.10.12.2	9
131.175.9.0/24	10.10.10.2	11

- Indicare il contenuto dei messaggi di risposta RIPv2 inviati da R1 a tutti gli altri *router* nel caso in cui si usi la versione con *Poisonous Reverse* del protocollo di *routing*.
- Dire come cambia la seguente tabella di *routing* del *router* R1 quando riceve un messaggio di risposta RIPv2 da R3 con il seguente contenuto:

131.175.124.0/24 costo: 1
131.180.0.0/16 costo: 16
131.180.12.0/23 costo: 11
131.175.9.0/24 costo: 13

a) Al *router* R2

131.175.124.0/24 costo: 16
131.180.0.0/16 costo: 6
131.175.16.0/24 costo: 9
131.175.9.0/24 costo: 16

Al *router* R3

131.175.124.0/24 costo: 3
131.180.0.0/16 costo: 16
131.175.16.0/24 costo: 9
131.175.9.0/24 costo: 11

Al router R4

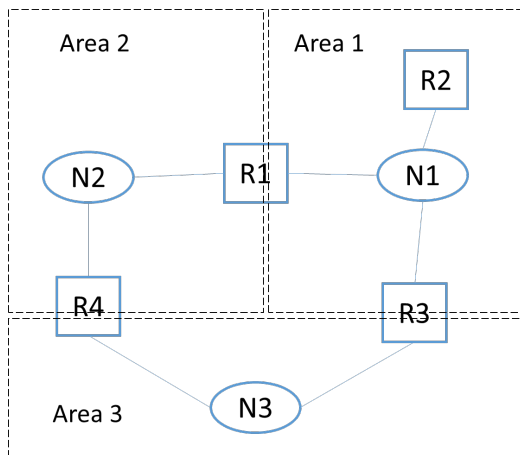
131.175.124.0/24	costo: 3
131.180.0.0/16	costo: 6
131.175.16.0/24	costo: 16
131.175.9.0/24	costo: 11

b)

131.175.124.0/24	10.10.11.2	2
131.180.0.0/16	10.10.11.2	16
131.180.12.0/23	10.10.11.2	12
131.175.16.0/24	10.10.12.2	9
131.175.9.0/24	10.10.10.2	11

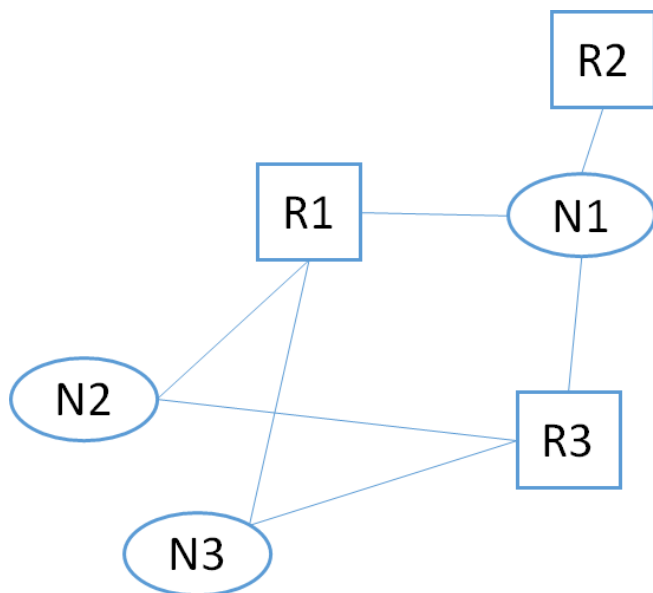
Esercizio 5.44

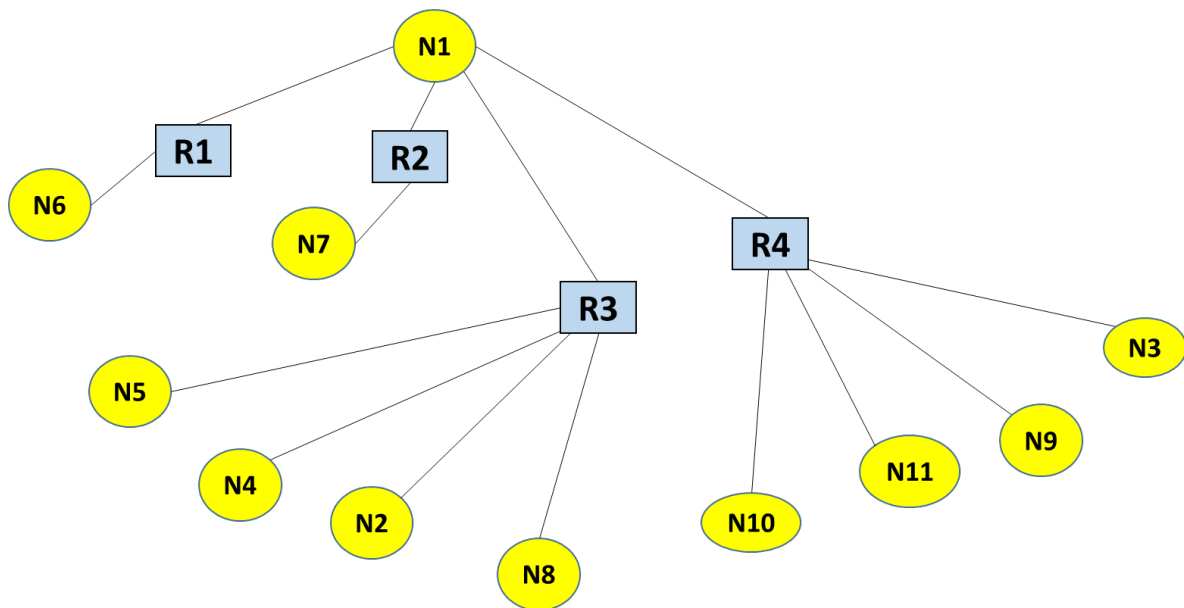
All'interno del dominio di *routing* in figura si usa il protocollo OSPF. Il dominio di *routing* è suddiviso in 3 aree come mostrato in figura (linee tratteggiate). Disegnare la topologia del dominio di *routing* “vista” dal router R2.



Il protocollo OSPF opera secondo questi due principi: (i) ogni *router* mantiene la rappresentazione di dettaglio di tutti i *link* presenti nella sua area, (ii) ogni *router* mantiene una rappresentazione sintetica di tutte le destinazioni (reti) raggiungibili al di fuori della sua area.

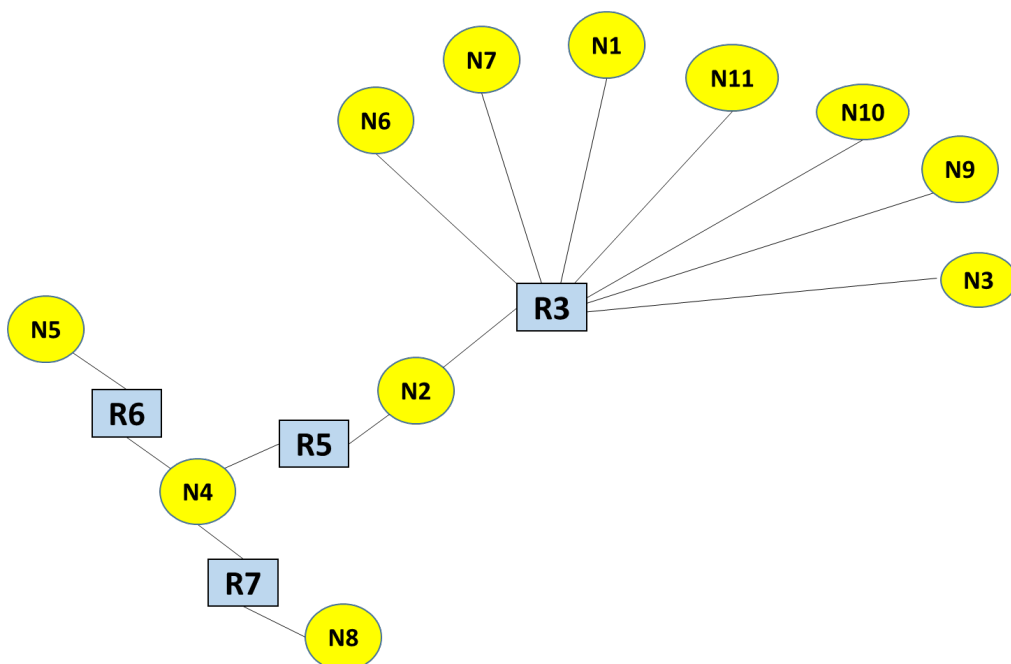
Nel caso specifico, il Router R2 avrà visione di dettaglio di tutti i *link* dell'Area 1, e “vedrà” tutte le reti esterne all'Area 1 (N2 e N3) come raggiungibili sia attraverso R1 che attraverso R3.





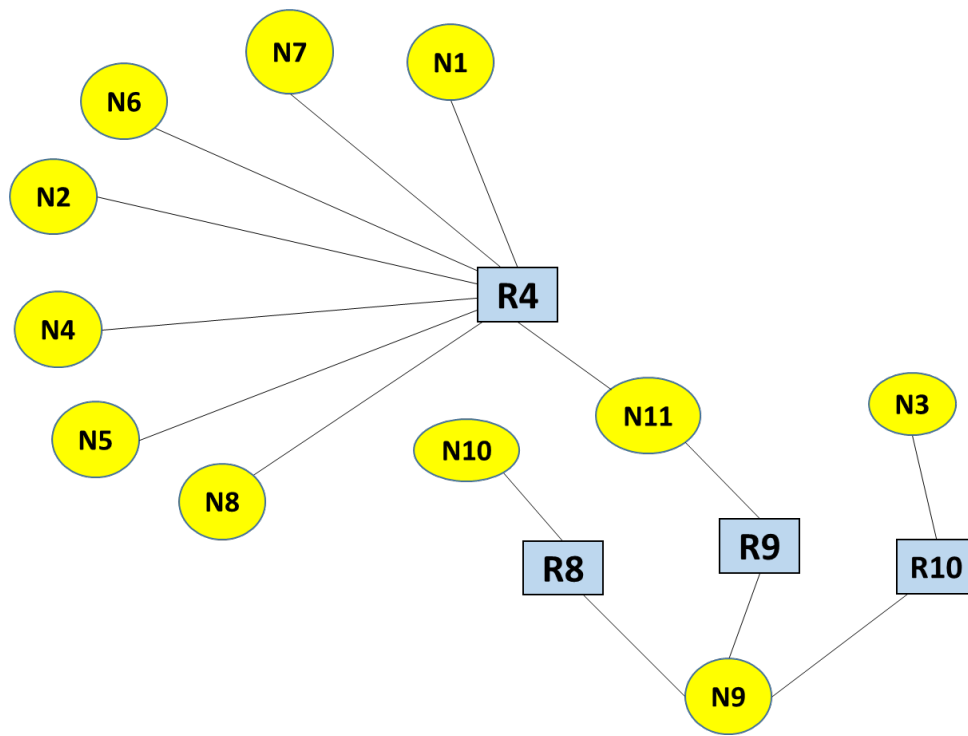
Visto da R7

Nel caso specifico, il Router R7 avrà visione di dettaglio di tutti i *costi* della sua area, e “vedrà” tutte le reti esterne alla sua area come raggiungibili attraverso R3.



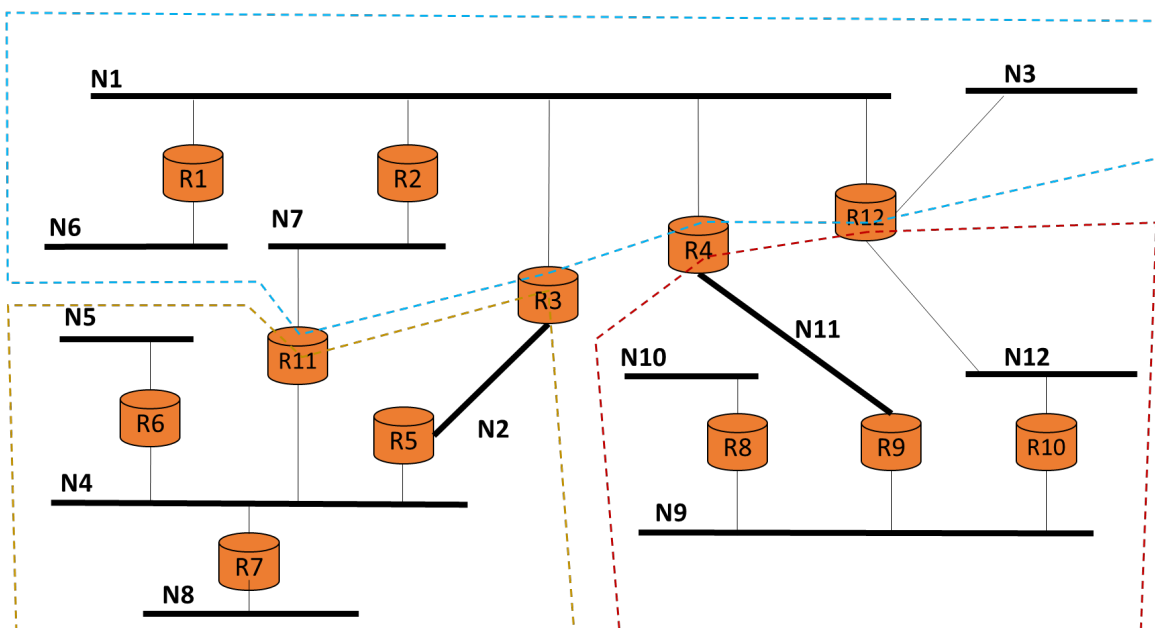
Visto da R10

Nel caso specifico, il Router R10 avrà visione di dettaglio di tutti i *costi* della sua area, e “vedrà” tutte le reti esterne alla sua area come raggiungibili attraverso R4.

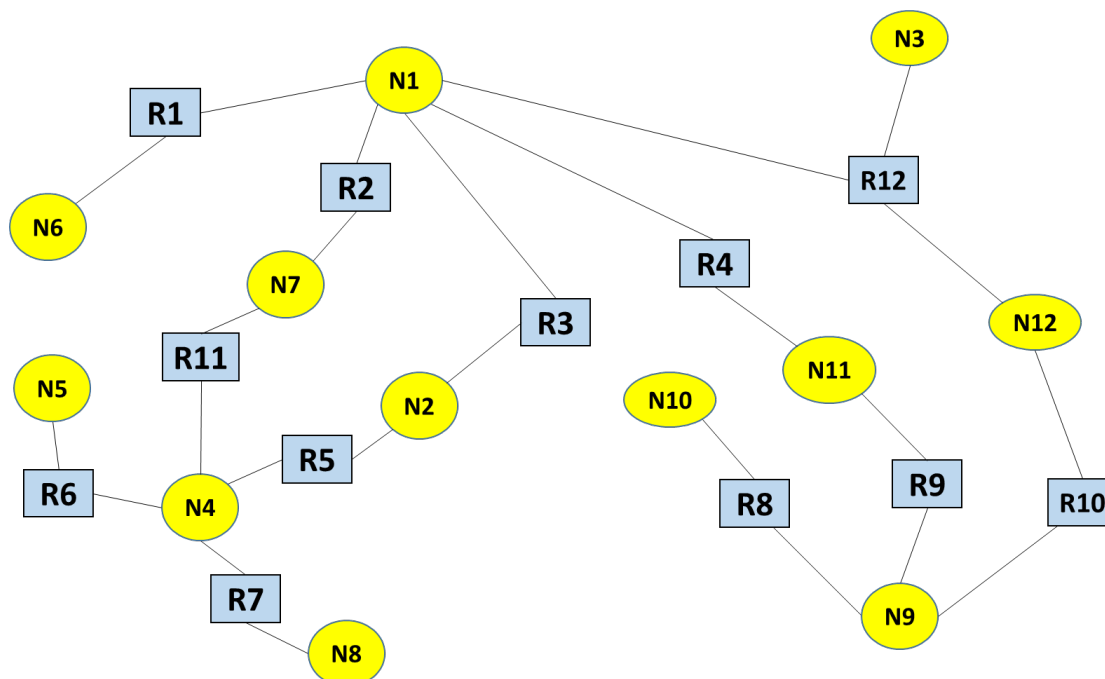


Esercizio 5.46

Si consideri la rete in figura dove sono indicati *router* e le reti da essi interfacciate. Si supponga di utilizzare il protocollo di *routing* OSPF. Si divida come mostrato in figura la rete in 3 aree e si disegnino i grafi che rappresentano la rete vista dal *router* R1, R7, ed R8.

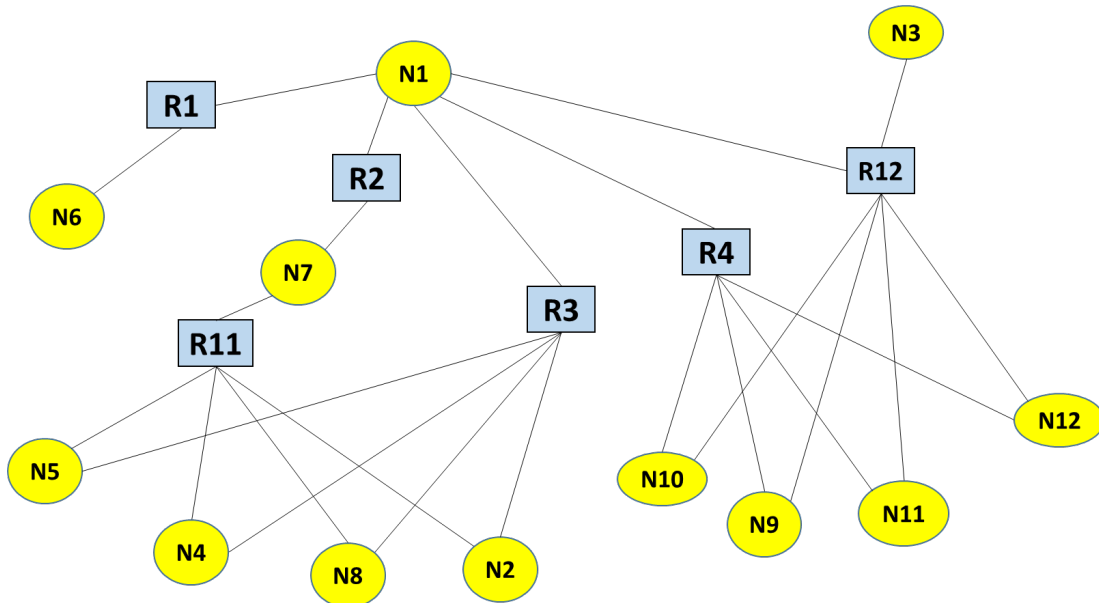


Unica area



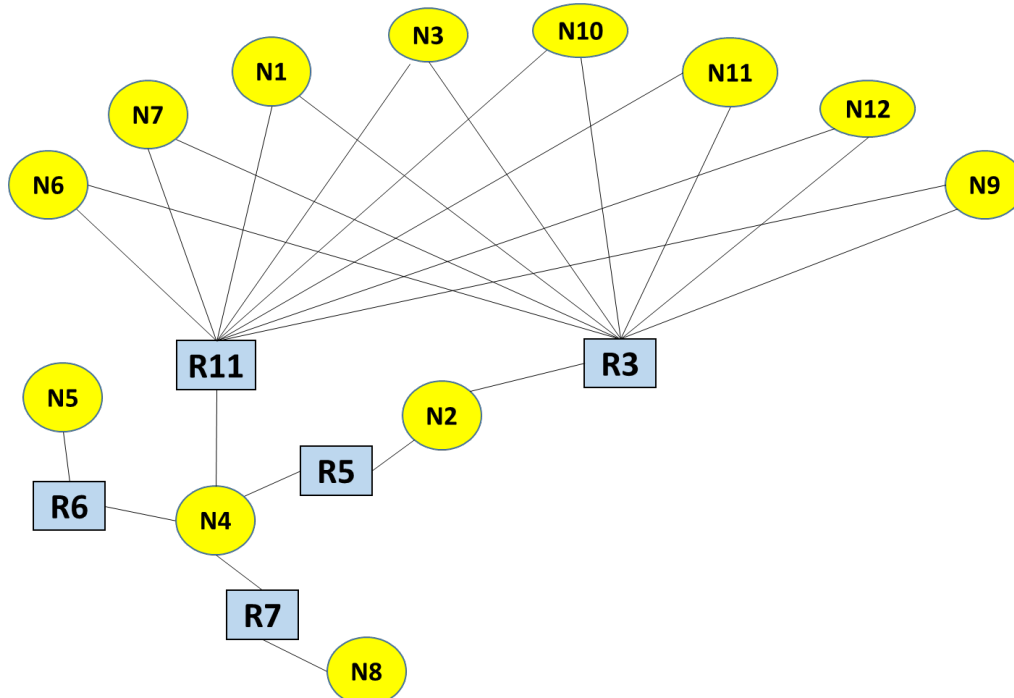
Visto da R1

In figura, i costi (in blu) mostrati per le reti al di fuori del dominio di R1 indicano il costo necessario per raggiungere tali reti **a partire** dai router R4 e R12.



Visto da R7

In figura, i costi (in blu) mostrati per le reti al di fuori del dominio di R1 indicano il costo necessario per raggiungere tali reti **a partire** dai router R4 e R12.

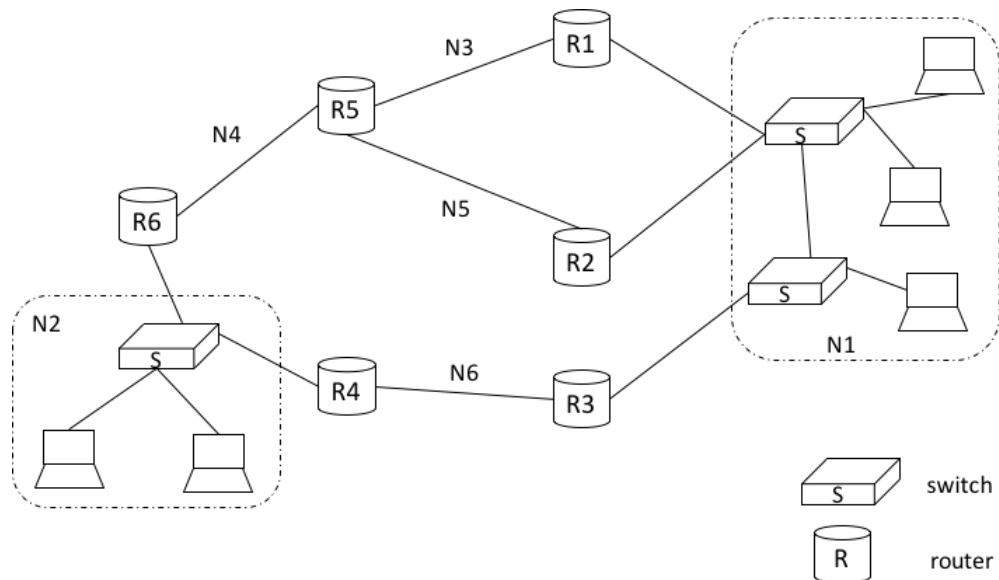


Visto da R8 è simile. Il *router* vede tutta la propria area e solo le reti esterne raggiungibili dai *router* di bordo area.

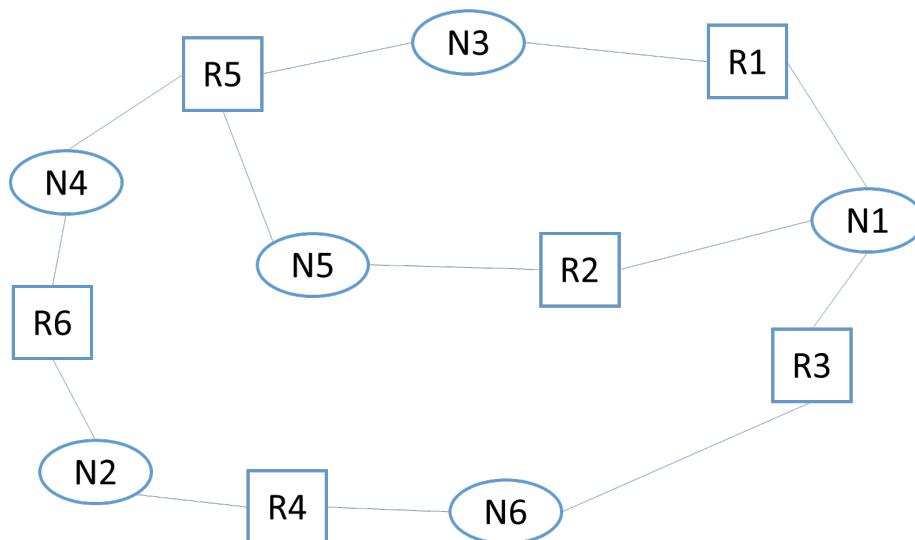
Esercizio 5.47 (E)

Si consideri la rete in figura composta da 6 *router* e 6 reti (inclusi i collegamenti punto-punto).

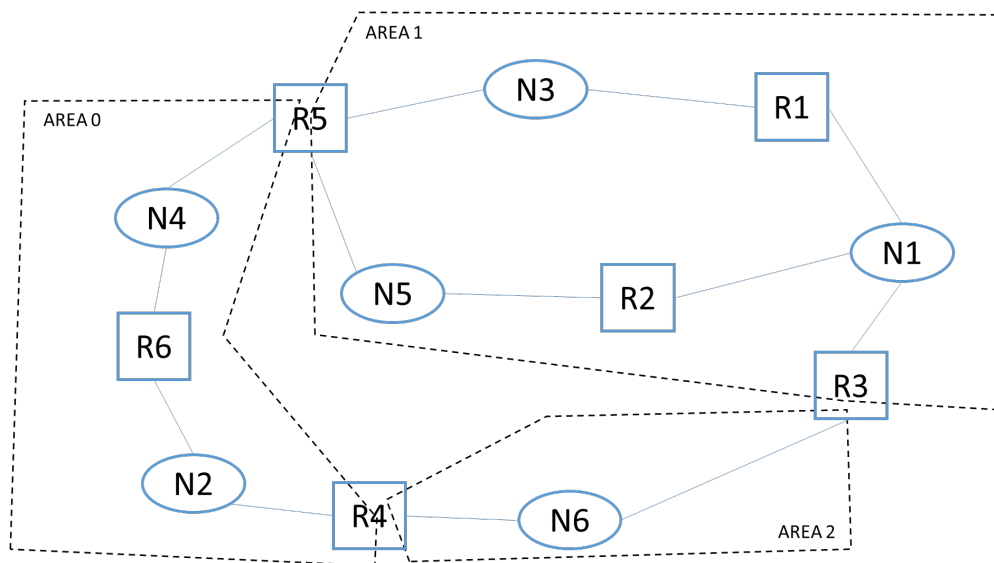
- Si rappresenti la rete con un grafo che ha come nodi i *router* (indicati con quadrati) e le reti (indicati cerchi), e come archi le interfacce dei *router* con le reti.
- Si assuma che la rete usi il protocollo OSPF e sia divisa in tre aree. Area 0 (R4, R5, R6, N2, N4), Area 1 (R1, R2, R3, R5, N1, N3, N5), Area 2 (R3, R4, N6). Si disegni la topologia della rete vista da R1.



a) Grafo



b) divisione in aree:



Topologia vista da R1. La topologia è quella dettagliata dell'area 1 a cui R1 appartiene e un riassunto della raggiungibilità attraverso gli area border router R5 e R3.

