

# Router

Un router ha queste funzioni principali:

- Instradamento dei dati
- Connessione tra le varie reti WAN
- Connessione a Internet
- Limitare il broadcast alle singole subnet

Il compito fondamentale del router è quindi l'instradamento, cioè una funzione del livello 3 (dell'ISO-OSI). Il router non fa altro che gestire gli insiemi di indirizzi, determinando percorsi computazionalmente validi affinché i messaggi siano consegnati ai vari destinatari.

La segmentazione della rete si ispira al concetto divide et impera.

Funzioni di routing:

- Creare e gestire tabelle di routing;
- Determinare tramite le tabelle di routing i percorsi (minimi...).

Un router ha le stesse componenti di un normale PC.

- CPU
- Memoria
- System bus
- Interfacce input/output

Nota: segue il modello di Von Neumann...

Di router ne esistono innumerevoli tipi e specie. Può essere usato per dirigere reti LAN domestiche con e senza connessione WI-FI. Ma ci sono router che possono regolarizzare il traffico su reti aziendali relativamente grandi.

Anche i router necessitano quindi un sistema operativo specifico, questo viene chiamato IOS (Internetwork Operating System).

Normalmente i router sono configurati dall'amministratore della rete.

Classifichiamo le componenti fisiche come:

- Porte di accesso ( $\geq 2$ )
- Porte di uscita ( $\geq 2$ )
- Matrice di commutazione: collega ciascuna porta di ingresso a ciascuna porta di uscita
- Processore di controllo, RAM, NVRAM, flash memory, ROM

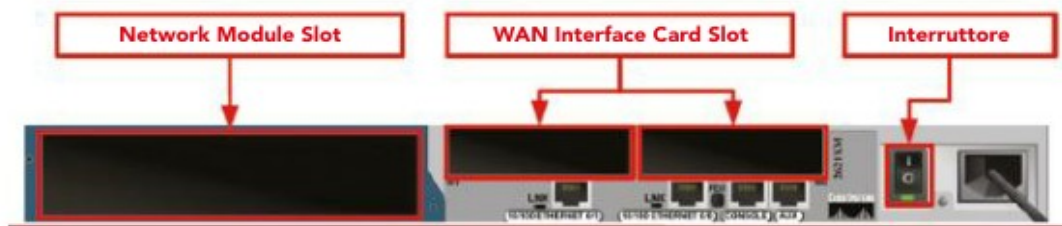
## Un esempio di router Cisco....

Prenderemo questo modello proprio perchè essendo un router modulare permette di installare su di esso qualsiasi configurazione. Nota: appartiene alla fascia midrange.. (70\$ su amazon usato)



- CPU: esegue IOS, gestisce il routing, possiamo avere più CPU
- RAM: ha le normali funzioni di una RAM... (DRAM, con moduli DIMM)

- Flash: contiene l'immagine del sistema operativo
- Bus:
  - Bus di sistema: cpu-interfacce
  - CPU-Bus: cpu-memorie
- NVRAM: contiene la configurazione all'avvio del router
- ROM:
  - Effettua la diagnostica
  - Carica l'IOS dalla Flash
- Porte:
  - Possiamo aggiungere tramite espansione:
    - Porte ethernet (Network Module Slot)
    - Porte seriali o ISDN (WAN Interface Card Slot)



- AUX/Console: usate per configurare il router, recovery o pw recovery
- Power supply: alimentazione

## Fondamenti di Routing

Il routing consente a due nodi A e B, non collegati direttamente di comunicare tra di loro mediante la collaborazione di altri nodi posti su un cammino nella rete che connette A e B.

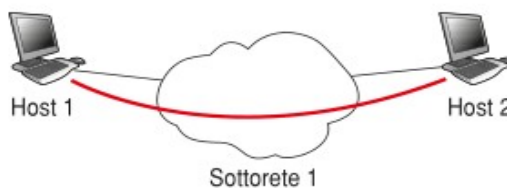
L'instradamento è alla base del livello 3 (ISO-OSI).

I terminali della rete vengono chiamati host e i nodi di commutazione router.

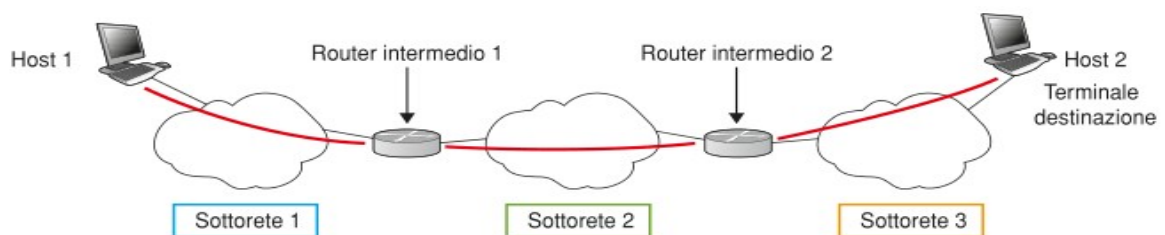
I router sono dispositivi con uscite multiple. Ricevono un pacchetto da un'interfaccia e lo instradano su un'altra a secondo dell'indirizzo univoco assegnato ad ogni host.

L'instradamento può essere:

- Diretto: due host solo collegati senza router intermedi alla stessa sottorete, il messaggio viene instradato a secondo del suo mac



- Indiretto: due host sono collegati a sottoreti diverse attraverso uno o più router. Il mittente fornisce l'indirizzo del router, quando questo perciò riceverà il pacchetto lo instraderà ad un altro router...., fino a destinazione!



Ogni router contiene una tabella di instradamento che contiene informazioni relative alle destinazioni conosciute. Una tabella è un insieme di righe, ciascuna delle quali contiene:

- Indirizzo della rete di destinazione (N)
- Maschera di rete (M)
- Indirizzo del prossimo router (Next Hop)
- Interfaccia su cui inoltrare i pacchetti

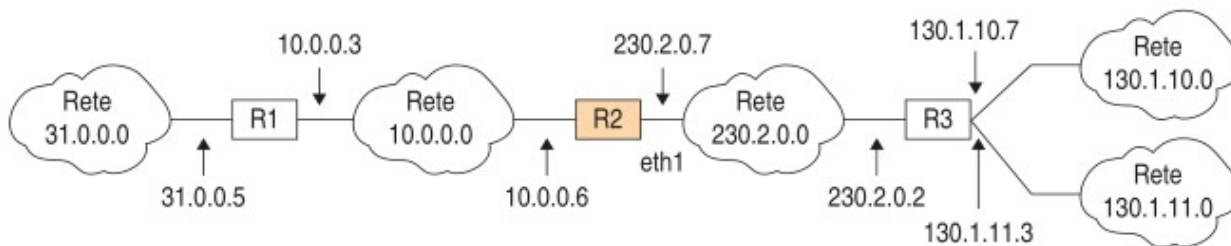


Tabella di routing per R2:

Destinazione N	Maschera M	Next hop NH	Interfaccia I
10.0.0.0	255.0.0.0	ind.diretto (10.0.0.6)	ethernet0
230.2.0.0	255.255.0.0	ind.diretto (230.2.0.7)	ethernet1
31.0.0.0	255.0.0.0	10.0.0.3	ethernet0
130.1.10.0	255.255.255.0	230.2.0.2	ethernet1
130.1.11.0	255.255.255.0	230.2.0.2	ethernet1

Se una destinazione non è presente nella tabella il pacchetto viene instradato al router di default.

L'indirizzo di questo viene inserito come ultima riga ed è codificato con tutti 0 sia nel campo N che nel campo M.

Destinazione N	Maschera M	Next hop NH	Interfaccia I
10.0.0.0	255.0.0.0	ind.diretto (10.0.0.6)	ethernet0
230.2.0.0	255.255.0.0	ind.diretto (230.2.0.7)	ethernet1
31.0.0.0	255.0.0.0	10.0.0.3	ethernet0
130.1.10.0	255.255.255.0	230.2.0.2	ethernet1
130.1.11.0	255.255.255.0	230.2.0.2	ethernet1
0.0.0.0	0.0.0.0	10.0.0.3	ethernet0

### Esercizio

Data la seguente tabella, individuare verso quale porta vengono instradati i pacchetti indirizzati ai seguenti indirizzi:

198.12.17.3

198.12.0.3

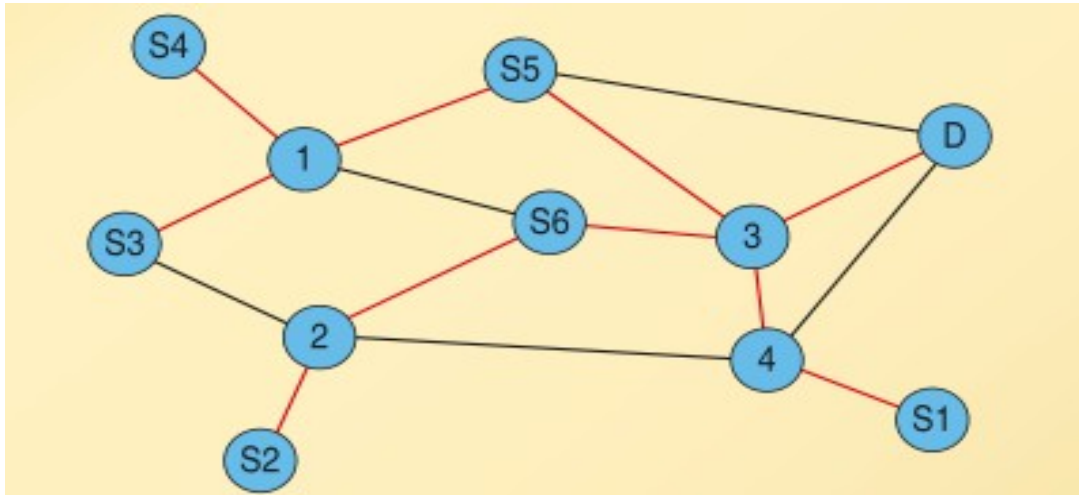
20.12.0.3

Destinazione (Y)	M	Porta di uscita
198.12.0.0	255.255.0.0	1
198.12.17.0	255.255.255.0	5
0.0.0.0	0.0.0.0	2

## Algoritmi di routing

Un algoritmo di routing stabilisce i criteri di scelta del cammino che devono effettuare i pacchetti all'interno della rete.

Il vincolo primario è che ogni cammino dal mittente di un pacchetto fino al destinatario deve necessariamente essere un albero.



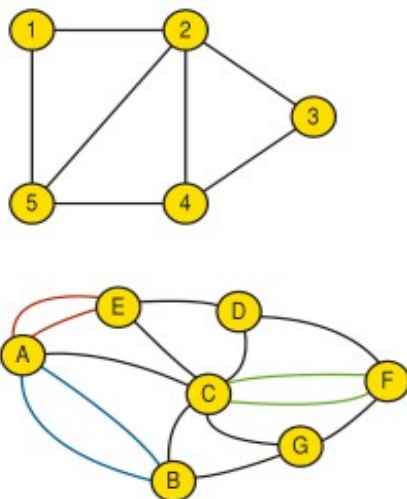
Ogni cammino potrà essere schematizzato come un albero, cioè senza percorsi di tipo ciclico. Ogni rete può essere schematizzata con un grafo. Un grafo è uno schema composto da nodi (o vertici) e archi (o spigoli). I nodi rappresentano i dispositivi che intervengono sulla rete stessa, mentre gli archi sono i vari collegamenti. I percorsi per ogni pacchetto devono seguire le leggi stabilite da un algoritmo. Questi algoritmi trovano il cammino minimo (quindi il cammino più conveniente possibile) dal mittente al destinatario. Ad ogni arco verrà associato un peso, anch'esso influirà nella ricerca del cammino minimo.

## Grafi

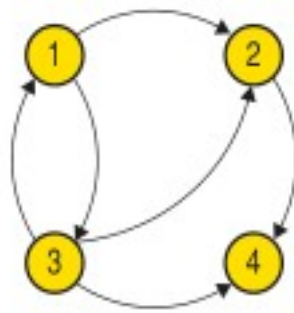
Come già detto in precedenza si può schematizzare una rete con un Grafo.

Un grafo  $G$  è una coppia  $(V, E)$ :

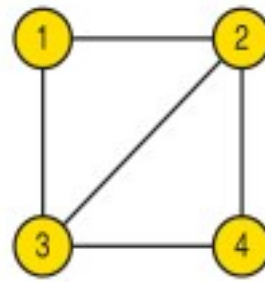
- $V$  insieme di vertici
- $E$  è l'insieme delle coppie di vertici, detti archi



Un grafo si definisce orientato se su ogni arco vi è definito l'ordinamento della coppia di vertici, cioè che la direzione dal nodo di partenza al nodo di origine.

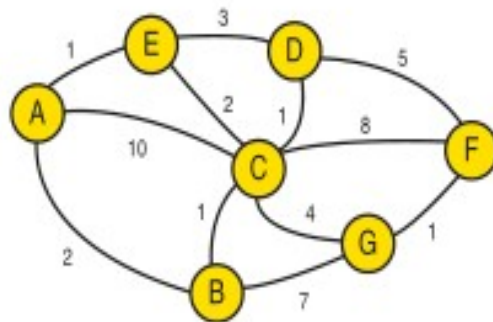


Grafo orientato

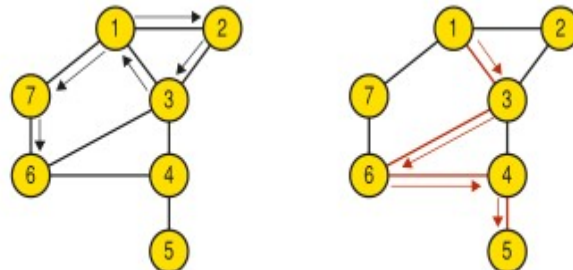


Grafo non orientato

In un grafo pesato ogni arco è contraddistinto da un peso.



Il cammino è una sequenza di vertici adiacenti, che ci permette di andare da un nodo ad un altro, è definito semplice se contiene solo nodi distinti.



Per lunghezza di un cammino si intende il numero di archi che lo compone.

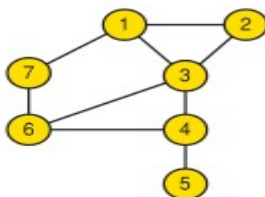
Il peso di un cammino è la somma di tutti i pesi associati ad ogni arco che fa parte di quel cammino.

Per distanza si intende il cammino più breve tra due nodi.

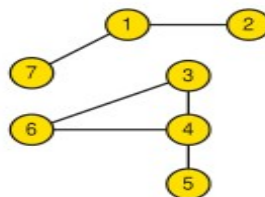
Un circuito è un cammino che partendo da un nodo  $n_1$  torna allo stesso senza archi ripetuti.

Un ciclo è un cammino che partendo da un nodo  $n_1$  torna allo stesso senza archi nè nodi ripetuti.

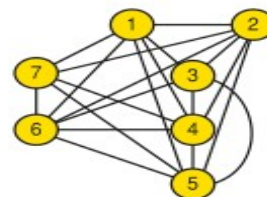
Un grafo si definisce aciclico se non contiene cicli, un grafo connesso e aciclico è definito un albero.



Grafo connesso

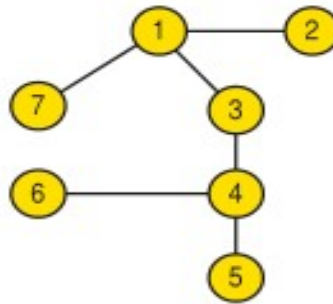


Grafo non connesso

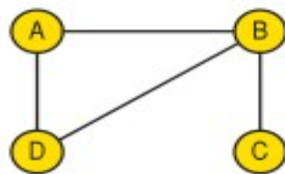


Grafo completamente connesso

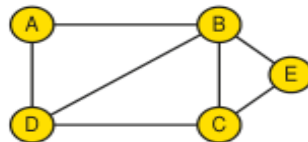
Albero



Per rappresentare un grafo possiamo usare la matrice di adiacenze. In tale matrice l'elemento è 1 se i nodi sono collegati direttamente, 0 altrimenti. Vediamo alcuni esempi:

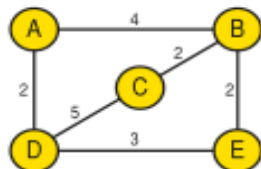


	A	B	C	D
A	0	1	0	1
B	1	0	1	1
C	0	1	0	0
D	1	1	0	0

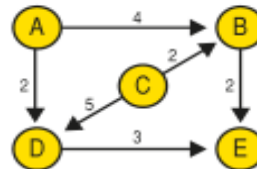


	A	B	C	D	E
A	0	1	0	1	0
B	1	0	1	1	1
C	0	1	0	1	1
D	1	1	1	0	0
E	0	1	1	0	0

Nei grafi pesati si mette il valore del peso al posto dell'1 e  $\infty$  al posto dello 0 (0 si lascia sulla diagonale...).

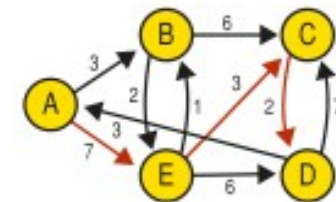
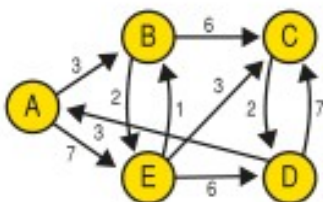


	A	B	C	D	E
A	0	4	$\infty$	2	$\infty$
B	4	0	2	$\infty$	2
C	$\infty$	2	0	5	$\infty$
D	2	$\infty$	5	0	3
E	$\infty$	2	$\infty$	3	0

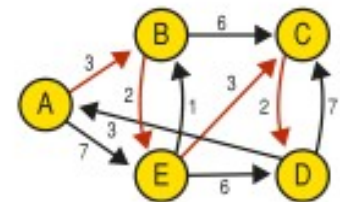


	A	B	C	D	E
A	0	4	$\infty$	2	$\infty$
B	$\infty$	0	$\infty$	$\infty$	2
C	$\infty$	2	0	5	$\infty$
D	$\infty$	$\infty$	$\infty$	0	3
E	$\infty$	$\infty$	$\infty$	$\infty$	0

Un cammino  $p_2$  è definito minimo (ottimo) in questo caso:



$p_1(A \dots D) = (A, E, C, D)$   
Peso di  $p_1 = 7 + 1 + 2 = 10$

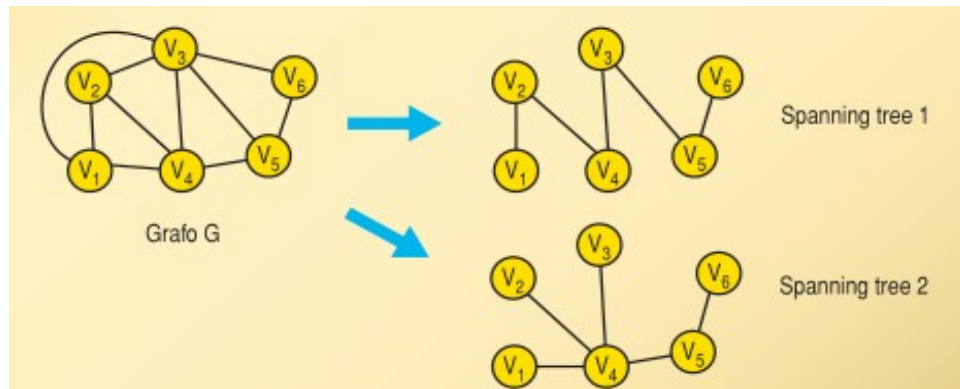


$p_2(A \dots D) = (A, B, E, C, D)$   
Peso di  $p_2 = 3 + 1 + 2 + 2 = 8$

L'insieme dei percorsi minimi definisce l'albero di inoltro dove il router sorgente viene chiamato radice (root) e i router destinazioni foglie (leaves).



È possibile trasformare un grafo in un albero di ricoprimento (spanning tree), cioè un suo sottografo (se gli insiemi dei suoi nodi e dei suoi spigoli sono sottoinsiemi di quelli del grafo..), privato di tutti i cicli.



Tra tutti gli alberi ricoprenti è possibile determinare il Minimum Spanning Tree, cioè quello con il peso minore.

## Algoritmi di routing statici

Ogni router contiene la topologia della rete di appartenenza. Le decisioni di router sono prese in anticipo, all'avvio della rete.

### Principio di ottimalità:

se il router j è nel cammino ottimo fra i e k, allora anche il cammino ottimo fra j e k è sulla stessa strada.

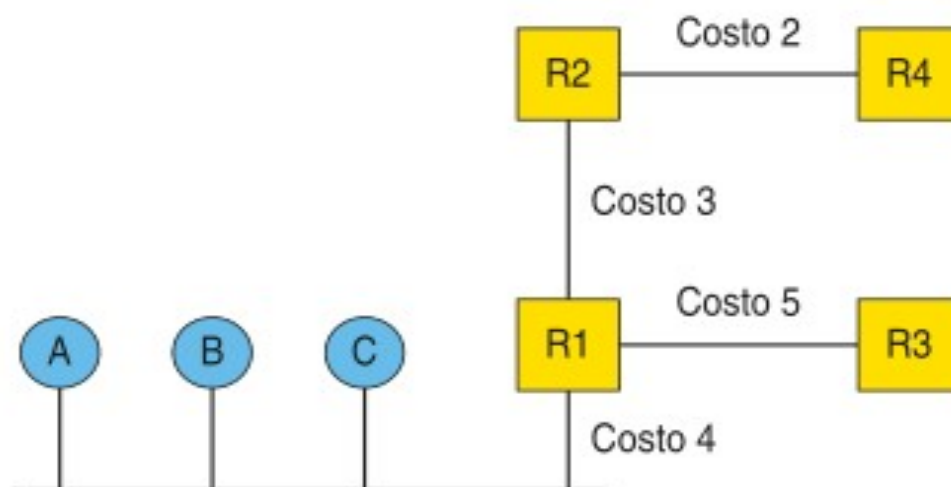
Gli algoritmi statici si definiscono anche Link State, perchè ogni router dopo che aver acquisito tutte le informazioni che gli servono invia a tutti gli altri nodi dei Link State Packet (LSP), contenenti una serie di dati come: lo stato di ogni link, il costo o anche il numero sequenziale per l'LSP.

Al ricevimento degli LSP ogni nodo si costruirà un database con questi e ricostruirà la topologia della rete; successivamente procederà a calcolare i cammini minimi.

Al ricevimento dell'LSP il router deve controllare il numero di sequenza:

- Se non ha mai ricevuto LSP da quel router o l'LSP è più recente di quello in suo possesso deve aggiornare il suo database e rimandare agli altri questo pacchetto nuovo
- Se il numero di sequenza è uguale a quello in suo possesso non farà nulla
- Se il router ha un LSP più recente di quello ricevuto, invia all'altro nodo il nuovo LSP

Proviamo a costruire un LS database da questa situazione:



R1 riceve il peso di ogni collegamento e ricostruisce la seguente tabella:

Adiacenze R1/x	Costo
A	4
B	4
C	4
R1	0
R2	3
R3	5

Ogni router quindi invierà il suo messaggio:

Adiacenze R2/x	Costo
R1	3
R2	0
R4	2

Adiacenze R4/x	Costo
R2	2
R4	0

Adiacenze R3/x	Costo
R1	5
R3	0

Unendo tutte le informazioni precedenti sarà possibile stabilire questa matrice di adiacenza:

	A	B	C	R1	R2	R3	R4
A	0	0	0	4			
B	0	0	0	4			
C	0	0	0	4			
R1	4	4	4	0	3	5	
R2				3	0		2
R3				5		0	
R4					2		0

## Esempi di algoritmi statici

### Flooding:

Questo algoritmo invia ogni pacchetto su tutte le porte, tranne su quella dalla quale è arrivato.

- Svantaggi: Logicamente questa tecnica potrebbe provocare la duplicazione dello stesso pacchetto sullo stesso nodo per molte volte e anche che ogni pacchetto rimbalzi da un router all'altro in continuazione.... Per evitare ciò si mette un contatore che una volta azzerato arresta l'invio e soprattutto una volta che il router ha ricevuto un pacchetto se lo riceve nuovamente lo ignora.
- Vantaggi: questo tipo di algoritmo è ottimale in termini di tempi di elaborazione dei router. Viene usato appunto su topologie molto semplici e quando è richiesto un messaggio broadcast.



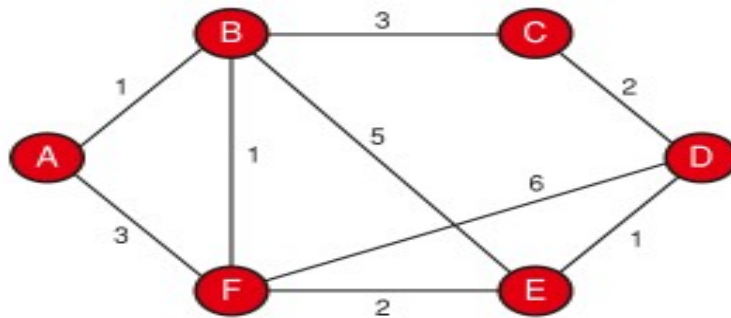
### Flow-based routing:

Questo algoritmo sceglie il percorso migliore calcolando il traffico medio di ogni linea. Un router può essere pensato come un automobilista che sceglie la strada migliore pensando al traffico che potrebbe trovare su ogni percorso. Peccato che per costruzione stessa dell'algoritmo questa tecnica non può essere usata da tutti i router in quanto tutti finirebbero ad instradare pacchetti sulle vie considerate meno percorse e lasciando libere le vie principali...

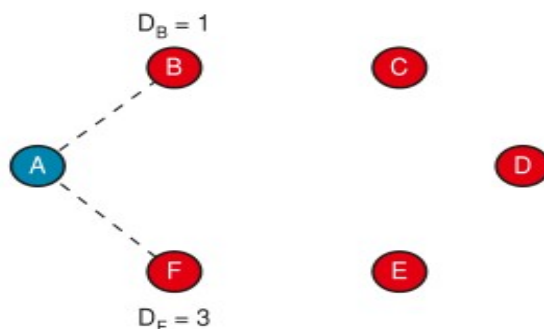
### L'algoritmo di Dijkstra:

È uno degli algoritmi più usati e conosciuti per routing statico. Permette di trovare l'albero dei cammini minimi partendo da ogni nodo; per questo viene definito centralizzato. Facciamo un esempio per spiegare il metodo:

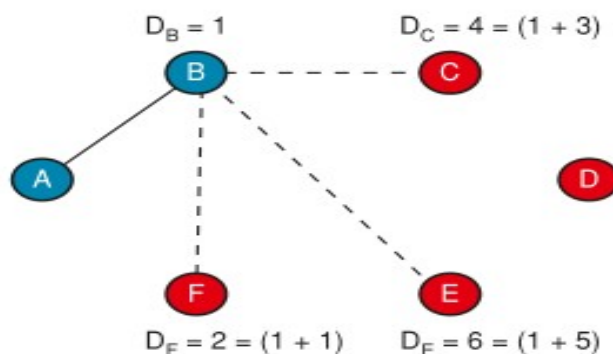
Prendiamo il grafo in figura e pensiamo di partire dal nodo A:



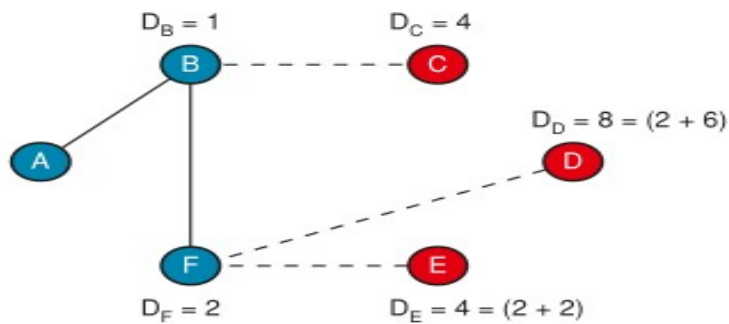
Etichettiamo B e F con il peso degli corrispondenti archi di collegamento con A



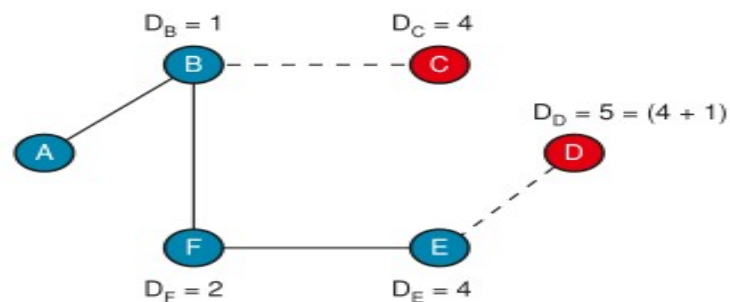
Scegliamo ora quindi l'arco più conveniente, quindi quello passante per B. A questo punto dobbiamo assegnare a tutti i nodi comunicanti con B una etichetta, ottenuta come somma dell'etichetta di B stesso e il peso di ogni arco. Se un nodo ha già un'etichetta, gli si assegna quella minore.



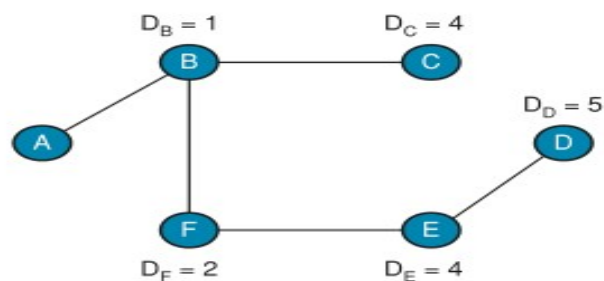
Scegliamo a questo punto di nuovo il percorso migliore, quindi quello passante per F ed aggiorniamo le etichette dei nodi ad esso adiacenti.



Il prossimo nodo da analizzare sarà E dal quale possiamo raggiungere il nodo D, al quale sarà assegnato un'etichetta minore.

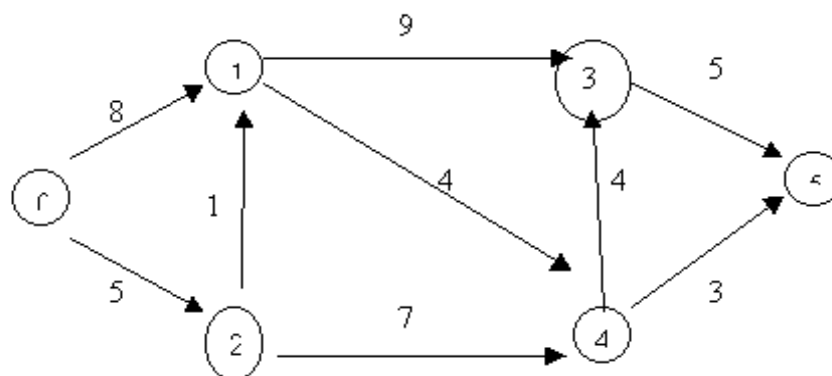


Il nodo rimanente sarà il nodo C al quale però conviene lasciare l'etichetta assegnata perchè un eventuale collegamento con D non porterebbe migliorie. A questo punto le etichette diventano permanenti e otteniamo il seguente albero minimo.



### Esercizio:

Esercizio: trovare il cammino minimo tra 0 e 5 e l'albero di ricoprimento nel seguente grafo orientato:



Soluzione: il cammino minimo è 0-2-1-4-5 con peso 13.

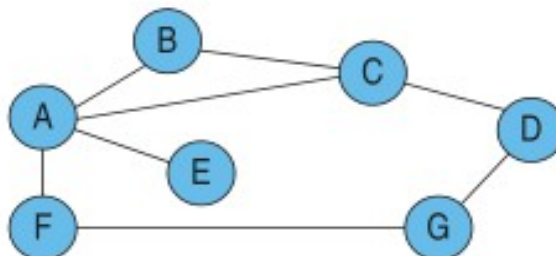
## Algoritmi dinamici

Per quanto possano essere efficienti, performanti e computazionalmente validi, agli algoritmi statici devono essere affiancati gli algoritmi dinamici. Questo accade a causa della continua espansione della rete e della sempre crescente innovazione tecnologica. Gli algoritmi dinamici sono capaci di scoprire i percorsi minimi su una rete e mantenerli costantemente aggiornati.

### Algoritmo di Bellman-Ford:

È un algoritmo dinamico che calcola per ciascun nodo il cammino migliore. Il calcolo viene fatto sulle informazioni ricevute dagli altri router e non conosce la topologia della rete. In pratica ogni nodo conosce solo i nodi adiacenti ad esso. Proviamo a fare un esempio sul seguente grafo ad archi con costo unitario:

	A	B	C	D	E	F	G
A	0	1	1	$\infty$	1	1	$\infty$
B	1	0	1	$\infty$	$\infty$	$\infty$	$\infty$
C	1	1	0	1	$\infty$	$\infty$	$\infty$
D	$\infty$	$\infty$	1	0	$\infty$	$\infty$	1
E	1	$\infty$	$\infty$	$\infty$	0	$\infty$	$\infty$
F	1	$\infty$	$\infty$	$\infty$	$\infty$	0	1
G	$\infty$	$\infty$	$\infty$	1	$\infty$	1	0



Alla prima accensione la tabella di instradamento per B sarà:

Destinazione	Next hop	Costo
A	A	1
C	C	1

A questo punto avviene la comunicazione tra i vari router che acquisiscono informazioni sulle tabelle dei loro vicini. Quindi possono riaggiornare le loro tabelle e la matrice di adiacenza diventa:

	A	B	C	D	E	F	G
A	0	1	1	2	1	1	2
B	1	0	1	2	2	2	3
C	1	1	0	1	2	2	2
D	2	2	1	0	3	2	1
E	1	2	2	3	0	2	3
F	1	2	2	2	2	0	1
G	2	3	2	1	3	1	0

Come avviene questo aggiornamento?

Ogni nodo riceve dai suoi adiacenti le coppie destinazione-costo. Quando un nodo riceve una nuova distanza, se è minore di quella in suo possesso procede con l'aggiornamento, altrimenti la ignora. Ad esempio A invia a B la distanza da E, che poi sommato con la distanza tra A e B attribuisce alla distanza tra B e E il valore 2; il valore che lui ha per ora è  $\infty$  quindi lo sostituisce. La sua tabella diventerà quindi:

Destinazione	Next hop	Costo
A	A	1
C	C	1
E	A	2

Alla fine del processo, cioè quando lo scambio di informazioni si sarà stabilizzato, la tabella di routing per B sarà la seguente:

Destinazione	Next hop	Costo
A	A	1
C	C	1
D	C	2
E	A	2
F	A	2
G	A	3

L'algoritmo è ricorsivo in quanto può essere scritto come:

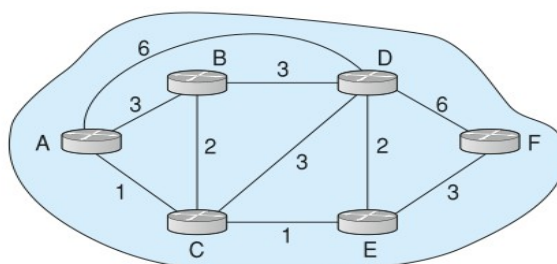
$$d_x(y) = \min_v \{c(x,v) + d_v(y)\}$$

La formula va interpretata così:

- La distanza minima tra x e y è uguale al minimo delle distanze tra le somme dei costi dei nodi adiacenti v e la loro distanza da y. Appunto il calcolo del secondo membro della somma è ottenuto sempre dalla formula stessa!!!.....

Facciamo un esempio.....

Dobbiamo calcolare il percorso minore tra A e F  $d_A(F)$ :



Prendiamo i nodi adiacenti ad A (B,C e D), avranno costi  $c(A,B)=3$ ,  $c(A,C)=1$ ,  $c(A,D)=6$

A questo punto dovremo calcolare le distanze  $d_B(F)$ ,  $d_C(F)$  e  $d_D(F)$  per poter poi applicare la formula:

$$d_A(F) = \min\{c(A,B) + d_B(F), c(A,C) + d_C(F), c(A,D) + d_D(F)\}$$

$$d_A(F) = \min\{3 + d_B(F), 1 + d_C(F), 6 + d_D(F)\}$$

$d_B(F)$ ,  $d_C(F)$  e  $d_D(F)$  verranno calcolati sempre con la stessa formula

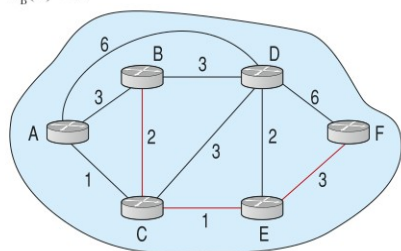
$$d_B(F) = \min\{c(B,C) + d_C(F), c(B,D) + d_D(F)\}$$

$$d_C(F) = \min\{c(C,D) + d_D(F), c(C,E) + d_E(F)\}$$

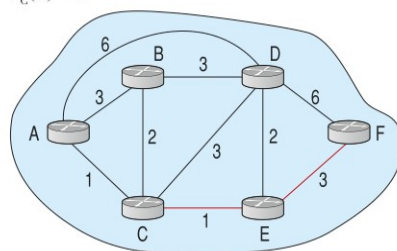
$$d_D(F) = 5, d_E(F) = 3 \text{ banalmente}$$

$$d_C(F) = \min\{3 + 5, 1 + 3\} = 4, d_B(F) = \min\{2 + 4, 3 + 5\}$$

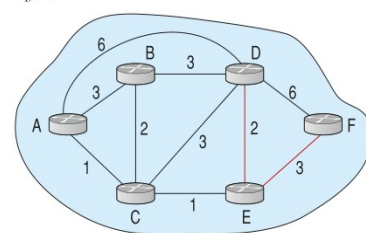
$$d_B(F) = 6$$



$$d_C(F) = 4$$



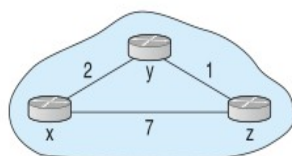
$$d_D(F) = 5$$



$$d_A(F) = \min\{3 + 6, 1 + 4, 6 + 5\} = 5$$

In conclusione potremo dire che A in strada a F tramite C a costo 5

Ora proviamo invece, analizzando un grafo più piccolo, a compilare le tabelle di routing. Prendiamo quindi la seguente rete:



Le tabelle all'inizio (al passo 0) saranno:

Tabella del nodo x

		Costo verso		
		x	y	z
Da	x	0	2	7
	y	∞	∞	∞
	z	∞	∞	∞

		Costo via	
		D <sup>x</sup>	
Dest.	y	2	∞
	z	∞	7

Tabella del nodo y

		Costo verso		
		x	y	z
Da	x	∞	∞	∞
	y	2	0	1
	z	∞	∞	∞

		Costo via	
		D <sup>y</sup>	
Dest.	x	2	∞
	z	∞	1

Tabella del nodo z

		Costo verso		
		x	y	z
Da	x	∞	∞	∞
	y	∞	∞	∞
	z	7	1	0

		Costo via	
		D <sup>z</sup>	
Dest.	x	7	∞
	y	∞	1

Passo 1 per il nodo x:

		Costo via	
Dest.	D <sup>x</sup>	y	z
	y	2	8
	z	3	7

Tabella del nodo x

		Costo verso		
Da		x	y	z
	x	0	2	3
	y	2	0	1
	z	7	1	0

$$d_x(z) = \min\{c(x,y) + D_y(z), c(x,z) + d_z(z)\} = \min\{2+1, 7+0\} = 3$$

→ x invia a z tramite y a costo 3

$$d_x(y) = \min\{c(x,y) + D_y(y), c(x,z) + d_z(y)\} = \min\{2+0, 7+1\} = 2$$

→ x invia a y tramite y a costo 2

Passo 1 per il nodo y:

		Costo via	
Dest.	D <sup>y</sup>	x	z
	x	2	8
	z	9	1

Tabella del nodo y

		Costo verso		
Da		x	y	z
	x	0	2	7
	y	2	0	1
	z	7	1	0

$$d_y(x) = \min\{c(y,z) + d_z(x), c(y,x) + d_x(x)\} = \min\{1+7, 2+0\} = 2$$

→ y invia a x tramite x a costo 2

$$d_y(z) = \min\{c(y,z) + d_z(z), c(y,x) + d_x(z)\} = \min\{1+0, 2+7\} = 1$$

→ y invia a z tramite y a costo 1

Passo 1 per il nodo z:

		Costo via	
Dest.	D <sup>z</sup>	x	y
	x	7	3
	y	9	1

Tabella del nodo z

		Costo verso		
Da		x	y	z
	x	0	2	7
	y	2	0	1
	z	3	1	0

$$d_z(x) = \min\{c(z,y) + d_y(x), c(z,x) + d_x(x)\} = \min\{1+2, 7+0\} = 3$$

→ z invia a x tramite y a costo 3

$$d_z(y) = \min\{c(z,y) + d_y(y), c(z,x) + d_x(y)\} = \min\{1+0, 7+2\} = 1$$

→ z invia a y tramite y a costo 1

Con un ulteriore passo l'algoritmo aggiorna le tabelle, se ha in memoria valori migliori. Se in caso abbia determinato ulteriori percorsi migliori, invia gli aggiornamenti:



Tabella del nodo x

		Costo verso		
		x	y	z
Da	x	0	2	3
	y	2	0	1
	z	3	1	0

Tabella del nodo y

		Costo verso		
		x	y	z
Da	x	0	2	3
	y	2	0	1
	z	3	1	0

Tabella del nodo z

		Costo verso		
		x	y	z
Da	x	0	2	3
	y	2	0	1
	z	3	1	0

A questo punto le tabelle convergono, quindi il sistema è stabilizzato. Le conseguenti tabelle di routing saranno:

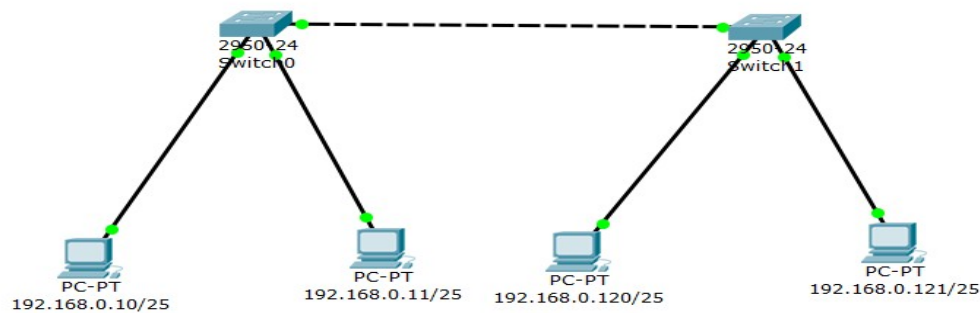
Destinazione/ Network	Next Hop	Costo/ Distance
Y	y	2
Z	y	3

Destinazione/ Network	Next Hop	Costo/ Distance
X	x	2
Z	z	1

Destinazione/ Network	Next Hop	Costo/ Distance
X	y	3
Y	y	1

## Laboratorio 1 Packet Tracer

Costruiamo due segmenti della stessa subnet con due switch differenti. Colleghiamo gli switch con un cavo copper cross-over. Possiamo trovarci a costruire una rete così perché ad esempio vogliamo usare la stessa subnet in posti fisicamente diversi.

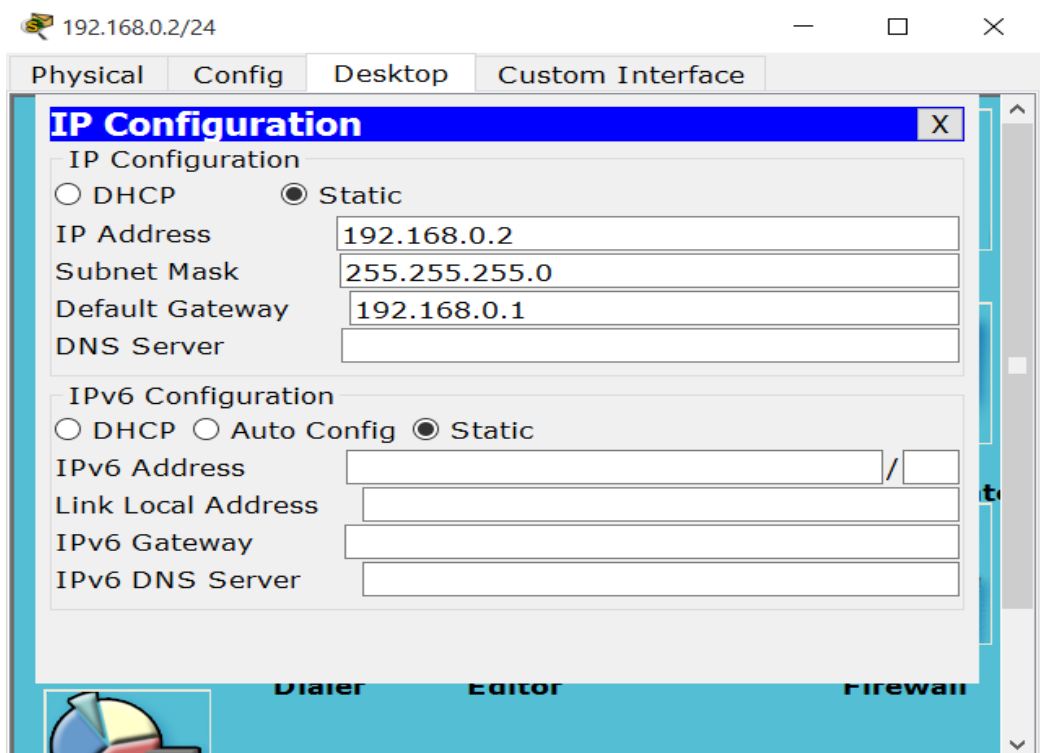


## Laboratorio 2 Packet Tracer

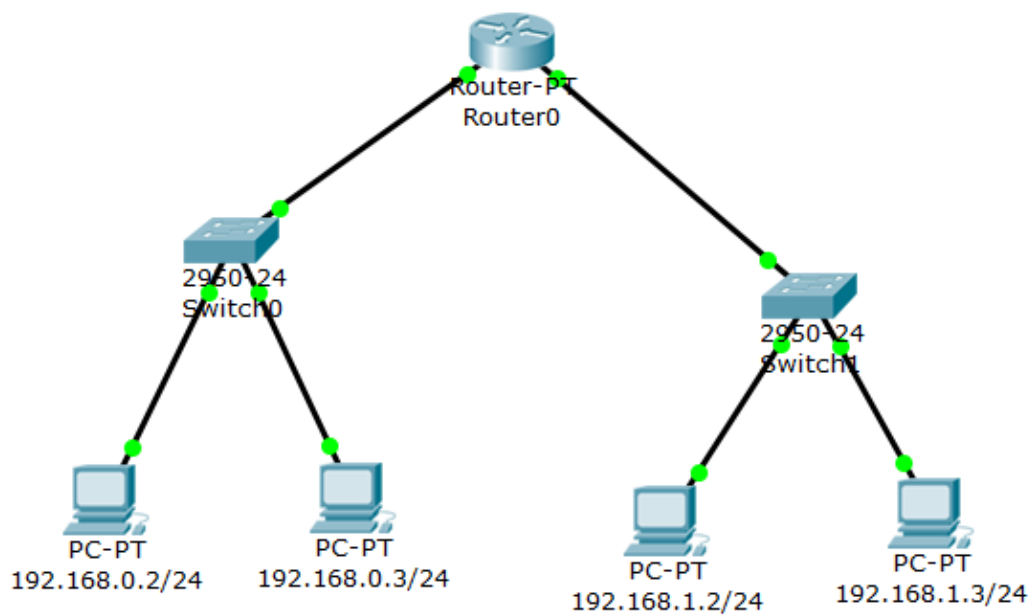
Vogliamo costruire due subnet e metterle in comunicazione. Configuriamo le due subnet 192.168.0.0/24 e 192.168.1.0/24 e le colleghiamo a due switch come già abbiamo visto in precedenza.

Dobbiamo configurare il default gateway che in sostanza indica su quale interfaccia inviare i pacchetti che risultano essere indirizzati a una rete diversa.

Stiamo attenti a configurare questa volta il default gateway quando assegniamo l'indirizzo IP a ogni PC:



Con 192.168.0.1 per la prima rete e 192.168.1.1 per la seconda rete.  
La configurazione della rete sarà la seguente:



Router0

Physical
 Config
 CLI

**GLOBAL**
Settings
Algorithm Settings
**ROUTING**
Static
RIP
**INTERFACE**
FastEthernet0/0
FastEthernet1/0
Serial2/0
Serial3/0
FastEthernet4/0
FastEthernet5/0

### FastEthernet0/0

Port Status ☒ On

Bandwidth ☐ 100 Mbps ☐ 10 Mbps ☒ Auto

Duplex ☐ Half Duplex ☒ Full Duplex ☒ Auto

MAC Address 0001.6465.96CD

IP Configuration

IP Address 192.168.0.1

Subnet Mask 255.255.255.0

Tx Ring Limit 10

 Equivalent IOS Commands
 

```

Router>enable
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface FastEthernet0/0
Router(config-if)#
    
```

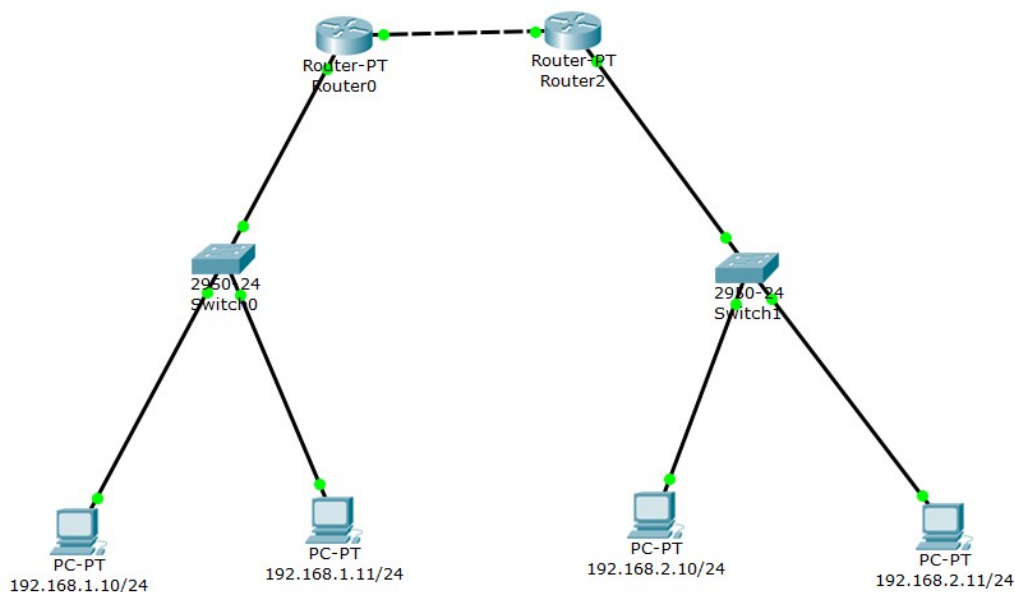
Ora andiamo a configurare le porte del router che sono state collegate agli switch.

Da config scegliere la porta da configurare FastEthernet0/0 con indirizzo IP 192.168.0.1 e la netmask 255.255.255.0 (come si vede nella figura precedente). Fare lo stesso sulla FastEthernet1/0. A questo punto non sarà necessario configurare le tabelle di routing come vedremo più avanti. In pratica i pacchetti che vanno da una rete all'altra non necessitano di essere indirizzati a un next hop, cioè su un altro router. In pratica utilizzano un indirizzamento "diretto".

## Laboratorio 3 Packet Tracer

Questa volta vogliamo configurare una rete con due router. Sempre configuriamo le due subnet come in precedenza.

Collegiamo come prima switch a router e poi i due router tra loro.



Collegare i due router con il cavo Copper cross-over.

Configurare le porte collegate agli switch come in precedenza. Le porte di collegamento tra i due router devono avere l'indirizzo IP di un'altra rete a parte come la 192.168.10.0.

Router0

Physical Config CLI

GLOBAL

Settings

Algorithm Settings

ROUTING

Static

RIP

INTERFACE

FastEthernet0/0

FastEthernet1/0

Serial2/0

Serial3/0

FastEthernet4/0

FastEthernet5/0

FastEthernet1/0

Port Status ☒ On  
Bandwidth ☒ 100 Mbps ☐ 10 Mbps ☒ Auto  
Duplex ☐ Half Duplex ☒ Full Duplex ☒ Auto  
MAC Address 0010.1141.17AC  
IP Configuration  
IP Address 192.168.10.1  
Subnet Mask 255.255.255.0  
Tx Ring Limit 10

Equivalent IOS Commands

```
Router>enable
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface FastEthernet1/0
Router(config-if)#
```

A questo punto si devono configurare le tabelle di routing.

Sempre da config andare su static sul primo router e impostare che per andare sulla rete 192.168.2.0 bisogna andare su 192.168.10.2. Cliccare su Add.

Physical Config CLI

GLOBAL

Settings

Algorithm Settings

ROUTING

Static

RIP

INTERFACE

FastEthernet0/0

Static Routes

Network 192.168.2.0  
Mask 255.255.255.0  
Next Hop 192.168.10.2

Add

Dovrebbe comparire:

Network Address

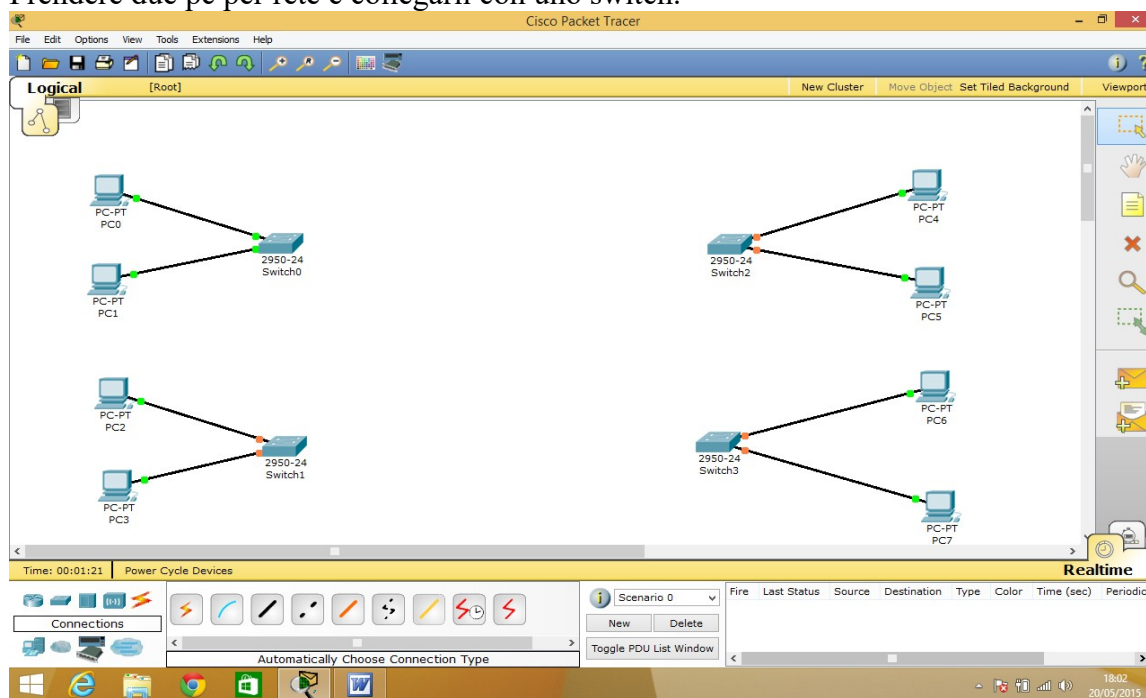
192.168.2.0/24 via 192.168.10.2

Remove

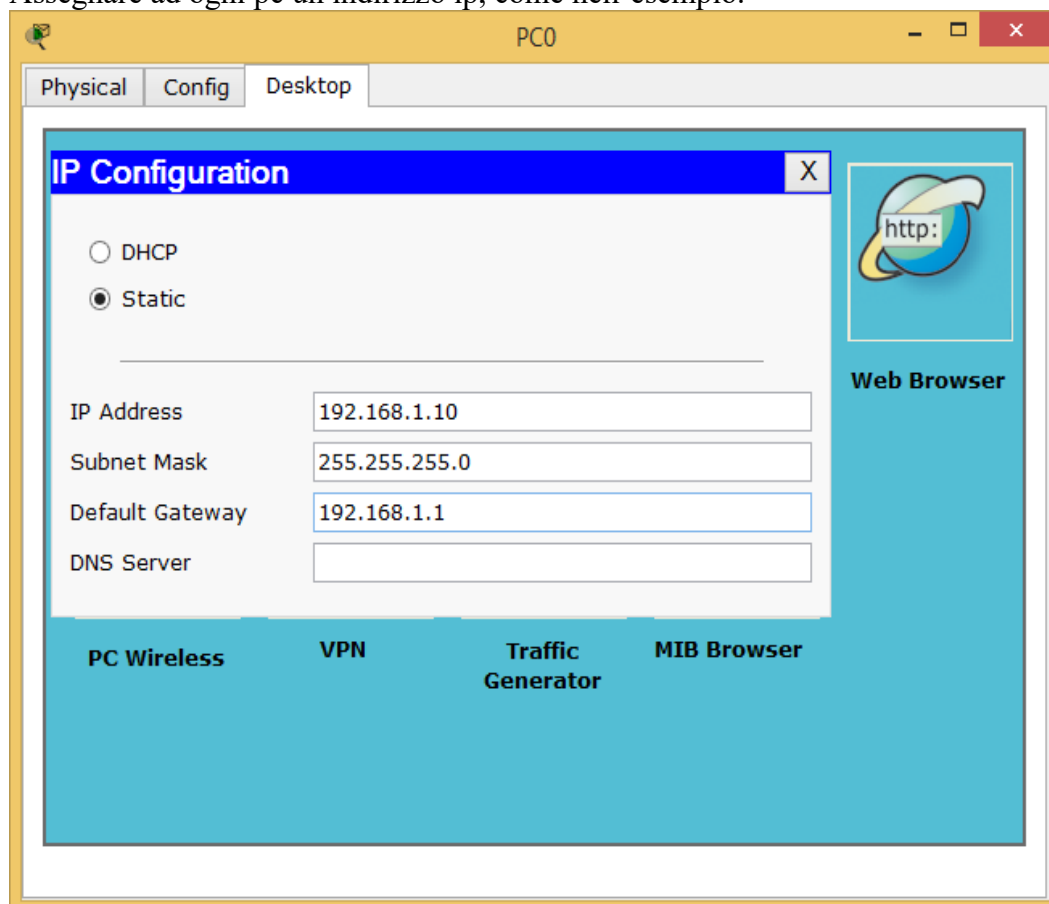
Fare l'opposto sull'altro router.

# Laboratorio 4 Packet Tracer

Prendere due pc per rete e collegarli con uno switch.



Assegnare ad ogni pc un indirizzo ip, come nell'esempio.

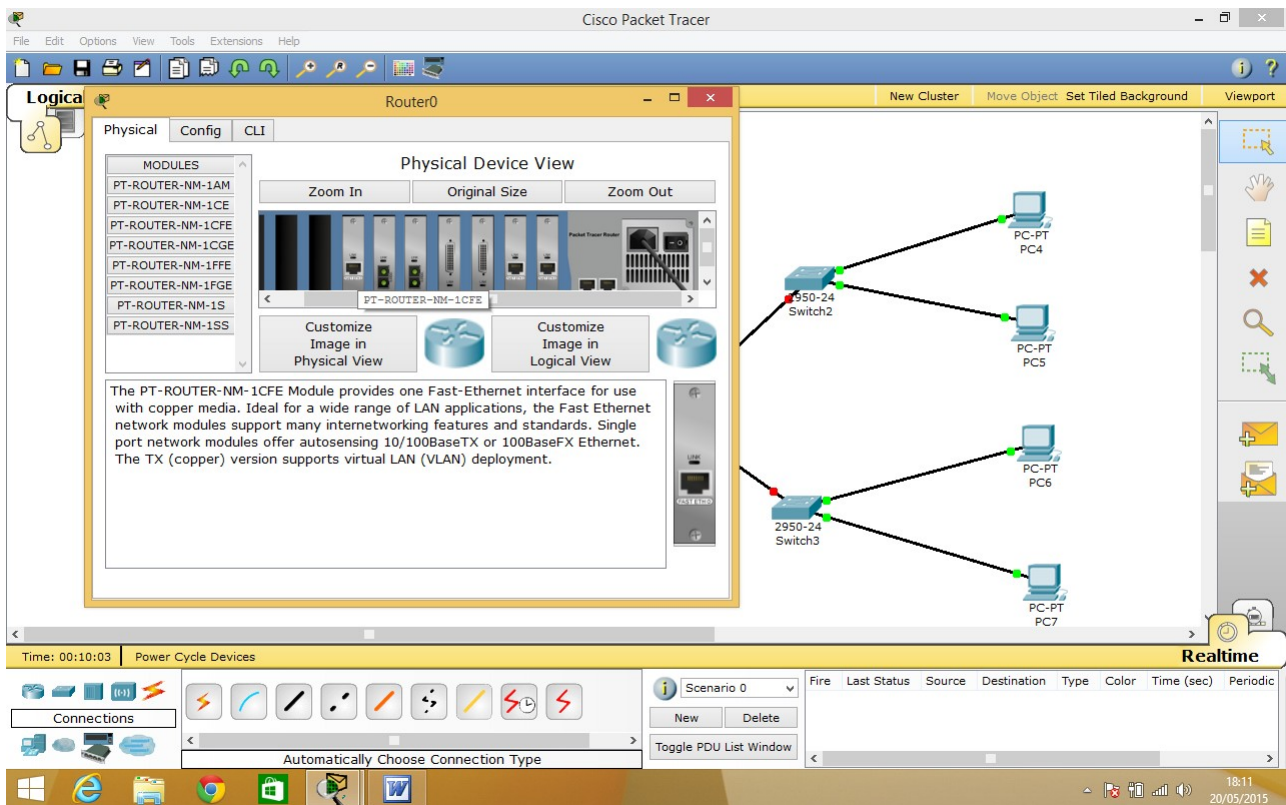


Scrivere le etichette.

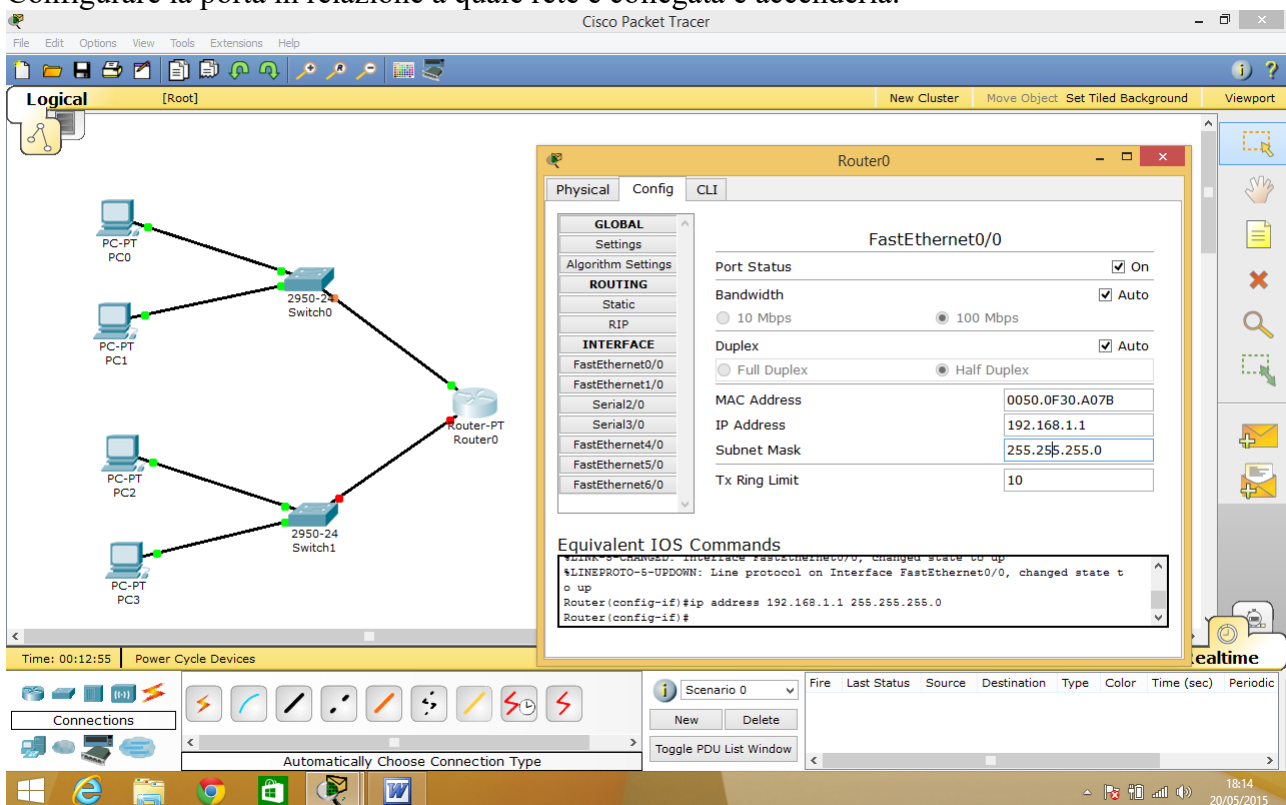
Prendere due router generic-PT.

Entrare nella configurazione fisica del router e spegnere. Aggiungere una porta fast-ethernet in tutte e due i router.





Configurare la porta in relazione a quale rete è collegata e accenderla.



Collegare i due router tramite le porte aggiunte.

Configurare le due porte come appartenenti alla stessa rete (ad esempio 192.168.10.1 e 192.168.10.2). Accendere le porte.

Router0

PhysicalConfigCLI

GLOBAL

Settings

Algorithm Settings

ROUTING

Static

RIP

INTERFACE

FastEthernet0/0

FastEthernet1/0

Serial2/0

Serial3/0

FastEthernet4/0

FastEthernet5/0

FastEthernet6/0

FastEthernet6/0

FastEthernet6/0

Port Status

Bandwidth

Duplex

MAC Address

IP Address

Subnet Mask

Tx Ring Limit

☒ On

☒ Auto  
☐ 10 Mbps    ☒ 100 Mbps

☒ Auto  
☐ Full Duplex    ☒ Half Duplex

0090.2BB5.D7BD

192.168.10.1

255.255.255.0

10

Equivalent IOS Commands

```

Router(config-if)#ip address 192.168.10.1 255.255.255.0
Router(config-if)#no shutdown

Router(config-if)#
%LINK-5-CHANGED: Interface FastEthernet6/0, changed state to up

```

Andare su “static” nel router e configure le tabelle di routine statico.  
Ad esempio sul router di sinistra:

Router0

PhysicalConfigCLI

GLOBAL

Settings

Algorithm Settings

ROUTING

Static

RIP

INTERFACE

FastEthernet0/0

FastEthernet1/0

Serial2/0

Serial3/0

FastEthernet4/0

FastEthernet5/0

FastEthernet6/0

Static Routes

Network

Mask

Next Hop

192.168.3.0

255.255.255.0

192.168.10.2

Add

Network Address

Remove

Equivalent IOS Commands

```

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet6/0, changed state to up
Router(config-if)#exit
Router(config)#

```

Dopo aver inserito gli indirizzi premere “add”.

Router0

Physical Config CLI

**GLOBAL**

Settings

Algorithm Settings

**ROUTING**

Static

RIP

**INTERFACE**

FastEthernet0/0

FastEthernet1/0

Serial2/0

Serial3/0

FastEthernet4/0

FastEthernet5/0

FastEthernet6/0

**Static Routes**

Network	192.168.4.0
Mask	255.255.255.0
Next Hop	192.168.10.2

Add

Network Address

192.168.3.0/24 via 192.168.10.2

192.168.4.0/24 via 192.168.10.2

Remove

**Equivalent IOS Commands**

```
Router(config-if)#exit
Router(config)#ip route 192.168.3.0 255.255.255.0 192.168.10.2
Router(config)#ip route 192.168.4.0 255.255.255.0 192.168.10.2
Router(config)#
```

Mentre per il router di destra:

Router1

Physical Config CLI

**GLOBAL**

Settings

Algorithm Settings

**ROUTING**

Static

RIP

**INTERFACE**

FastEthernet0/0

FastEthernet1/0

Serial2/0

Serial3/0

FastEthernet4/0

FastEthernet5/0

FastEthernet6/0

**Static Routes**

Network	192.168.1.0
Mask	255.255.255.0
Next Hop	192.168.10.1

Add

Network Address

Remove

**Equivalent IOS Commands**

```
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet6/0, changed state to up
Router(config-if)#exit
Router(config)#
```

Router1

Physical Config CLI

**GLOBAL**

Settings

Algorithm Settings

**ROUTING**

Static

RIP

**INTERFACE**

FastEthernet0/0

FastEthernet1/0

Serial2/0

Serial3/0

FastEthernet4/0

FastEthernet5/0

FastEthernet6/0

**Static Routes**

Network	192.168.2.0
Mask	255.255.255.0
Next Hop	192.168.10.1

Add

Network Address

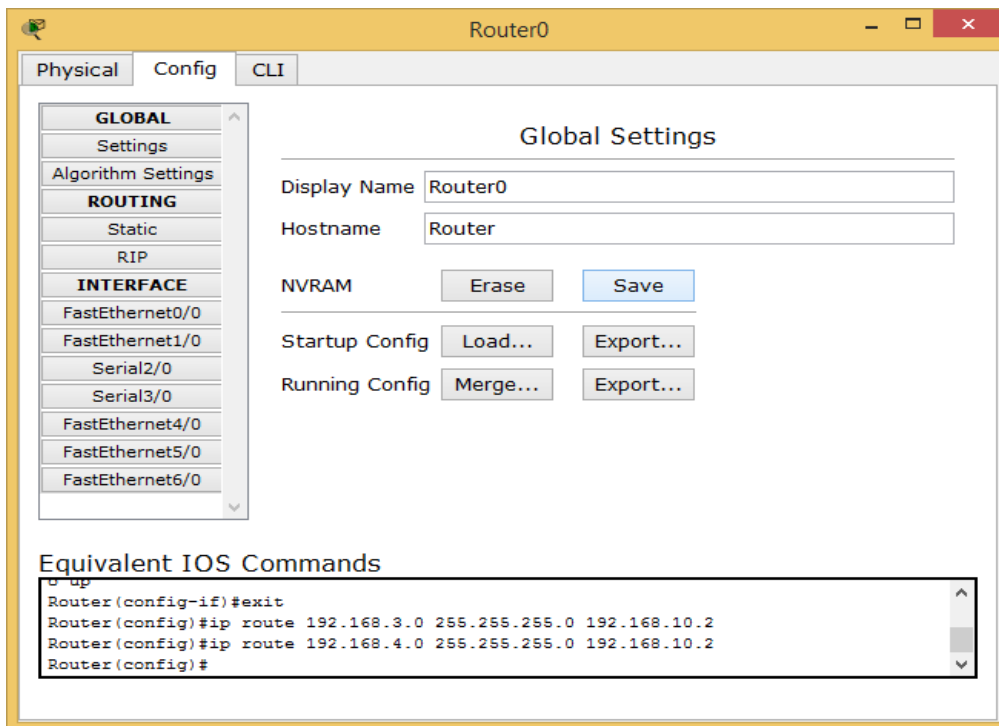
192.168.2.0/24 via 192.168.10.1

Remove

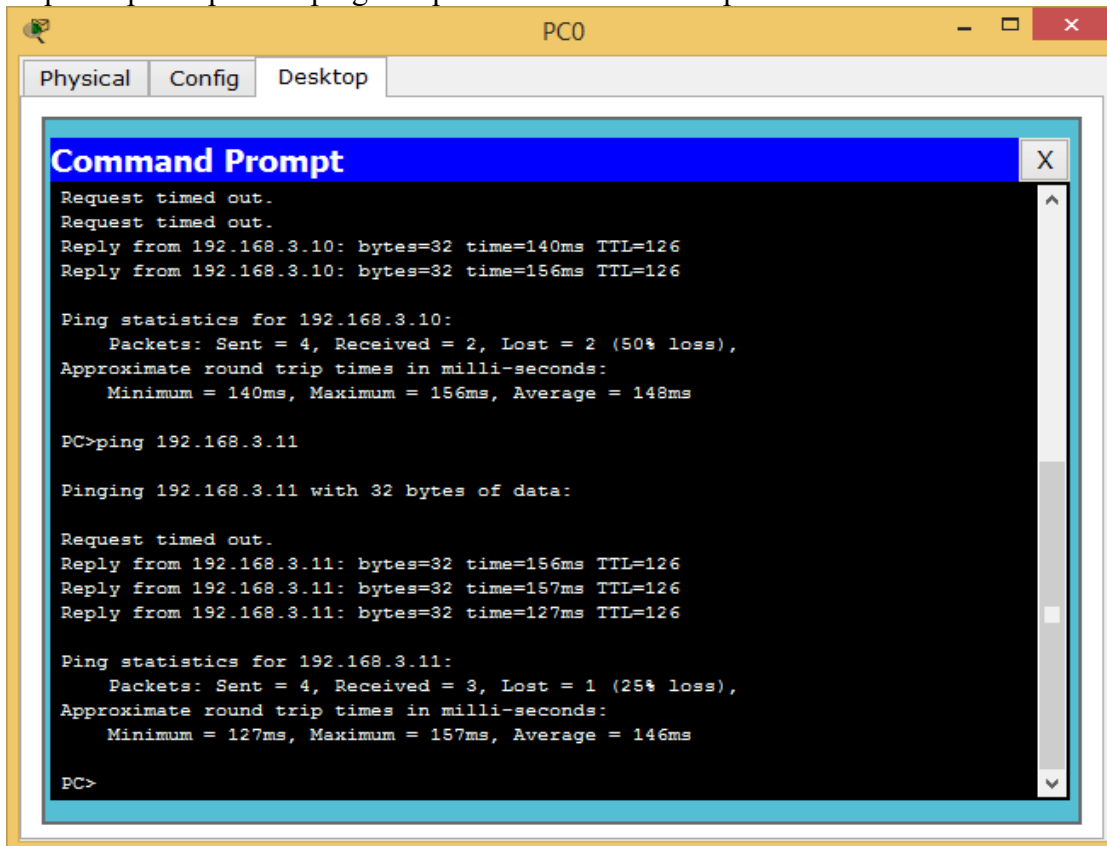
**Equivalent IOS Commands**

```
o up
Router(config-if)#exit
Router(config)#ip route 192.168.2.0 255.255.255.0 192.168.10.1
Router(config)#
```

Andare su “global setting” su entrambi e salvare:



A questo punto prova a pingare i pc tra di loro. Ad esempio dal PC0 al PC7 e viceversa.



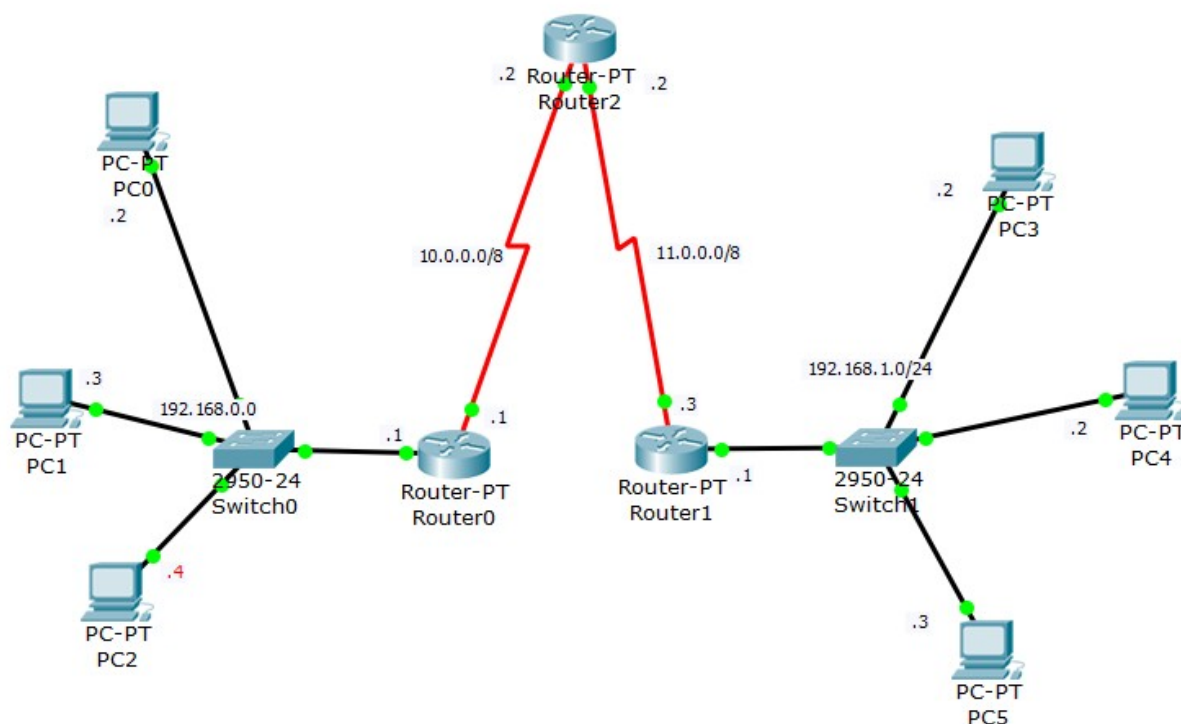
## Laboratorio 5 Packet Tracer

Configuriamo una rete con un router centrale e altri due collegati a parte a due reti diverse. Questa volta li colleghiamo attraverso la porta seriale e usiamo il RIP (Router Information Protocol), cioè non dovremo assegnare le route in maniera statica, ma basterà assegnare le reti adiacenti per ogni route e poi le tabelle si autocompleteranno in maniera dinamica attraverso un algoritmo di tipo Distance Vector (Bellman-Ford). Quindi la rete di sinistra sarà 192.168.0.0/24 e quella di destra 192.168.1.0/24; mentre invece quelle che collegheranno i tre router saranno 10.0.0.0/8 e 11.0.0.0/8.

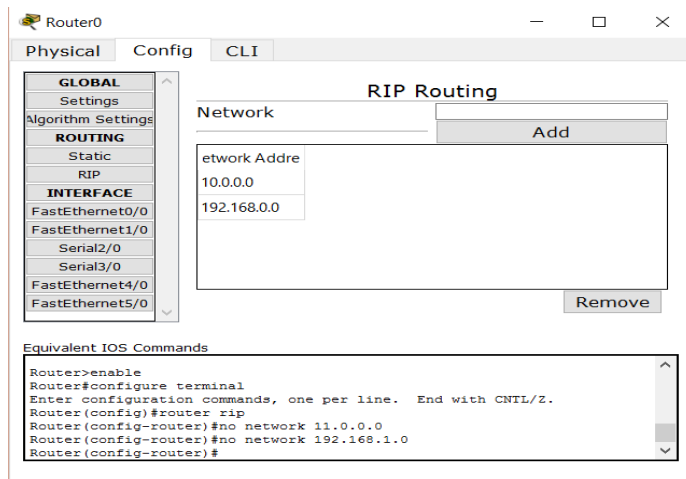
Facciamo attenzione questa volta a collegare i due router che sono collegati direttamente alle due subnet. Infatti il Router0 e il Router1 vanno collegati al Router2 attraverso la porta seriale, usando il cavo serial DCE. I dispositivi DCE forniscono il clock (timing interno), mentre i dispositivi DTE sincronizzano l'orologio fornito (timing esterno).

Generalmente il DTE è un terminale o un personal computer, mentre il DCE è un modem.

Stiamo attenti a collegare i cavi prima al router in cima (che potrebbe essere, con molta fantasia, il nostro ISP) e poi agli altri due in maniera tale da definirlo come terminale DCE. In questo sarà possibile infatti definire il clock. Il DTE adotterà il clock settato sul DCE.



Logicamente ora dovremo dire al Router0 cosa farne dei pacchetti indirizzati alla rete 192.168.1.0/24 e viceversa al Router1 cosa farne dei pacchetti 192.168.1.0/24. Lo potremo fare in modalità statica come fatto finora... Ma ora useremo il RIP. Quindi andiamo a cliccare su RIP (sotto static) e indichiamo le reti adiacenti:



Facciamo lo stesso per gli altri due router e proviamo a pingare....

## Laboratorio 6 Packet Tracer

Impariamo a usare ora la CLI (Command Line Interface).

All'apertura scrivere *no* e invio... (si possono impostare alcune cose, ma non approfondisco ora..).

Per abilitare la configurazione scrivere *enable* (o anche solo *en*) e dare invio. Poi *configure terminal* (anche *configure t*) e invio. A questo punto possiamo impostare ad esempio l'indirizzo IP di un'interfaccia:

```
interface FastEthernet0/0
```

```
ip address 192.168.0.1 255.255.255.0
```

Poi accendiamo l'interfaccia con *no shutdown*.

Per uscire usare *exit* (o anche solo *ex*), se non vi ricordate un comando iniziate a scriverlo e poi basterà usare il tab per l'autocompletamento.

Per salvare il tutto bisogna tornare alla fase enable ("Router#") e quindi scrivere *copy running-config startup-config* e dare invio due volte.

Da config possiamo anche impostare le tabelle di route in modalità statica:

```
ip route 192.168.0.0 255.255.255.0 10.0.0.1
```

Come impostarle in modalità dinamica lo vedremo nel prossimo esercizio.

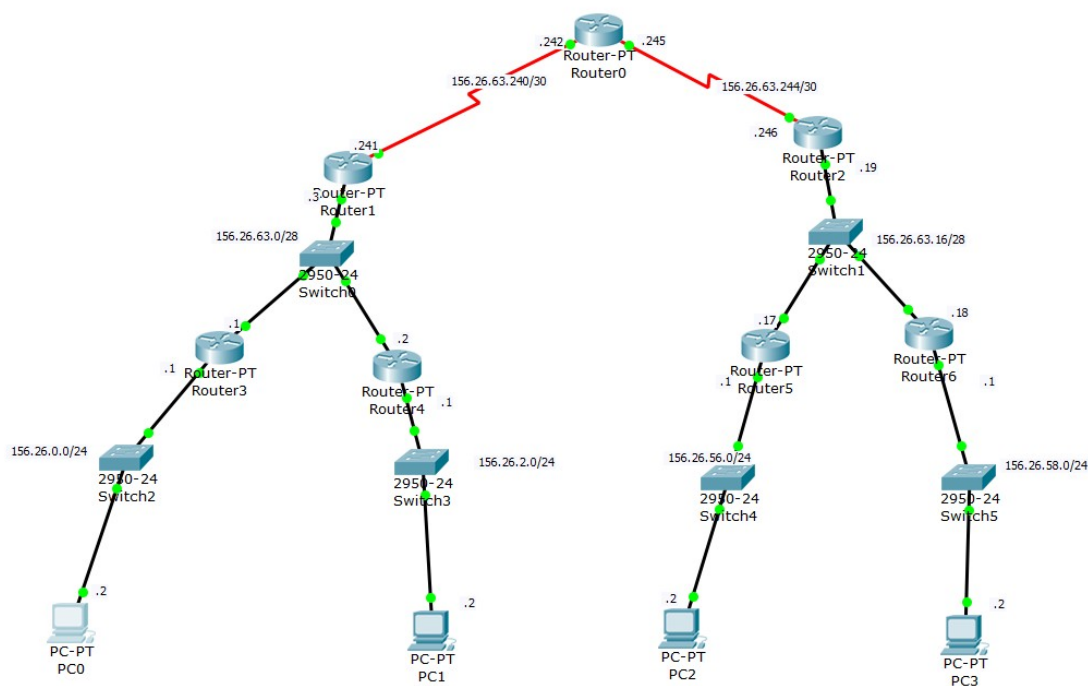
## Laboratorio 7 Packet Tracer

Vogliamo simulare il partizionamento della rete 156.26.0.0 come fatto in un esercizio precedente, riportando la struttura di rete sul packet tracer.

Riportiamo per ora solo una parte di quel partizionamento. Questa situazione risulta differente dalle configurazioni precedenti in quanto la netmask è variabile.

Nota: l'elaborazione dell'esercizio è particolarmente difficile quindi vi chiedo di armarvi di pazienza.....

Usiamo due reti punto-punto per collegare il router centrale con altri due router e poi andiamo risalendo l'albero che abbiamo disegnato nell'esercizio svolto....



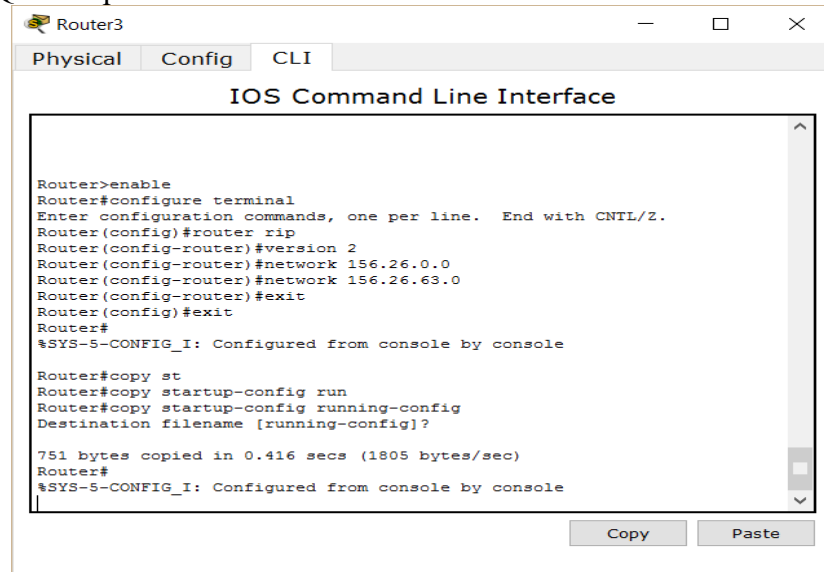
Come si vede le componenti usate sono parecchie e quindi, dopo aver configurato nel giusto modo tutte le interfacce, dovremo andar a configurare le tabelle di route. Se usassimo un routing statico come fatto in precedenza dovremmo armarci di pazienza e configurare 12 route per ogni router... Onestamente sarebbe una follia... Ci sono due strade che a questo punto possiamo seguire: una è di



usare il “riepilogo di route”, l'altra di usare il “routing dinamico”. A mio avviso il riepilogo di route potrebbe ridurre le route da inserire ma rimarrebbe sempre una soluzione molto complicata se dovessimo configurare 100 router.

Ecco perché in questo caso usiamo il RIP.

Per fare ciò useremo la CLI (Command Line Interface, non possiamo farlo da interfaccia grafica come prima....). Quindi apriamo la CLI del Router3:



```
Router3
Physical Config CLI
IOS Command Line Interface

Router>enable
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#router rip
Router(config-router)#version 2
Router(config-router)#network 156.26.0.0
Router(config-router)#network 156.26.63.0
Router(config-router)#exit
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console

Router#copy st
Router#copy startup-config run
Router#copy startup-config running-config
Destination filename [running-config]?

751 bytes copied in 0.416 secs (1805 bytes/sec)
Router#
%SYS-5-CONFIG_I: Configured from console by console
```

Come si vede prima entriamo nella configurazione dinamica e poi impostiamo la versione 2. Successivamente dovremo dare solo le due reti che sono adiacenti alle interfacce del router in questione. Ripetiamo questa configurazione per tutti i router e proviamo a pingare....