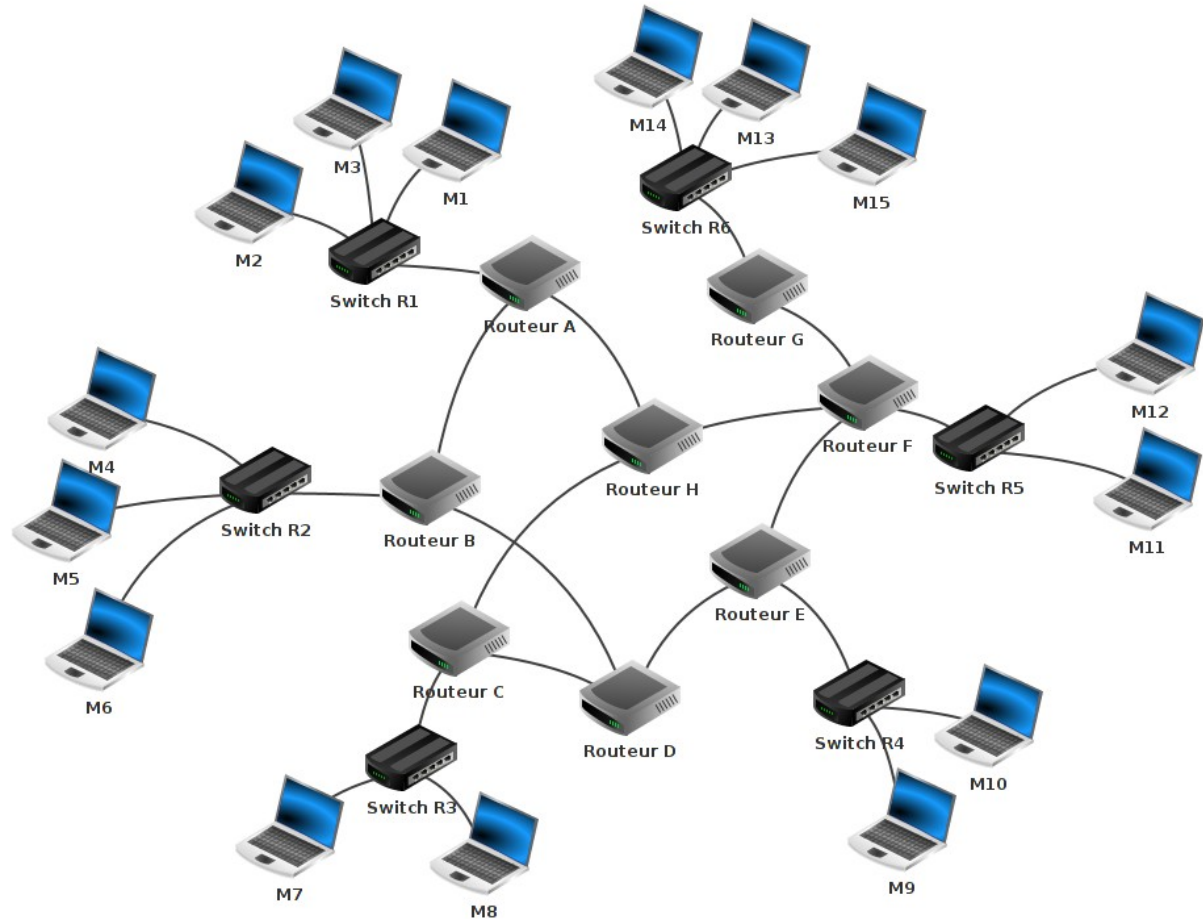


# Introduction

Représenter ce réseau sous la forme d'un **graphe** où les routeurs sont les sommets.

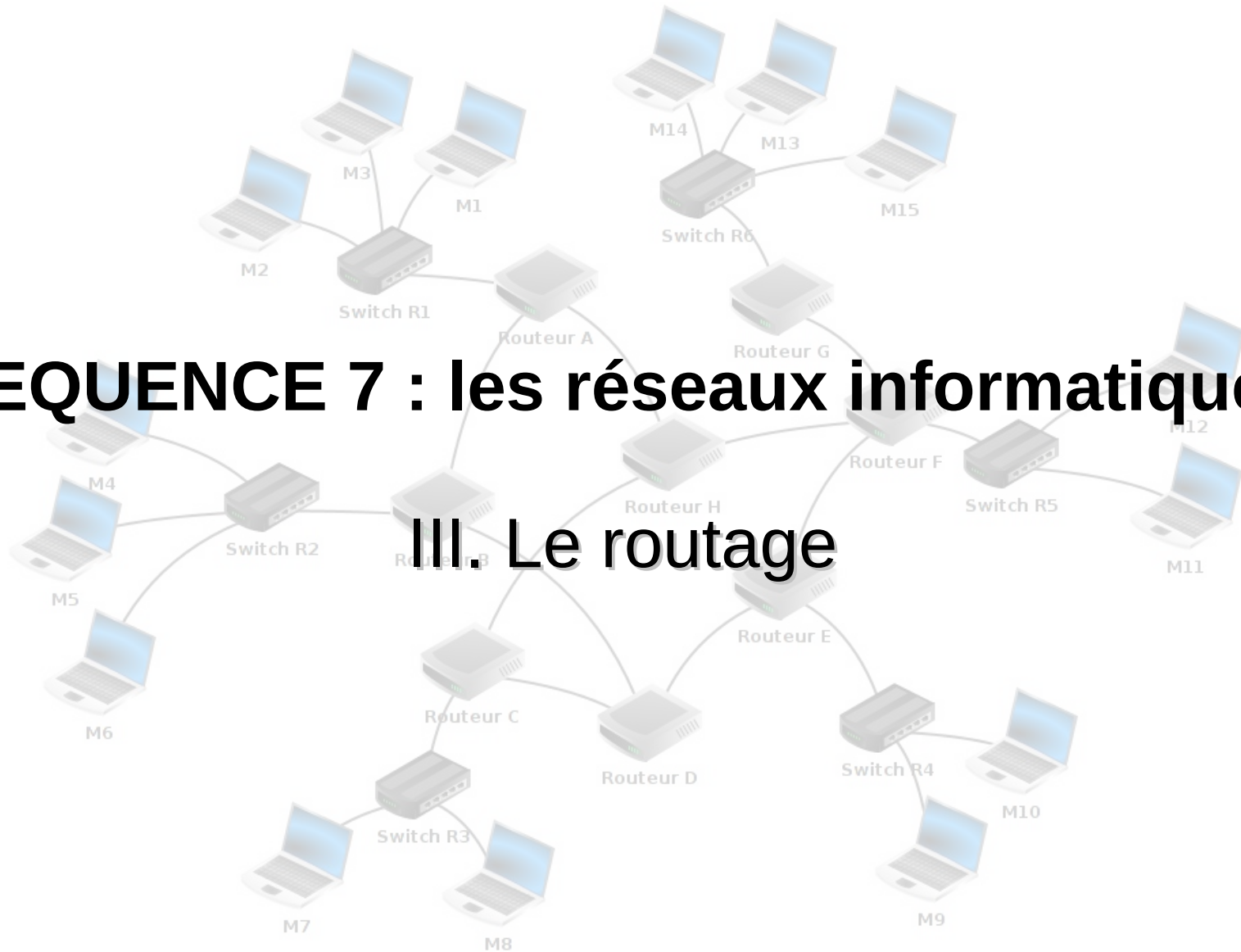
Comment peut-on le qualifier ?

Quels chemins permettent de relier M1 à M11 ?



# SEQUENCE 7 : les réseaux informatiques

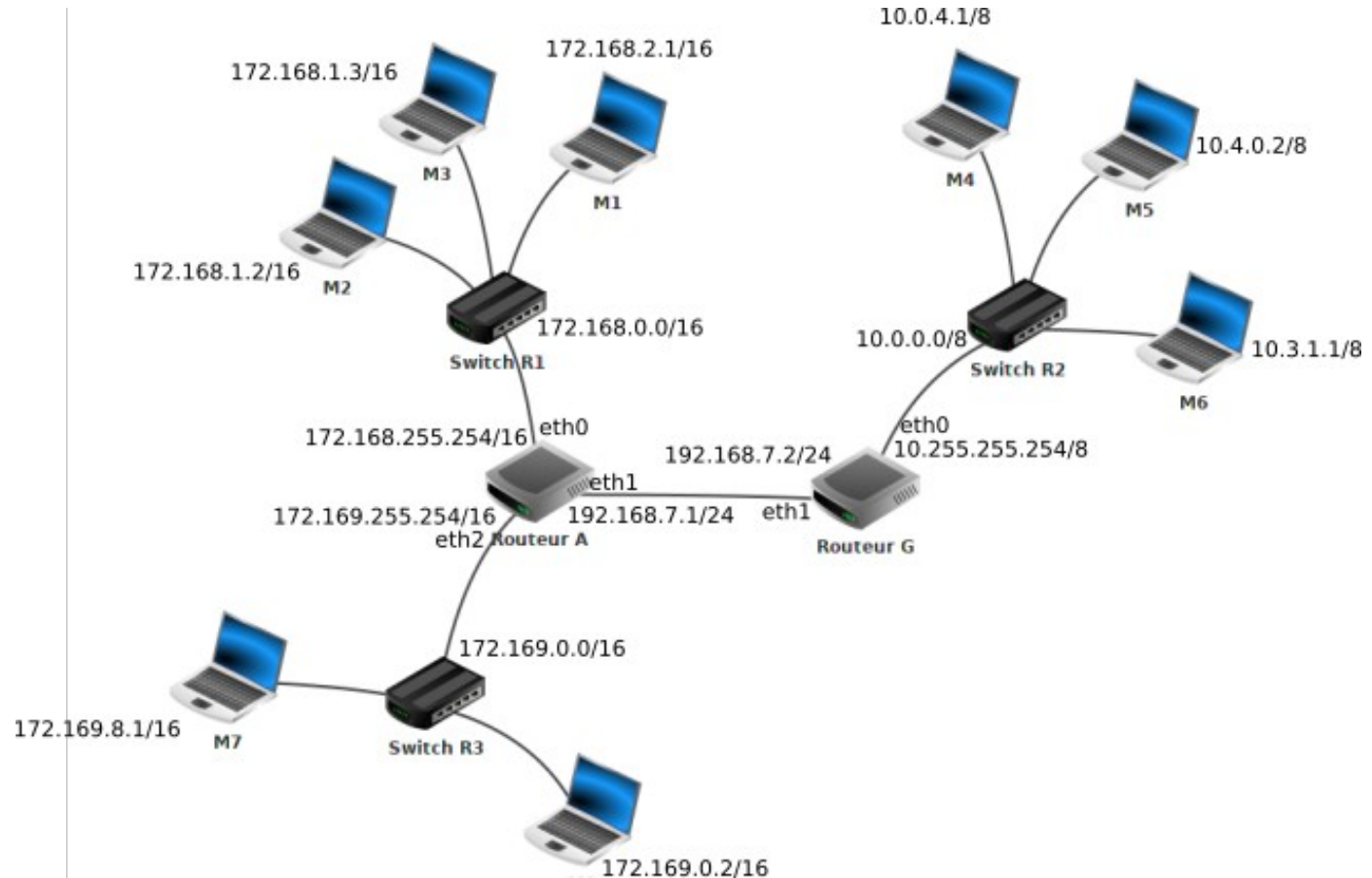
## III. Le routage



# Communication entre sous-réseaux

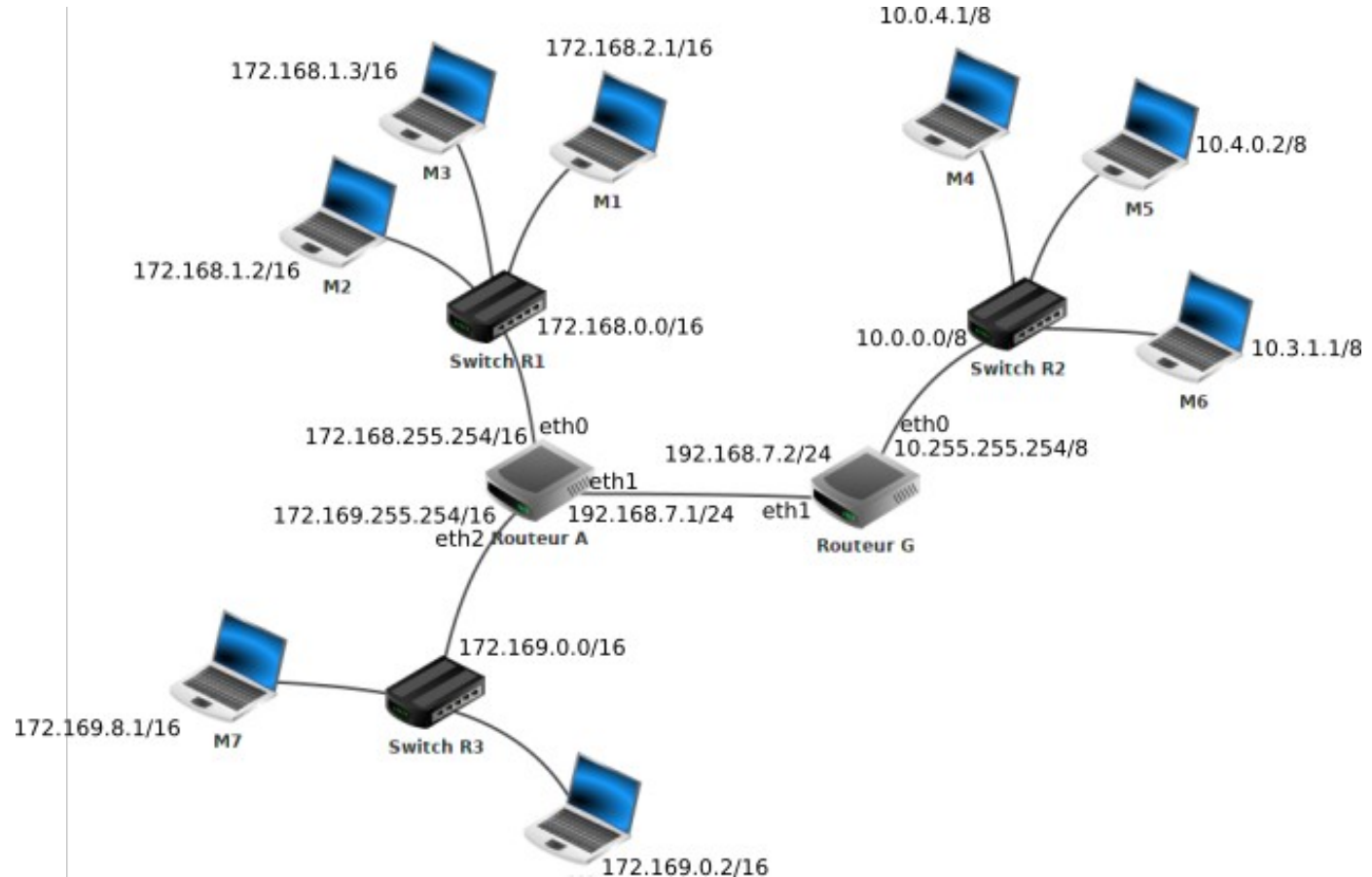
1. Identifier les différents sous-réseaux.

2. Vérifier que les adresses des machines concordent.



# Communication entre sous-réseaux

Les routeurs servent à guider les paquets d'un sous-réseau à un autre, grâce à leurs **tables de routage**.



# Initialisation des tables de routage

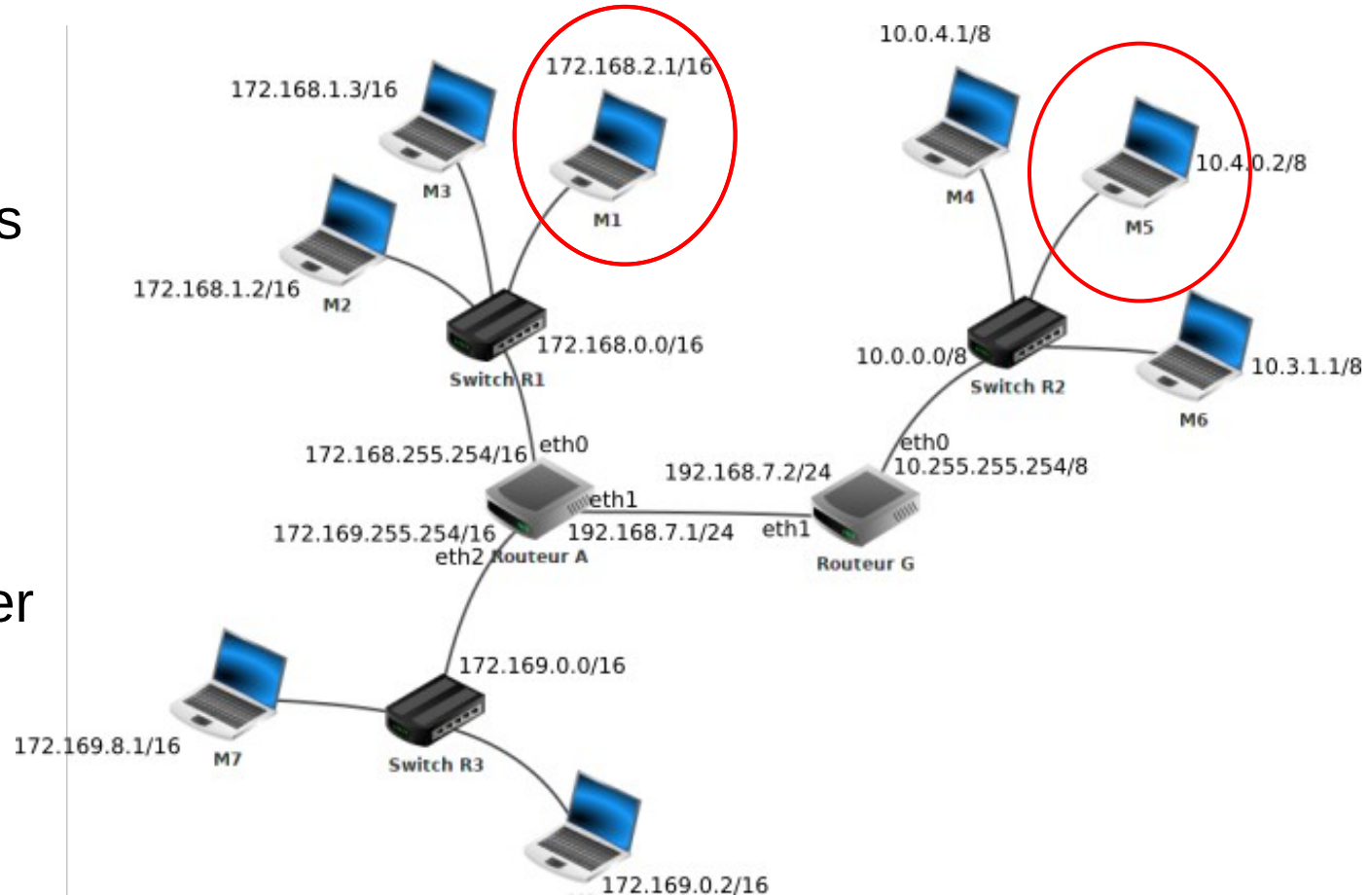
Les routeurs A et G commencent par initialiser leur table avec les informations sur les réseaux auxquels ils sont directement connectés :

Routeur	Réseau	Moyen de l'atteindre
A	172.168.0.0	eth0
	192.168.7.0	eth1
	172.169.0.0	eth2
G	10.0.0.0	eth0
	192.168.7.0	eth1

# Communication entre sous-réseaux

Cette table ne suffit pas  
pour réussir à  
communiquer dans  
l'ensemble du réseau.

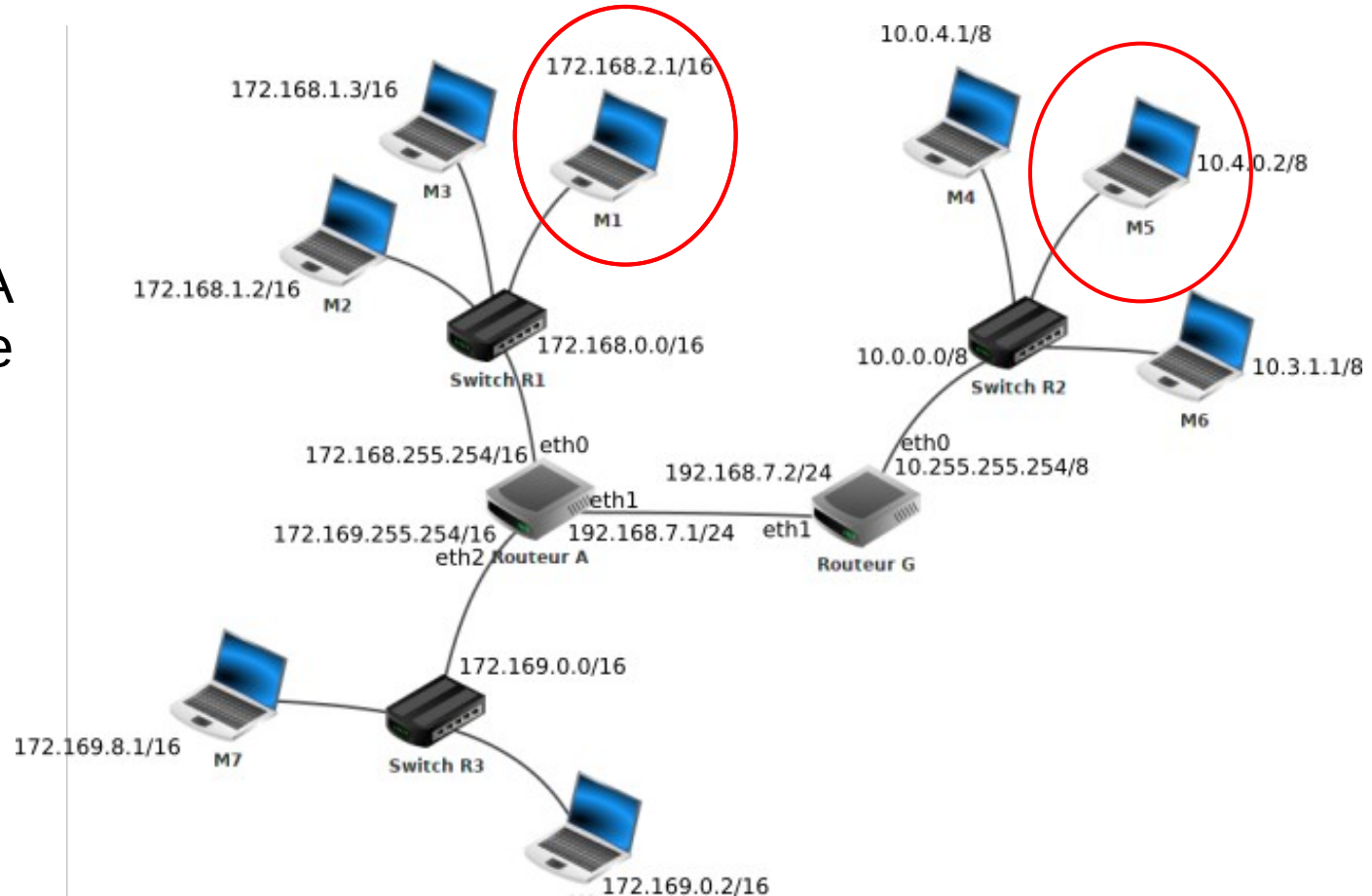
Que se passerait-il si  
**M1** voulait communiquer  
avec **M5** ?



# Communication entre sous-réseaux

On pourrait finir de remplir les tables de routage des routeurs A et G à la main : c'est le *routage statique*.

Plutôt, on utilise des **protocoles de routage** : c'est le *routage dynamique*.



# RIP

**Routing Information Protocol (RIP)** est basé sur deux règles :

- L'*envoi périodique* de la table de routage de chaque routeur aux routeurs de ses réseaux locaux.
- Un *décompte* du nombre de routeurs (= de sauts) traversés qui est la **métrique** permettant de déterminer le plus court chemin pour atteindre un destinataire.



# Table de routage avec RIP

Voici la table de notre réseau, après application de RIP :

Routeur	Réseau	Moyen de l'atteindre	Métrique
A	172.168.0.0	eth0	0
	192.168.7.0	eth1	0
	172.169.0.0	eth2	0
G	10.0.0.0	192.168.7.2	1
	10.0.0.0	eth0	0
	192.168.7.0	eth1	0
	172.168.0.0	192.168.7.1	1
	172.169.0.0	192.168.7.1	1

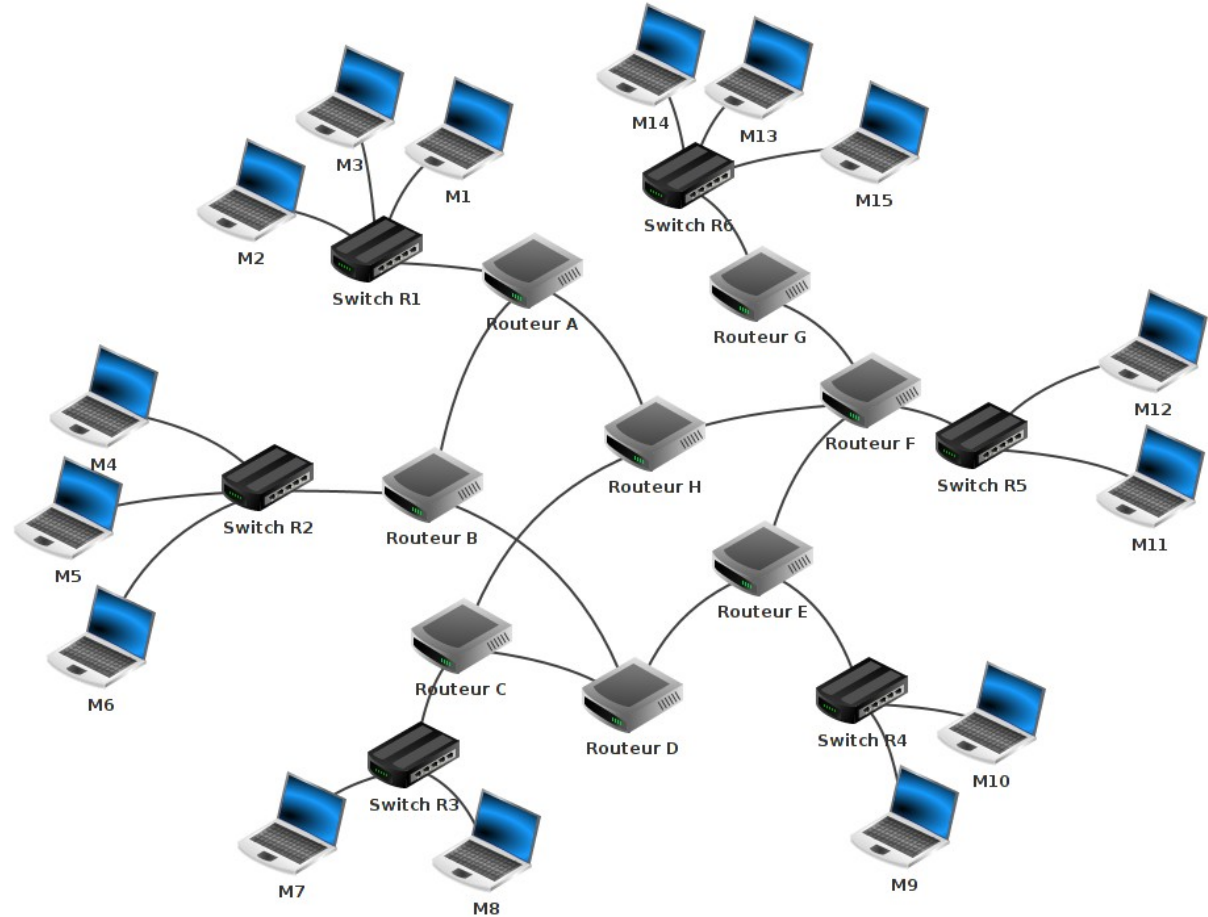
# Table de routage avec RIP

Voici la table de notre réseau, après application de RIP :

Routeur	Réseau	Moyen de l'atteindre	Métrique
A	172.168.0.0	eth0	0
	192.168.7.0	eth1	0
	172.169.0.0	eth2	0
G	10.0.0.0	Routeur G	1
	10.0.0.0	eth0	0
	192.168.7.0	eth1	0
	172.168.0.0	Routeur A	1
	172.169.0.0	Routeur A	1

# Exercice

A partir de ce réseau représenté sous forme de graphe, appliquer RIP pour déterminer le plus court chemin de M1 à M11.



# Problème de RIP

**Qu'est-ce que RIP ne prend pas en compte ?**

# Problème de RIP

**Qu'est-ce que RIP ne prend pas en compte ?**

Que le plus court chemin entre deux machines dépend du débit des connexions empruntées.

# OSPF

Le protocole **Open Short Path First (OSPF)** fonctionne également sur un échange périodique des tables de routage entre routeurs voisins.

La différence est que la métrique utilisée est fonction du débit de la connexion entre les routeurs :

$$\text{coût} = \frac{10^8}{\text{débit}}$$

Un *coût* est associé à chaque connexion.

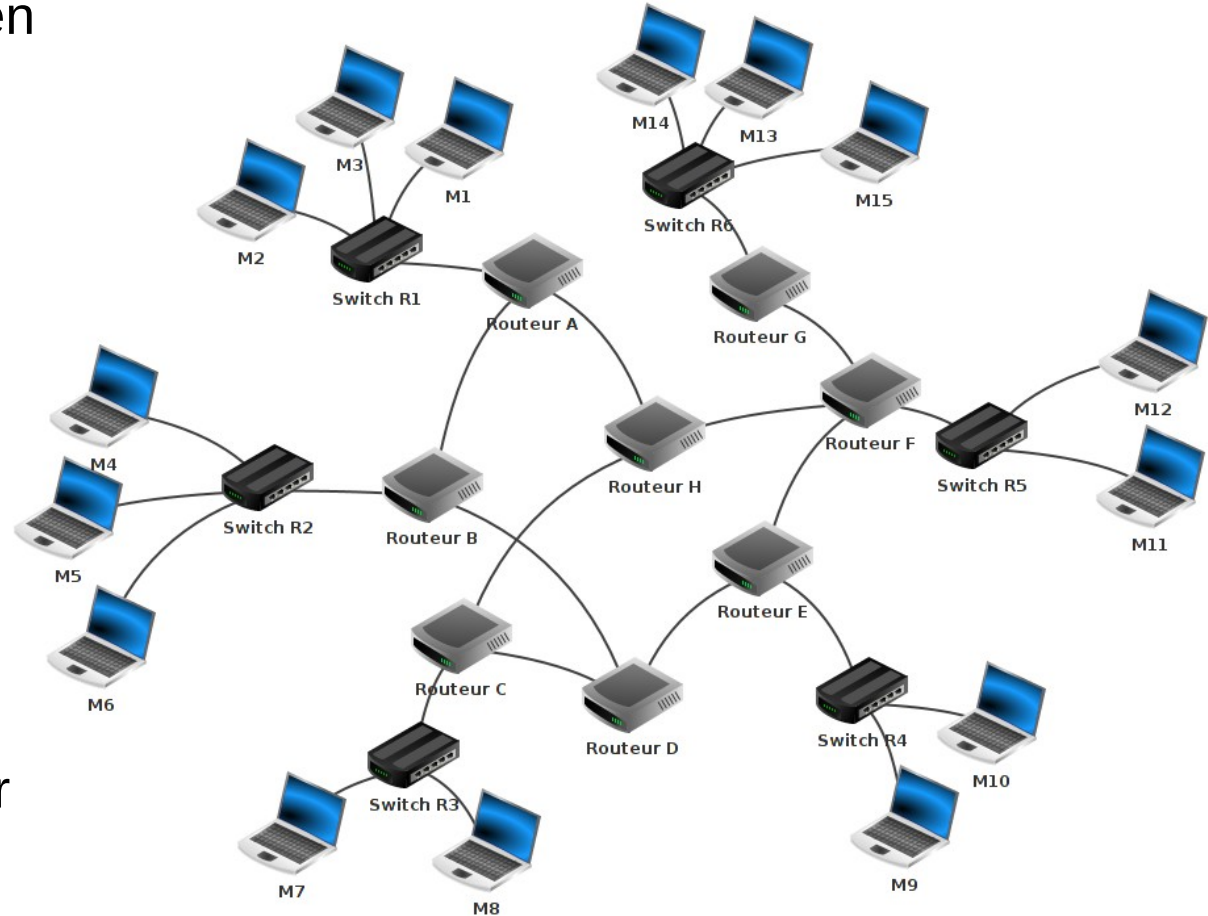
Le débit est en bits par seconde.

# Exercice

Les débits des connexions entre routeurs sont indiqués ci-dessus en Mb/s :

- A – B : 10
- A – H : 1
- B – D : 100
- C – D : 10
- C – H : 10
- D – E : 10
- E – F : 100
- F – G : 1
- H – F : 10

Ajouter les coûts des connexions au graphe et appliquer OSPF pour déterminer le plus court chemin entre M1 et M11.



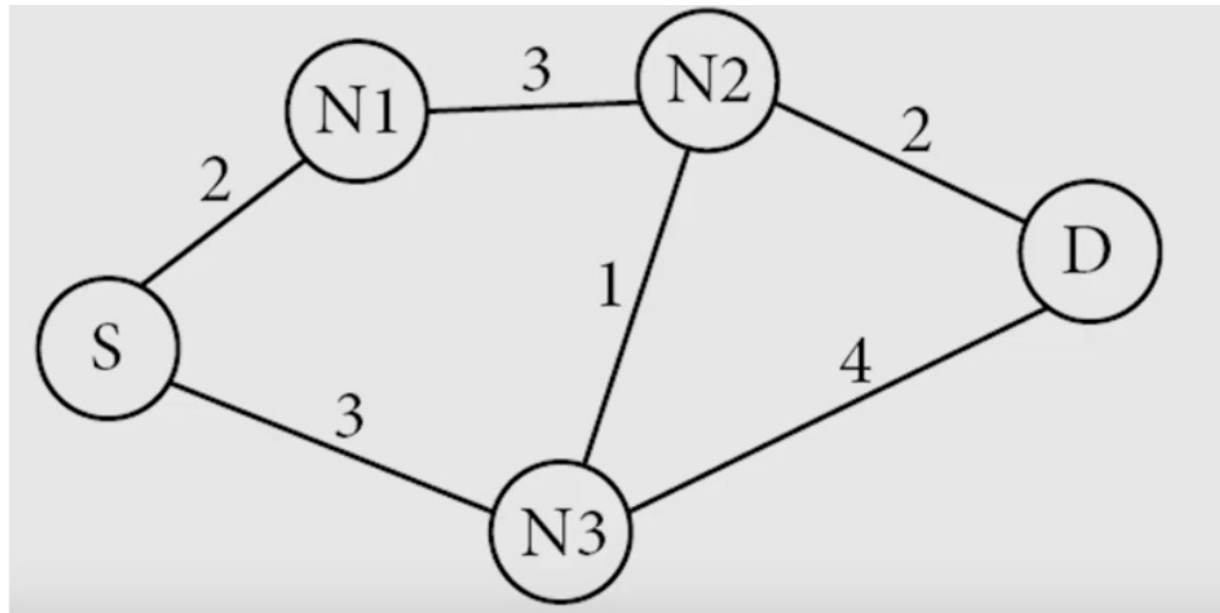
# Algorithme du plus court chemin

En pratique, l'**algorithme de Dijkstra** permet de trouver *le plus court chemin* dans un graphe pondéré en identifiant celui dont la somme des poids est minimale.

Algorithme de Dijkstra



# Algorithme du plus court chemin



# Algorithme du plus court chemin

- Initialisation du tableau avec la valeur «  $\infty$  » pour tous les sommets, sauf la source, mise à 0.
- Mise à jour de la valeur des **sommets voisins** avec la valeur du sommet de départ + le poids de arête permettant de l'atteindre.
- On réitère ces étapes à partir du sommet associé à la **plus petite distance**, non déjà traité. Si une valeur de distance est déjà attribuée à un sommet, on ne met celle-ci à jour que si la nouvelle est **inférieure**.
- L'algorithme s'arrête lorsque l'on arrive au sommet destination.