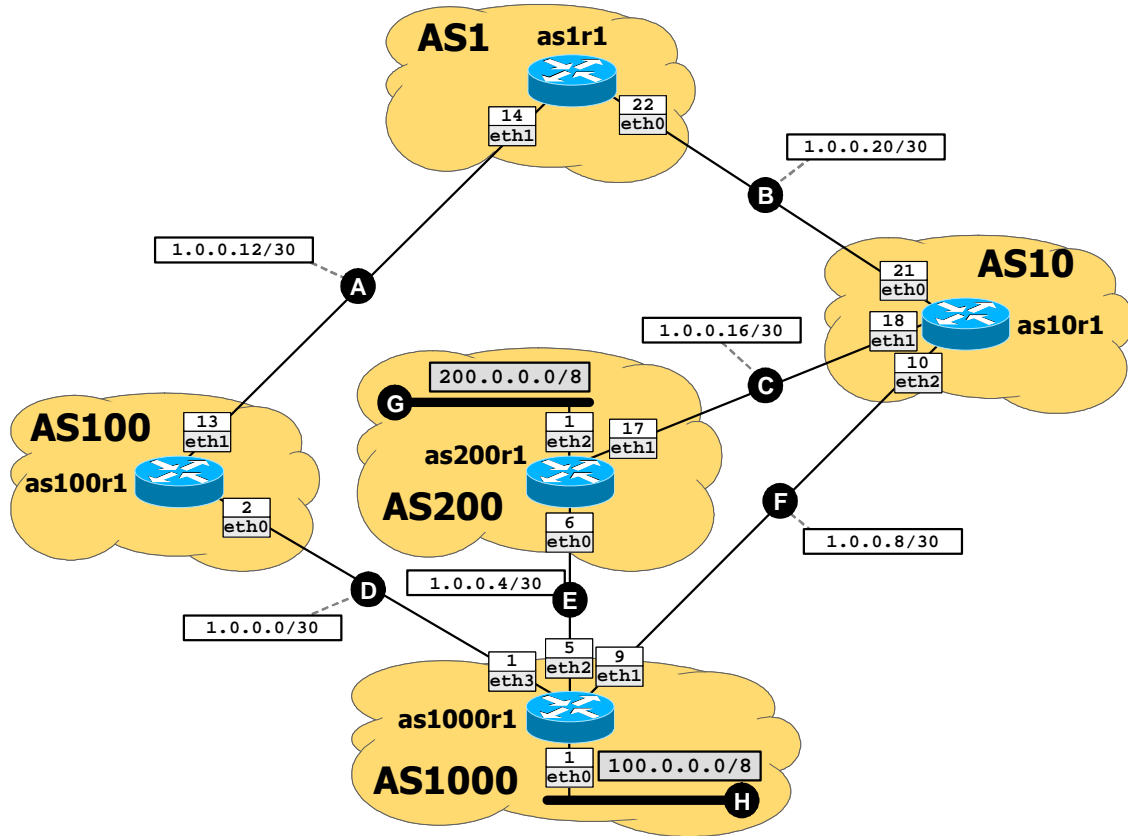**Available time: 120 minutes.**

Using Netkit, implement the network shown in the figure and described below (you can use the following items as a checklist).
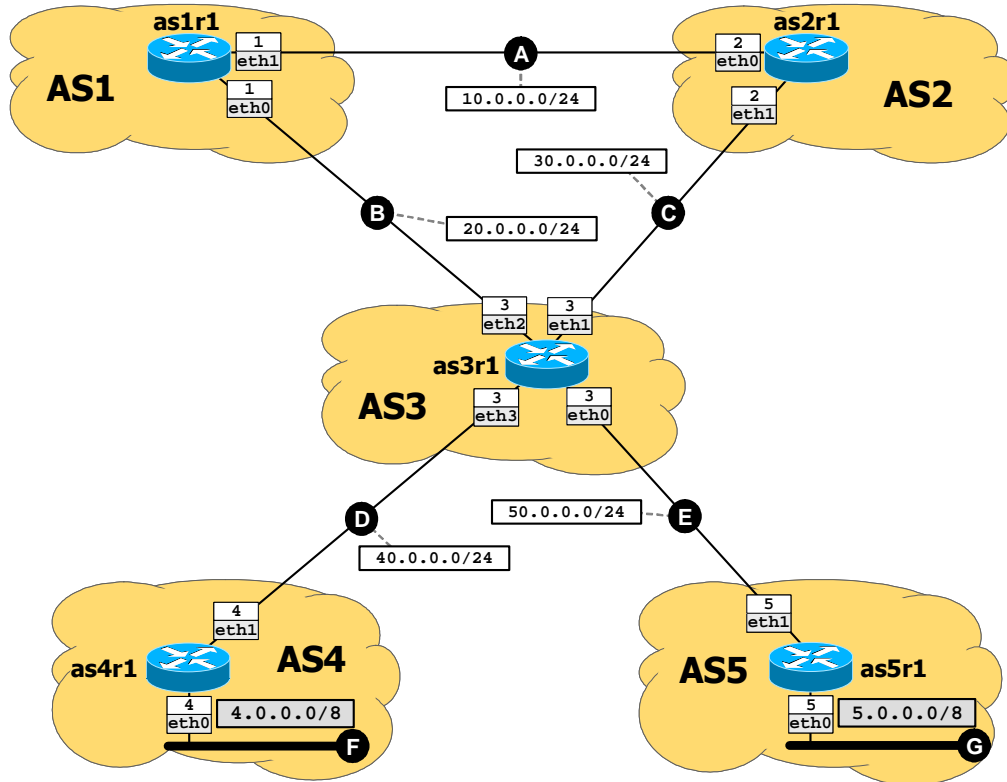
☐ Peering LANs are announced in BGP. `as1000r1` and `as200r1` also announce their internal subnets (in gray).
- The network is structured in a customer-provider hierarchy (higher routers are providers for lower routers). Therefore:
  ☐ All routers except `as1r1` and `as1000r1` use `default-originate` to announce <u>only</u> the default route to their customers.
  ☐ `as1000r1` <u>filters</u> outgoing BGP announcements to prevent being traversed by traffic among its providers.
- Routers tag BGP announcements with communities in the following way:
  ☐ `as1000r1` tags announcements of network `100.0.0.0/8` sent to `as10r1` with community `1000:200`.
    Tip: use command `set community communityValue` in a `route-map`.
  ☐ `as1r1` and `as100r1` <u>prefer</u> announcements that are tagged with community `1000:200`.
    Tip: use `match community clName` in a `route-map` and specify the community to be matched with
    `ip community-list standard clName permit communityValue`.
- Note: communities are only visible if you use `show ip bgp prefix`.

Goals:
- All routers must be able to reach any IP addresses on the network.
- `as100r1` must reach `100.0.0.1` <u>preferably</u> using links A, B, and F.
- `as100r1` must be able to reach `200.0.0.1` <u>only</u> using links A, B, and C.

Available time: <u>120 minutes</u>.

as1r1

AS1

1
eth1

1
eth0

A

10.0.0.0/24

2
eth0

as2r1

AS2

2
eth1

30.0.0.0/24

B

20.0.0.0/24

C

3
eth2

3
eth1

as3r1

AS3

3
eth3

3
eth0

D

50.0.0.0/24

E

40.0.0.0/24

4
eth1

as4r1

AS4

4
eth0

4.0.0.0/8

F

5
eth1

AS5

as5r1

5
eth0

5.0.0.0/8

G

Using Netkit, implement the network shown in the figure and described below (you can use the following items as a checklist).
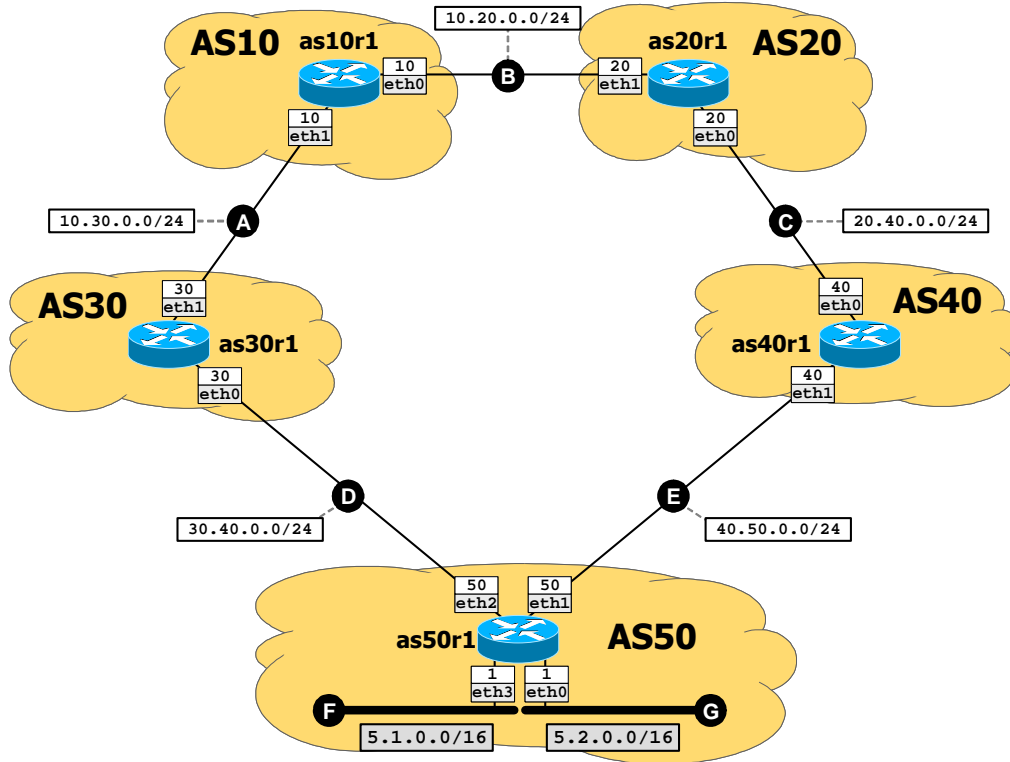
☐ Peering LANs are announced in BGP. `as4r1` and `as5r1` also announce their internal subnets (in gray).
- The network is structured in a customer-provider hierarchy (higher routers are providers for lower routers). Therefore:
    - ☐ `as3r1` announces <u>only</u> the default route to its customers, using `default-originate`.
    - ☐ `as3r1` <u>filters</u> outgoing BGP announcements to prevent being traversed by traffic among its providers.
- `as3r1` tags BGP announcements with communities. In particular:
    - ☐ it tags announcements of network `5.0.0.0/8` sent to `as2r1` with community `3:5`.
      Tip: use command `set community` *communityValue* in a `route-map`.
    - ☐ `as1r1` <u>prefers</u> announcements with community `3:5`.
      Tip: use `match community` *cName* in a `route-map` and specify the community to be matched with
      `ip community-list standard` *cName* `permit` *communityValue*.
- Note: communities are only visible if you use `show ip bgp` *prefix*.

Goals:
- All routers must be able to reach any IP addresses on the network.
- No entries in the BGP routing table of `as1r1` must use AS path `3 2`; no entries on `as2r1` must use `3 1`.
- `as1r1` must reach `5.0.0.5` preferably using links A, C, and E.

Available time: 120 minutes.

Using Netkit, implement the network shown in the figure and described below (you can use the following items as a checklist).
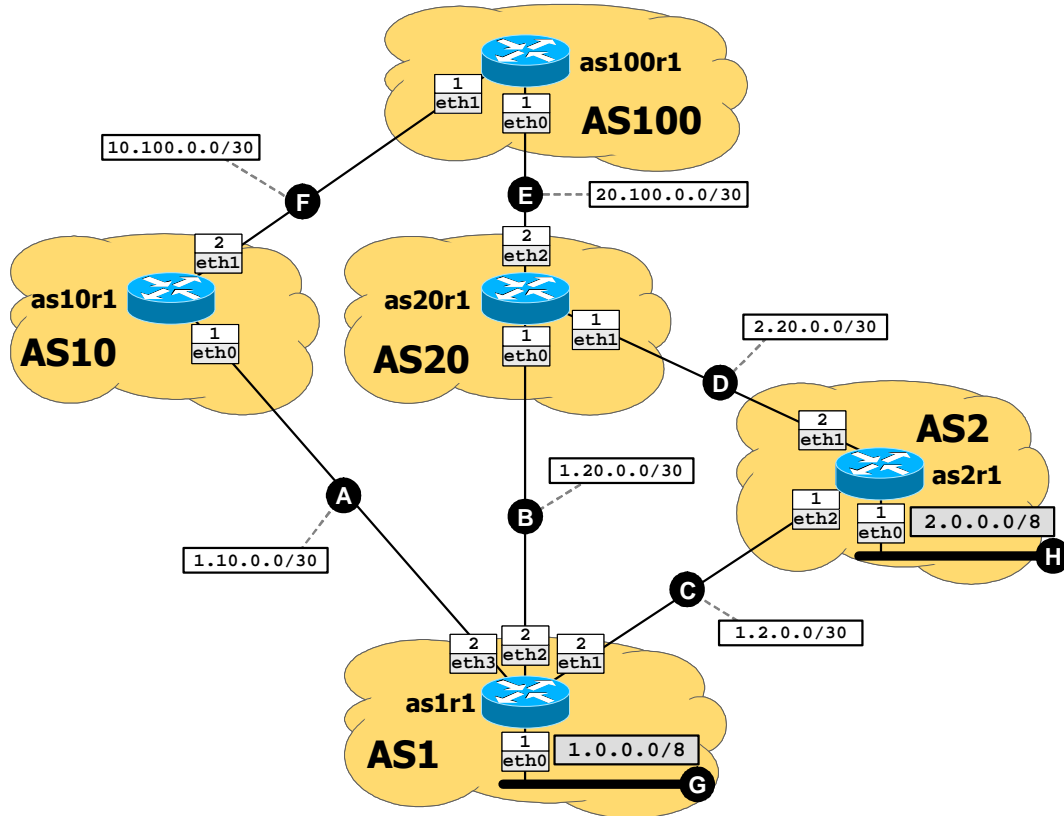
☐ Peering LANs are announced in BGP. `as50r1` also announces its internal subnets (in gray).
- The network is structured in a customer-provider hierarchy (higher routers are providers for lower routers). Therefore:
  - ☐ `as10r1` and `as20r1` <u>do not</u> announce the default route.
  - ☐ `as30r1` and `as40r1` use `default-originate` to announce <u>only</u> the default route to their customers.
  - ☐ `as50r1` <u>filters</u> outgoing BGP announcements to prevent being traversed by traffic among its providers.
- `as50r1` tags BGP announcements with communities. In particular:
  - ☐ it tags announcements of network `5.1.0.0/16` sent to `as30r1` with community `50:1` and announcements of network `5.2.0.0/16` sent to `as40r1` with community `50:2`.
    Tip: use command `set community` *communityValue* in a `route-map`.
  - ☐ `as10r1` and `as30r1` <u>prefer</u> announcements that are tagged with community `50:2`, whereas `as20r1` and `as40r1` <u>prefer</u> announcements tagged with community `50:1`.
    Tip: use `match community` *clName* in a `route-map` and specify the community to be matched with
    `ip community-list standard` *clName* `permit` *communityValue*.
- Note: communities are only visible if you use `show ip bgp` *prefix*.

Goals:
- All routers must be able to reach any IP addresses on the network.
- No entries in the BGP routing table of `as30r1` must use AS path `50 40`; no entries on `as40r1` must use `50 30`.
- `as30r1` must reach `5.2.0.1` preferably using links A, B, C, and E.
- `as40r1` must reach `5.1.0.1` preferably using links C, B, A, and D.

Available time: 120 minutes.

as100r1
1
eth1
1
eth0
AS100

10.100.0.0/30

F

E
20.100.0.0/30

2
eth1
2
eth2

as10r1
AS10
1
eth0

as20r1
1
eth0
AS20
1
eth1
2.20.0.0/30

D

2
eth1
AS2
as2r1
1
eth2
1
eth0
2.0.0.0/8
H

1.20.0.0/30

A

B

C
1.2.0.0/30

1.10.0.0/30

2
eth3
2
eth2
2
eth1

as1r1
AS1
1
eth0
1.0.0.0/8
G

Using Netkit, implement the network shown in the figure and described below (you can use the following items as a checklist).

☐ Peering LANs are announced in BGP. `as1r1` and `as2r1` also announce their internal subnets (in gray).
- The network is structured in a customer-provider hierarchy (higher routers are providers for lower routers). Therefore:
  ☐ All routers except `as100r1` and `as1r1` use `default-originate` to announce <u>only</u> the default route to their customers.
  ☐ `as1r1` <u>filters</u> outgoing BGP announcements to prevent being traversed by traffic among its providers.
- `as1r1` tags BGP announcements with communities. In particular:
  ☐ it tags announcements of network `1.0.0.0/8` sent to `as2r1` with community `1:200`.
    Tip: use command `set community` *communityValue* in a `route-map`.
  ☐ `as20r1` <u>prefers</u> announcements with community `1:200`.
    Tip: use `match community` *clName* in a `route-map` and specify the community to be matched with
    `ip community-list standard` *clName* `permit` *communityValue*.
- Note: communities are only visible if you use `show ip bgp` *prefix*.

Goals:
- All routers must be able to reach any IP addresses on the network.
- `as10r1` must be able to reach `2.0.0.1` <u>only</u> using links F, E, and D.
- `as20r1` must reach `1.0.0.1` <u>preferably</u> using links D and C.