DTN Scalability Using Regions

Scott Burleigh

12 November 2021

Why?

- Here’s the motivation for this work:
    - Hypothesis: adoption of DTN is slowed by the absence of commercial products.
    - Commercial products won’t emerge until there is – at least potentially – a large market for DTN products.
    - That can happen only if networks built on DTN can grow to arbitrary size.

What’s the problem?

- Nothing intrinsic to Bundle Protocol or BPSec limits the size of a DTN-based network.
- But there is a second-order problem. DTN can readily be configured to support an extremely wide range of communication scenarios, but this configurability makes deployment and management of DTN nodes somewhat labor-intensive.

How do we solve this problem?

- By automating the routine tasks of network management:
    1. Automate the management of forwarding decisions.
    2. Automate the deployment of new nodes.
- For #1, the approach proposed here is the decomposition of the network into “regions” of limited forwarding scope, supported by Inter-Regional Forwarding (IRF).
- For #2, the approach proposed here is DTN Node Auto-Configuration (DNAC). DNAC automatically configures new DTN nodes, much as DHCP automates the configuration of devices on IP networks.

Inter-Regional Forwarding (1 of 2)

- We assume the emergence of delay-tolerant routing mechanisms that can enable the automation of forwarding among limited numbers of DTN nodes.
  - Contact graph routing (CGR), currently in operational use on the International Space Station, is an existence proof.
- We then define a region as the set of nodes encompassed by a single routing regime based on one of these mechanisms.
  - For example, the set of all nodes cited in any of the contacts in a given contact plan would be a CGR-based region.
- Regions are a little like autonomous systems. They could additionally function as administrative domains, security domains, etc.

Inter-Regional Forwarding (2 of 2)

- This leaves the problem of forwarding from a node that is a member of one region to a node that is a member of another region.
- The Inter-Regional Forwarding strategy is to use the first bundle destined for a given remote node as a probe. Positive and negative feedback from candidate forwarding points propagates back to the source, enabling subsequent bundles destined for the same node to be forwarded more efficiently.

Underpinnings

- DNAC and IRF as currently implemented work only for nodes identified by EIDs formed in the “ipn” URI scheme – that is, nodes identified by number rather than by string. The same principles could be extended to other EID schemes, but that is beyond the scope of the current work.
- Additionally, DNAC as currently implemented supports only the ION implementation of DTN.
- DNAC and IRF are built on some underlying capabilities prototyped in ION:
  - Automatic contact plan synchronization, which in turn is built on…
  - BP multicast, which is based on contact graph routing.
  - Also delay-tolerant key administration, which is built on…
  - Trusted Collective structures, which again rely on BP multicast.

An illustration

- The following slides are an example of the exercise of DNAC and IRF, intended to provide some idea of their capabilities.
- The example is not hypothetical. ION currently performs all of the operations shown. This was actually the test case used to debug the software.

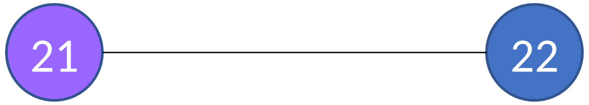
Start by creating node 21 in “root” region 1.

1

Node 21 is instantiated in the usual, managed way (e.g., ionstart).

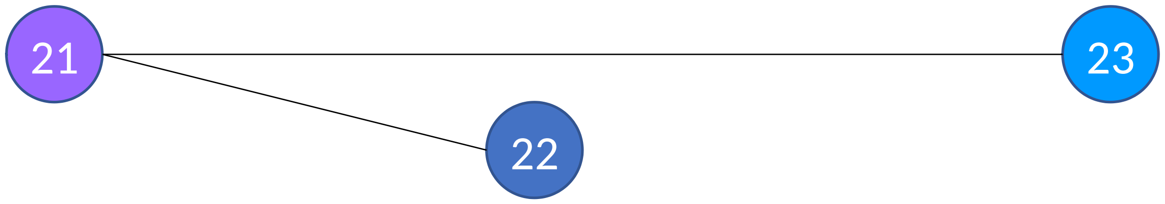
Use DNAC to create node 22…

1



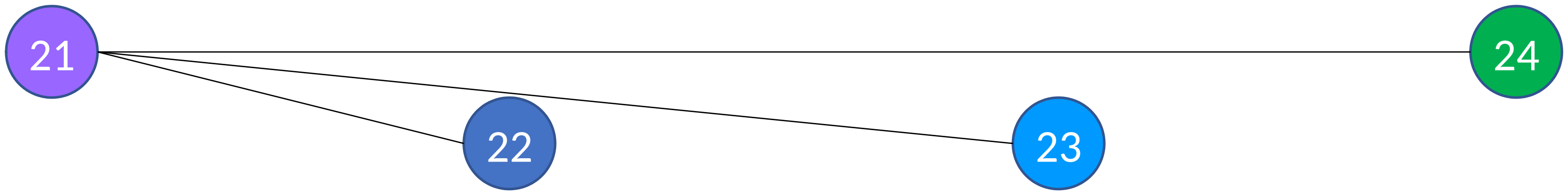
…and node 23…

1

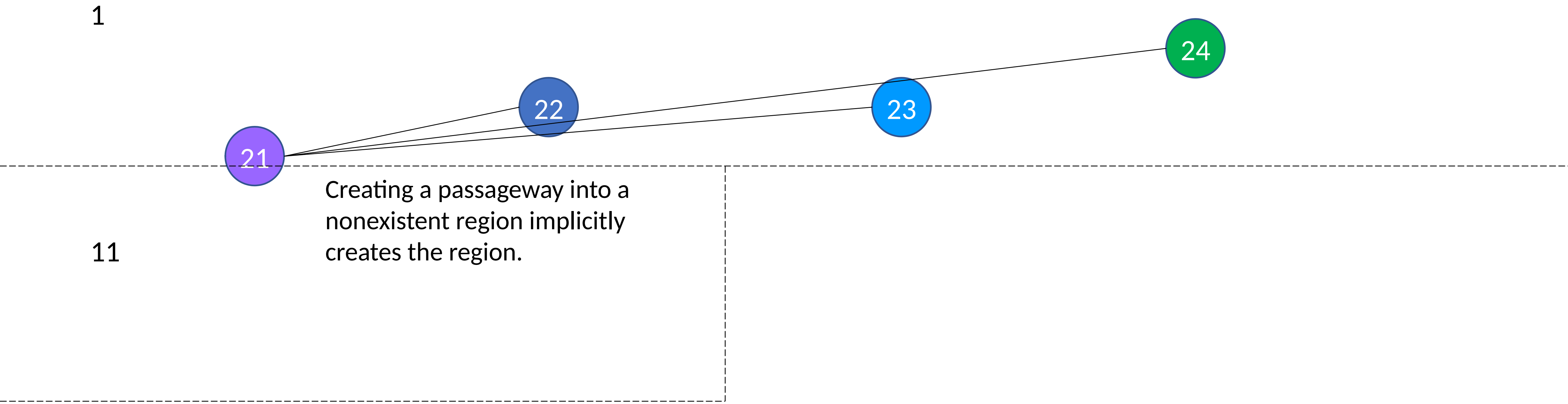


…and node 24.

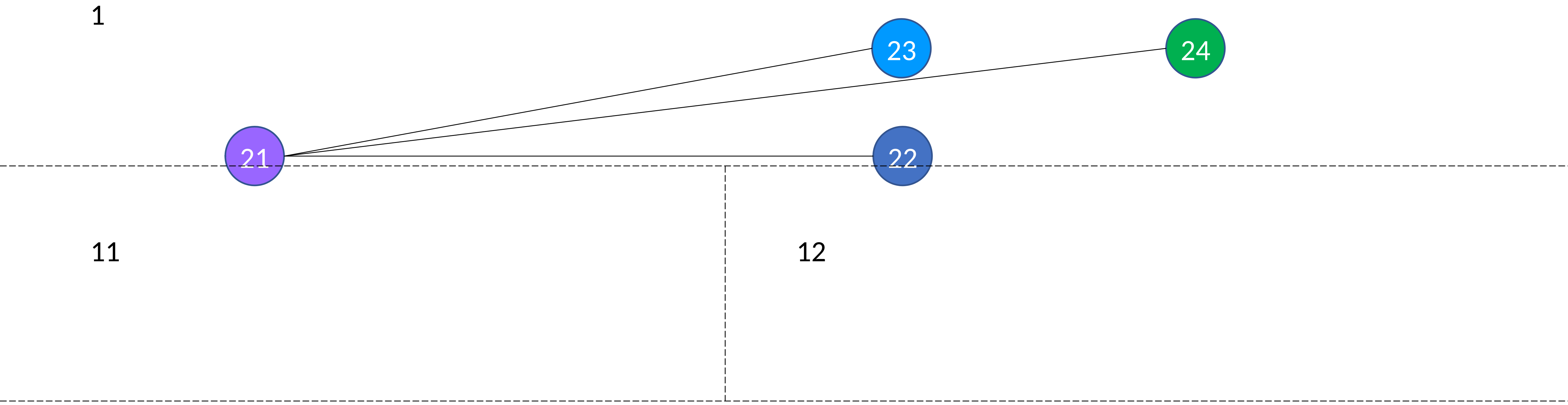
1



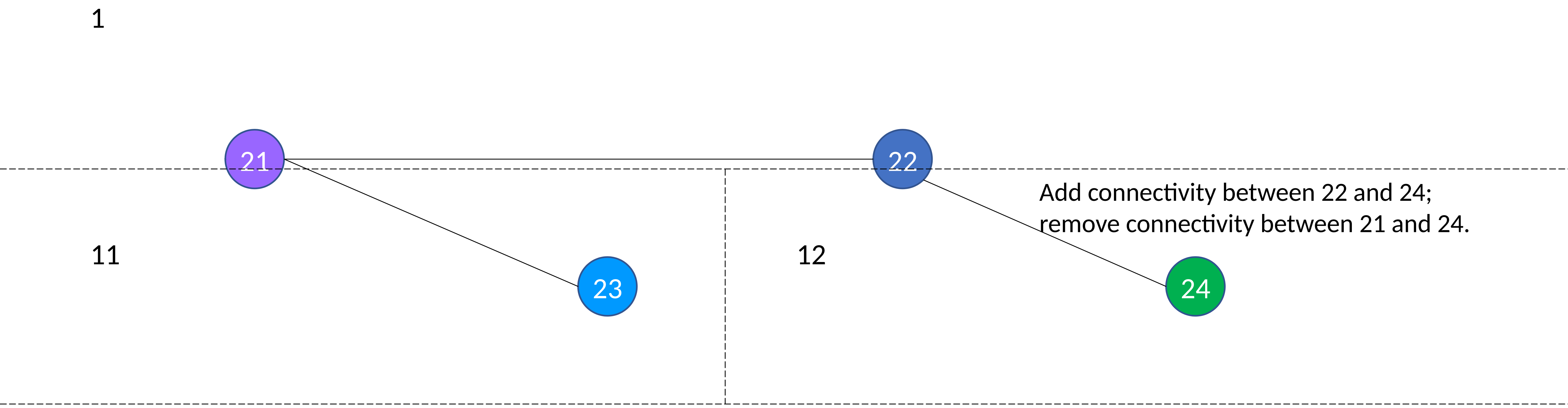
Use irfadmin to make 21 a passageway to 11



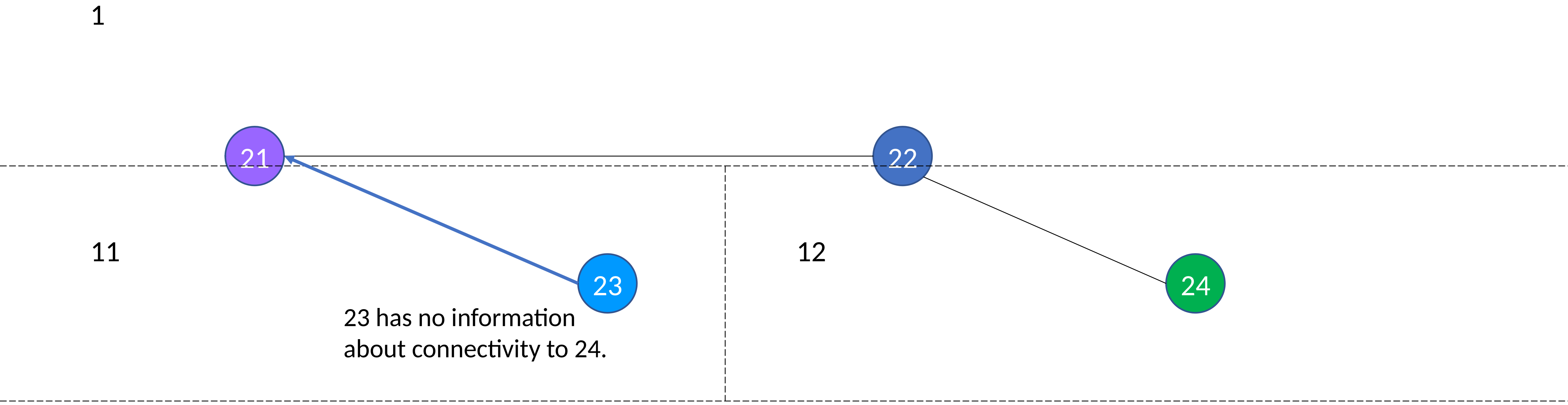
…and to make 22 a passageway to 12.



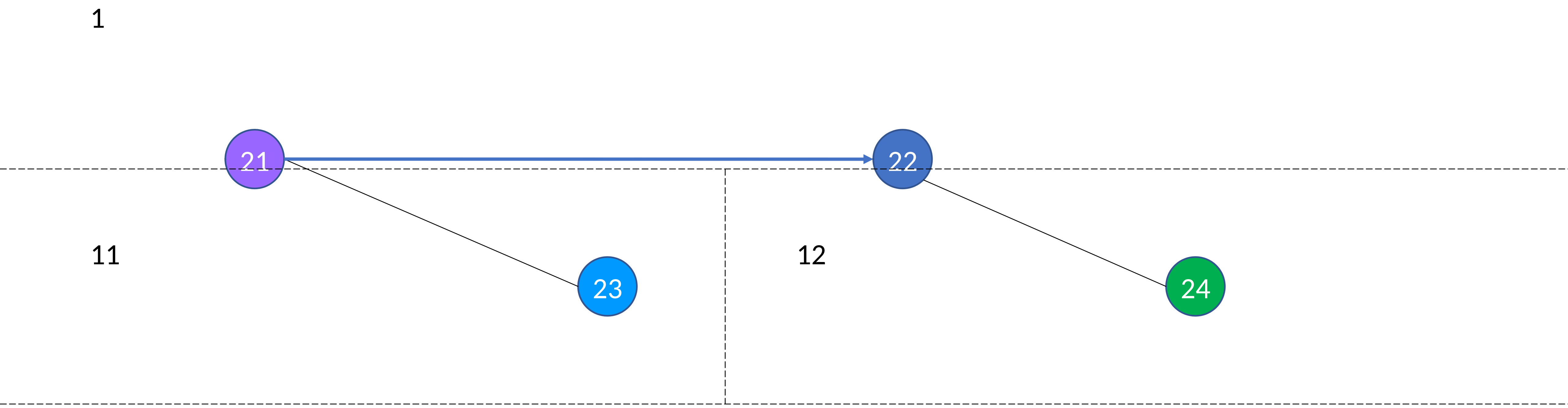
Move node 23 into 11 and node 24 into 12.



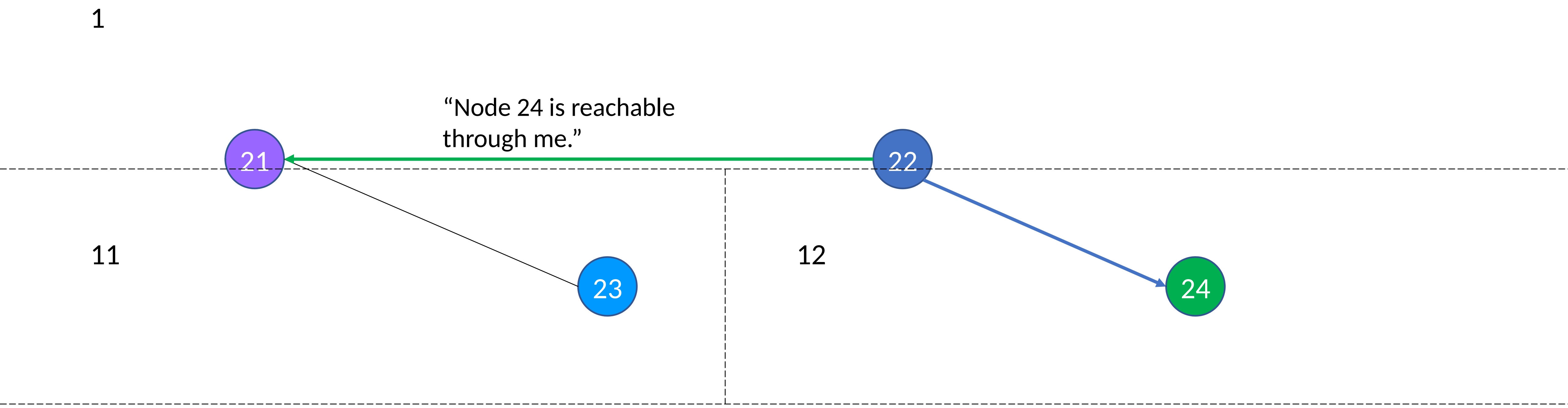
Now send file by BP from 23 to 24.



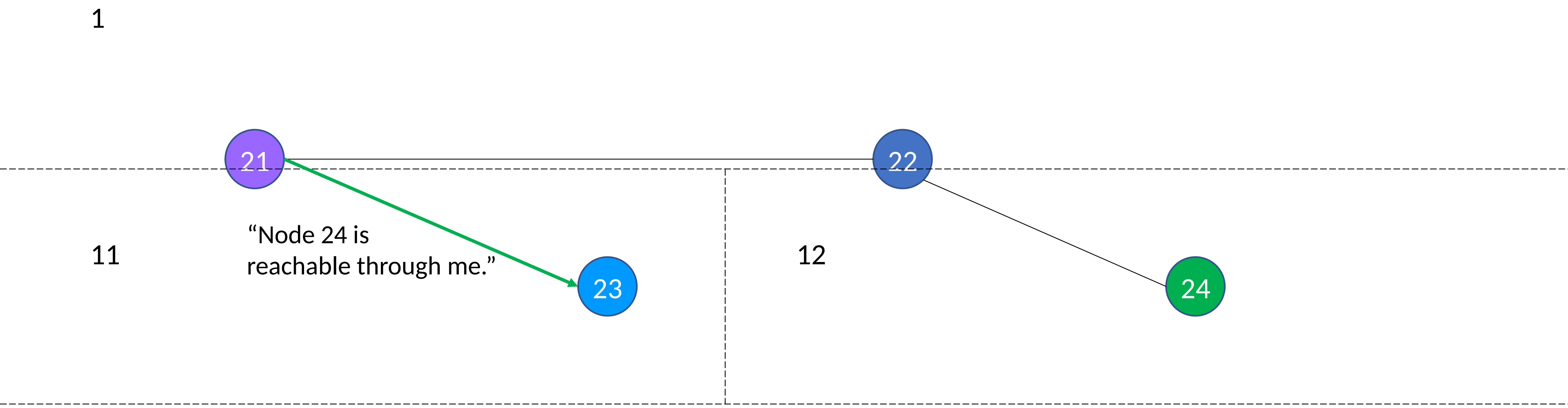
(Forward between passageways)



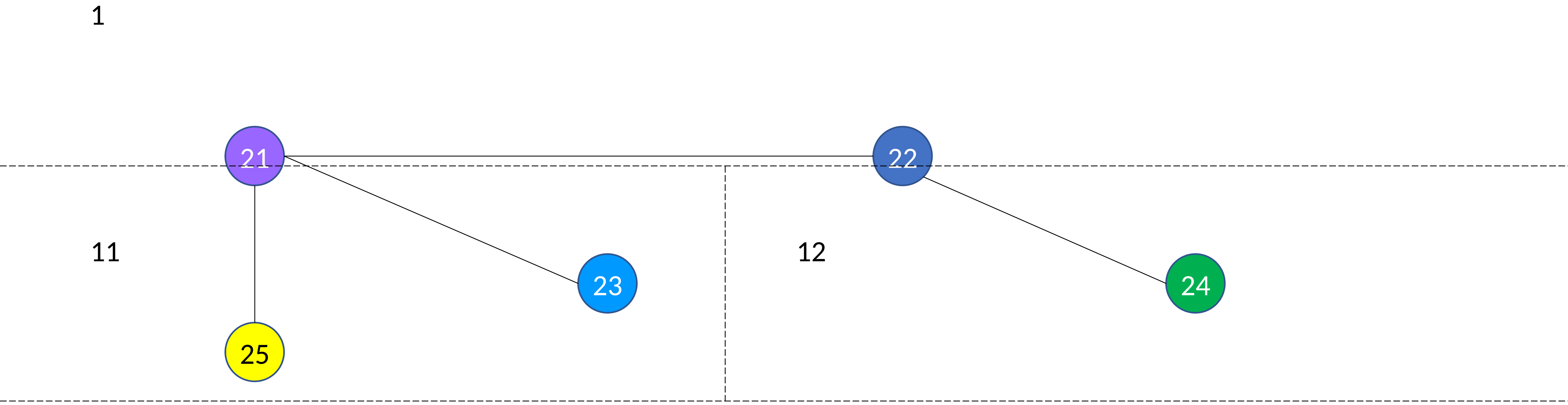
Encouraging feedback (1 of 2)



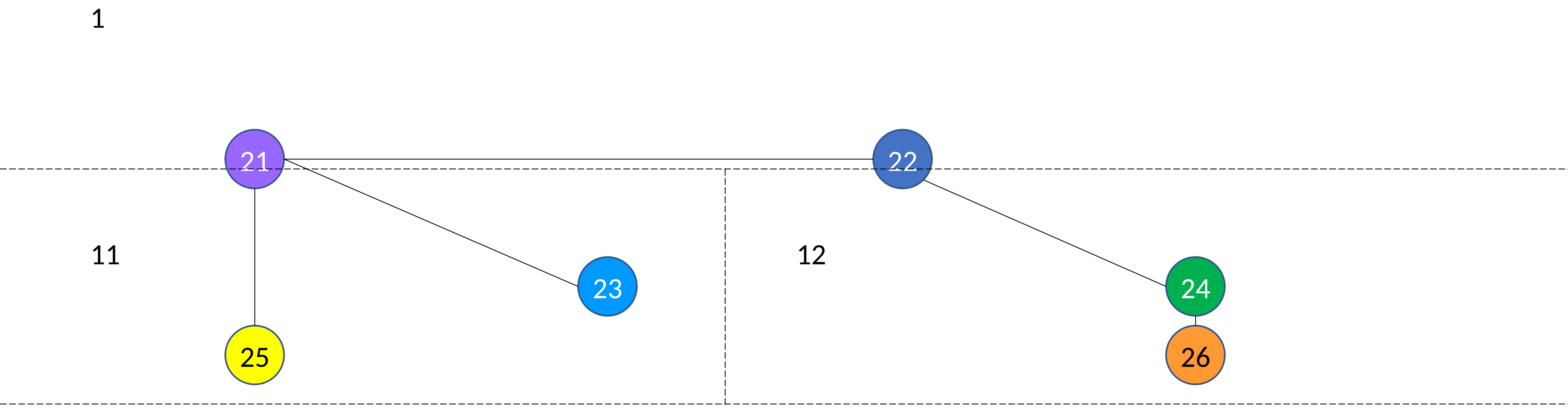
Encouraging feedback (2 of 2)



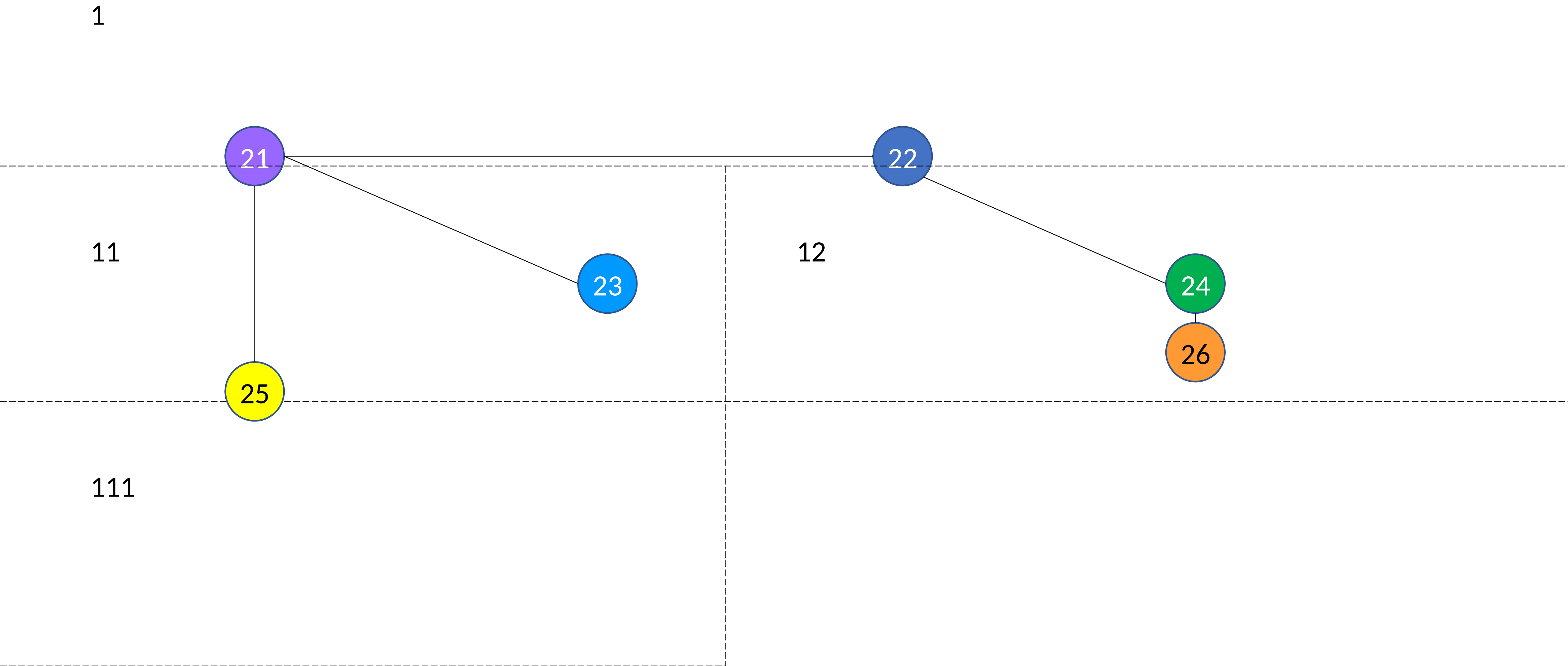
Now use DNAC to create node 25 in region 11



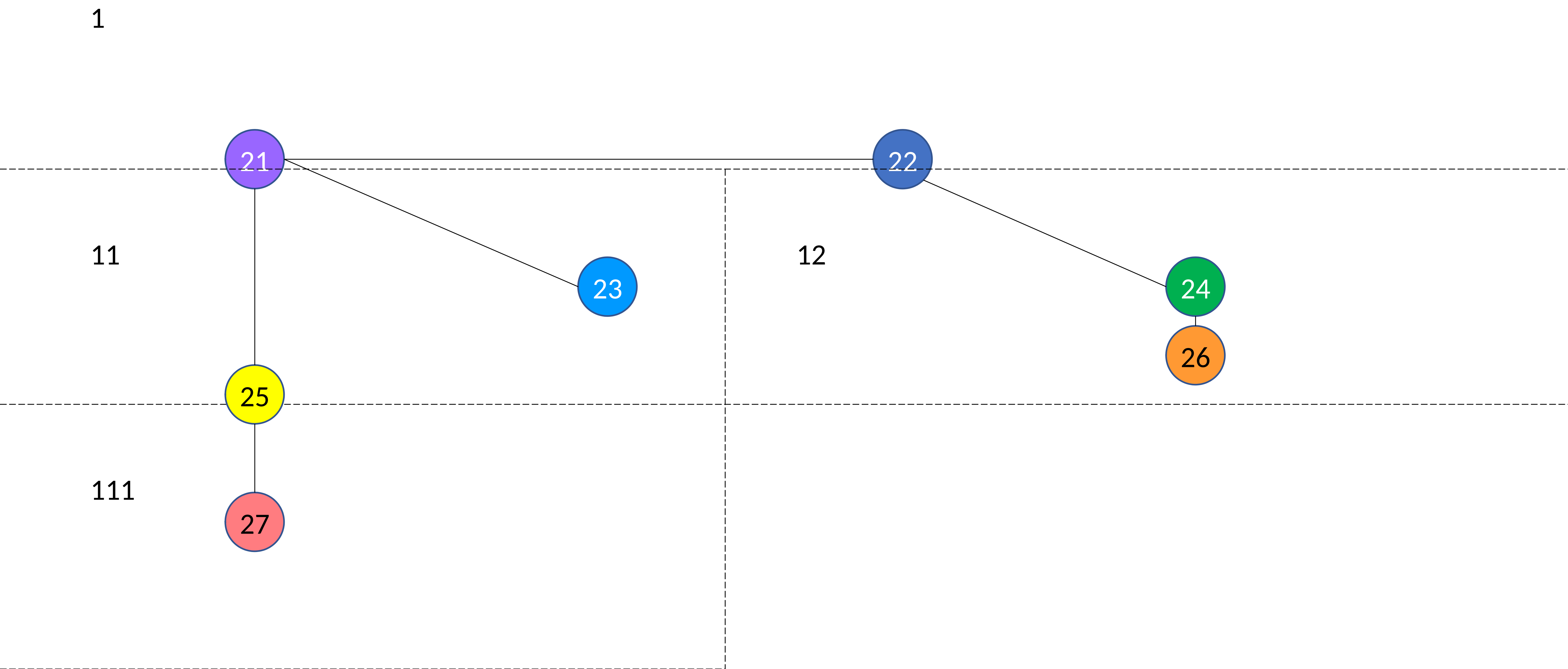
…and to create node 26 in region 12.



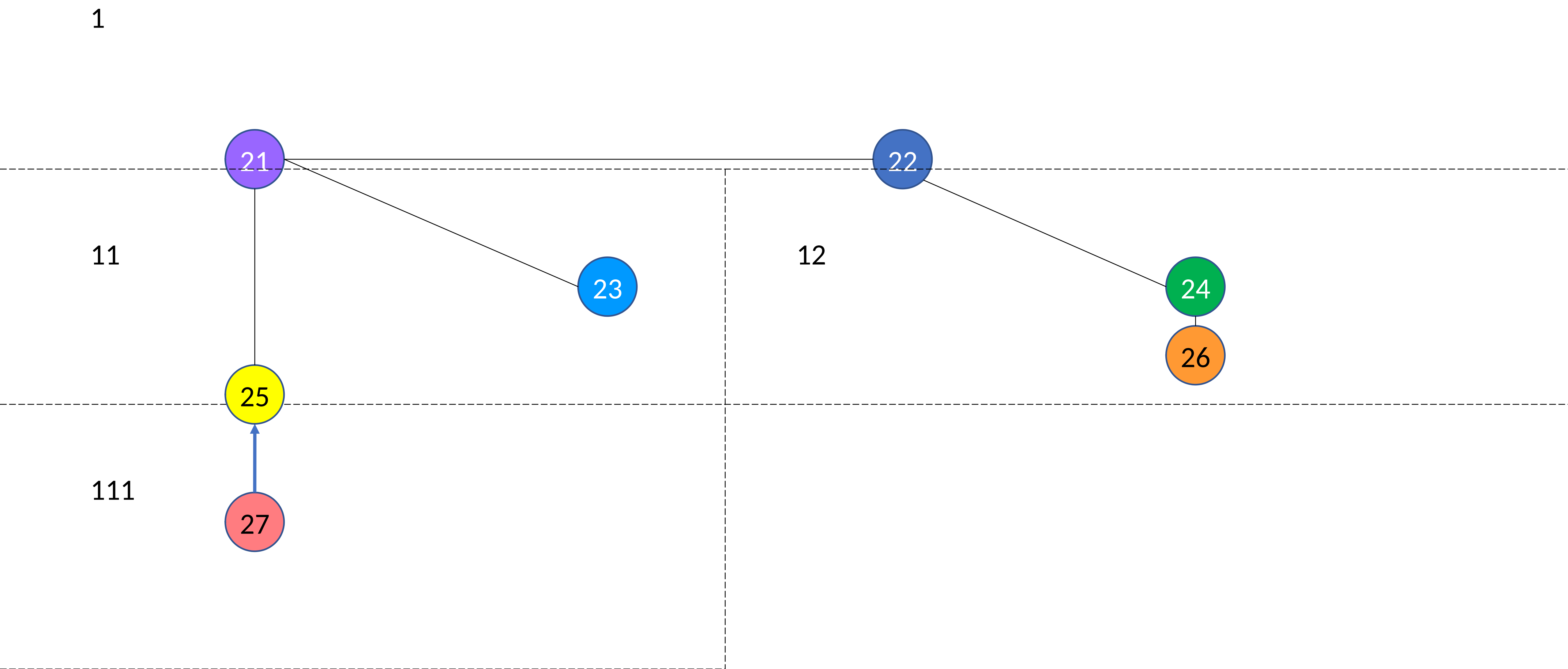
Then make node 25 a passageway into 111…

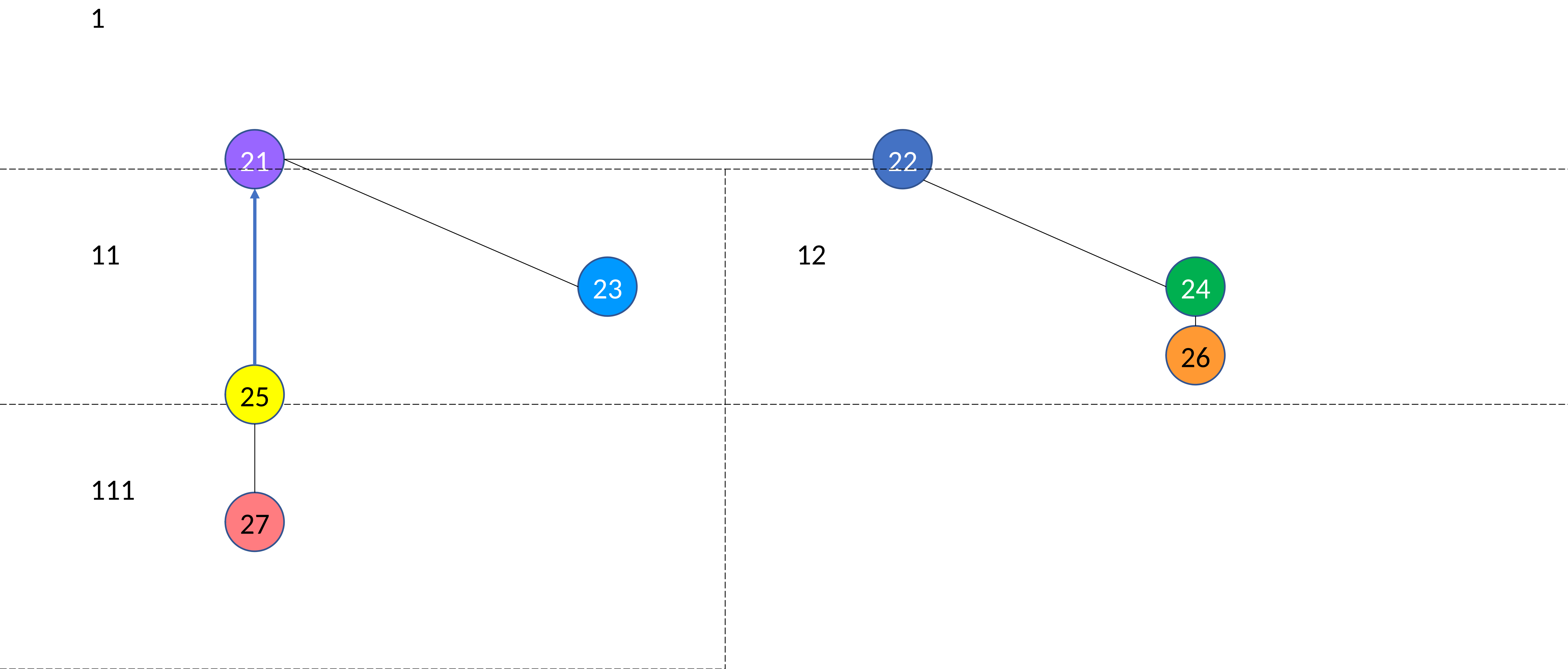


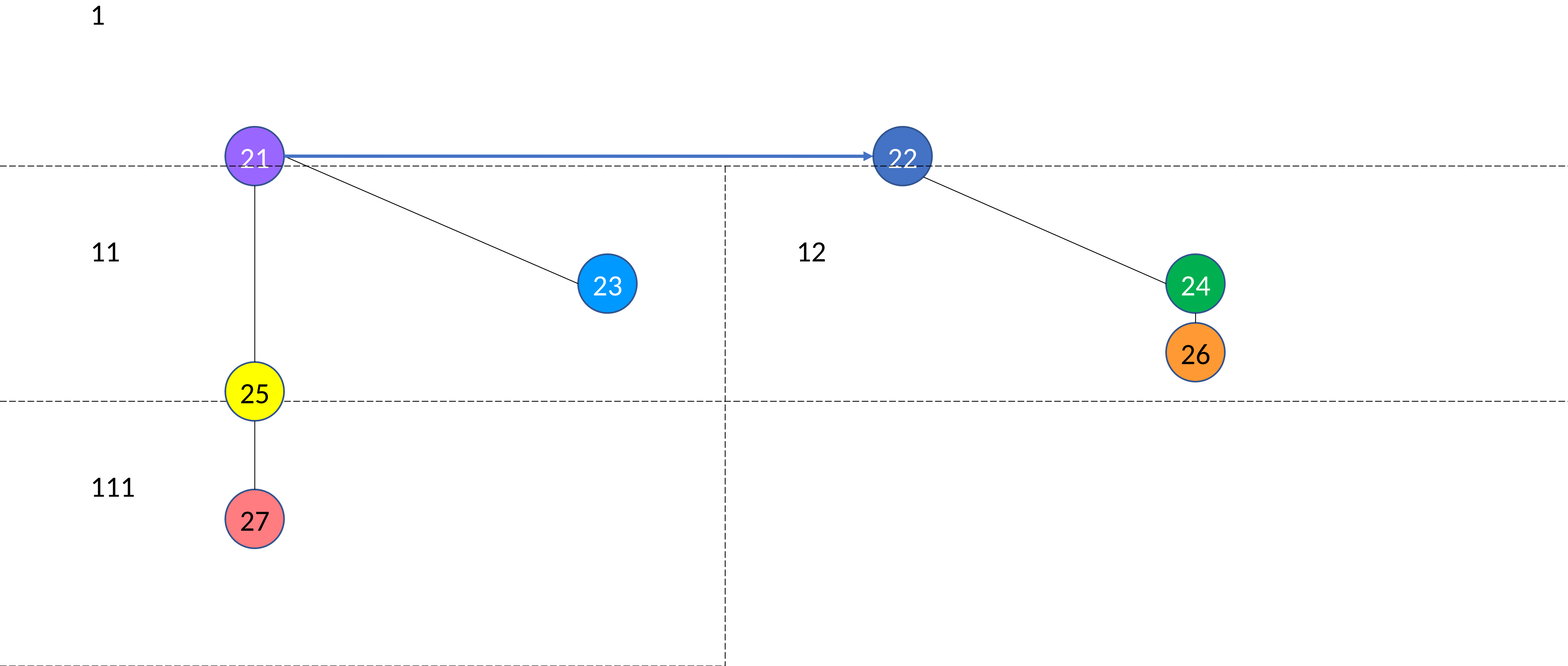
And create node 27 in new region 111.

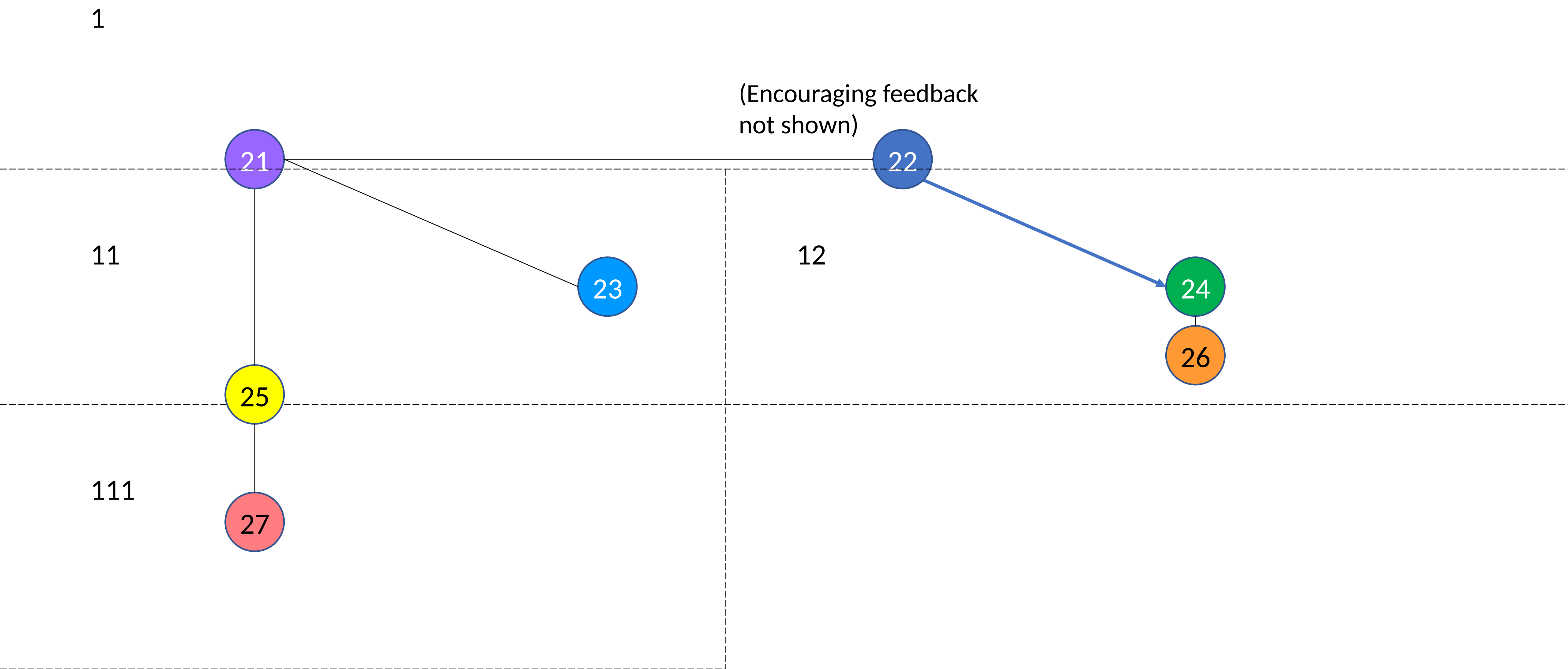


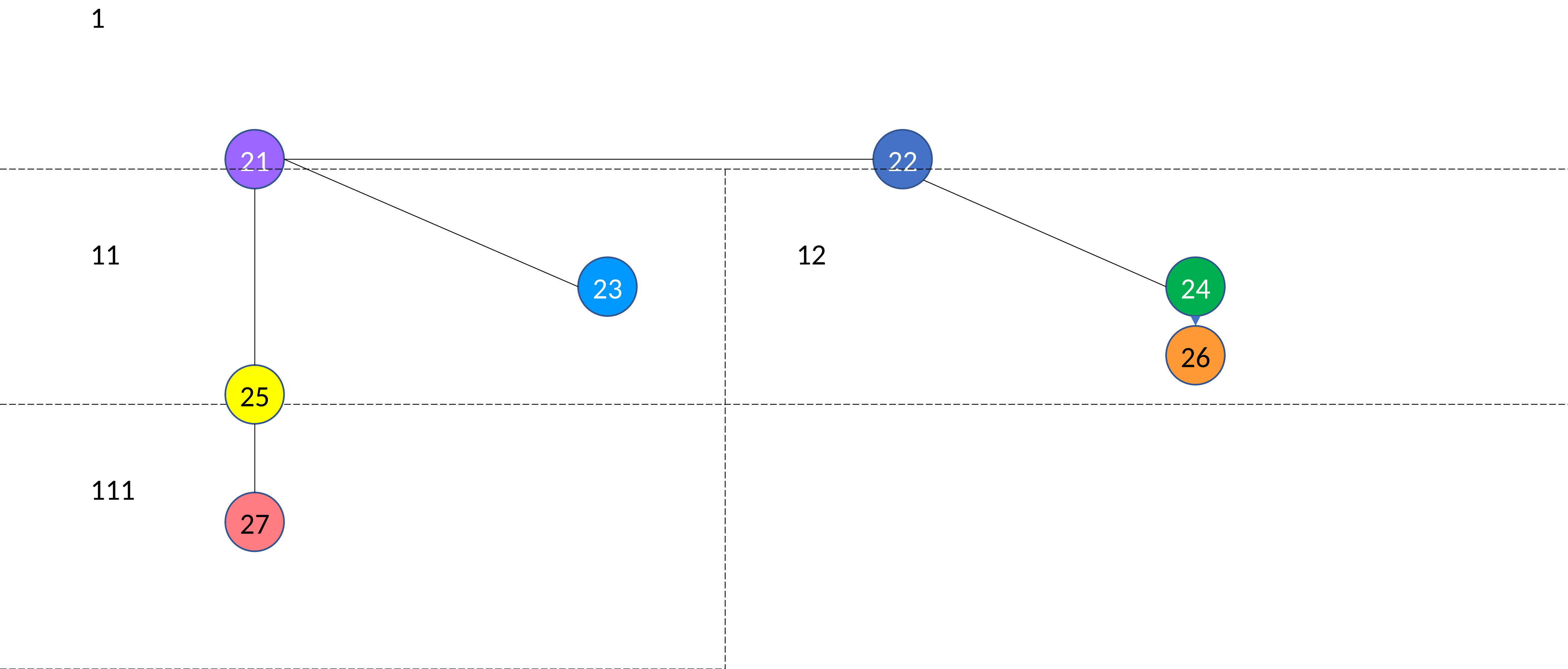
Now send file by BP from 27 to 26.



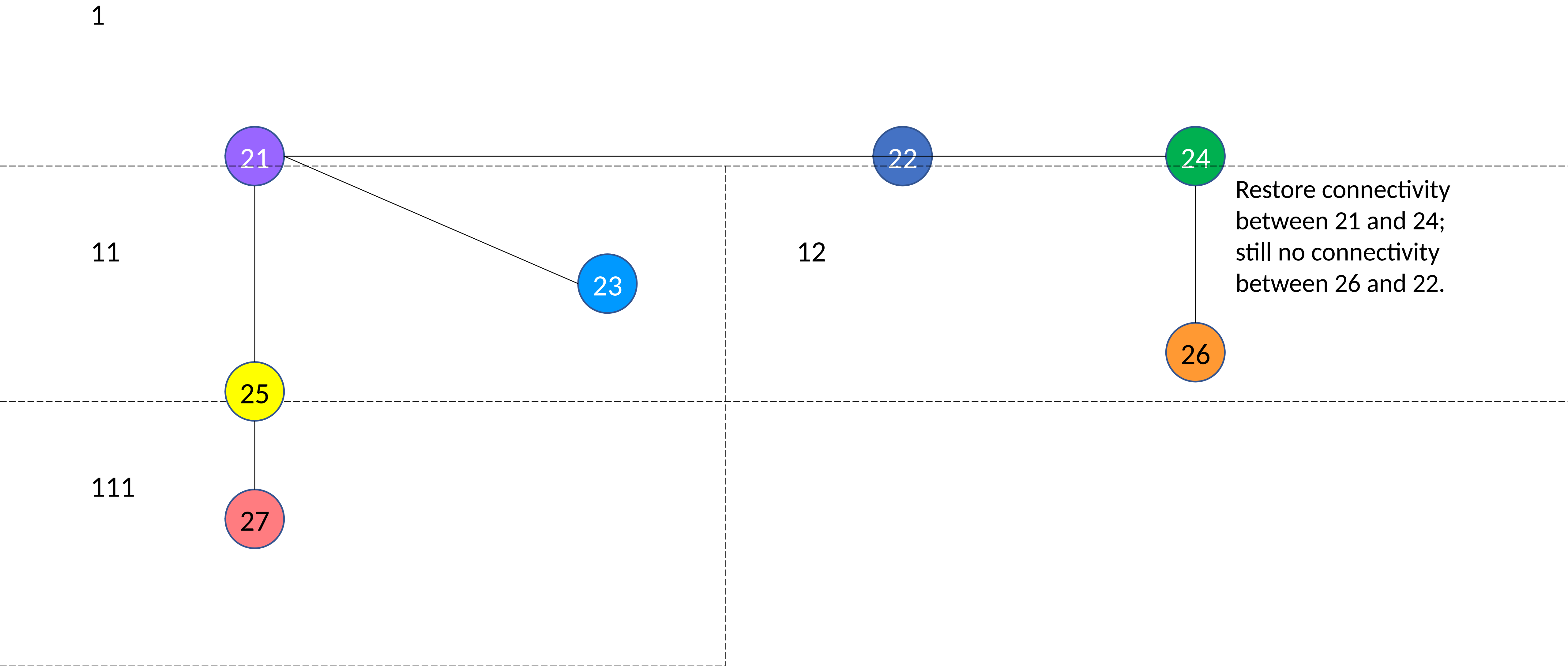

1
21
22
11
23
12
24
26
25
111
27


1
11
12
111
21
22
23
24
25
26
27

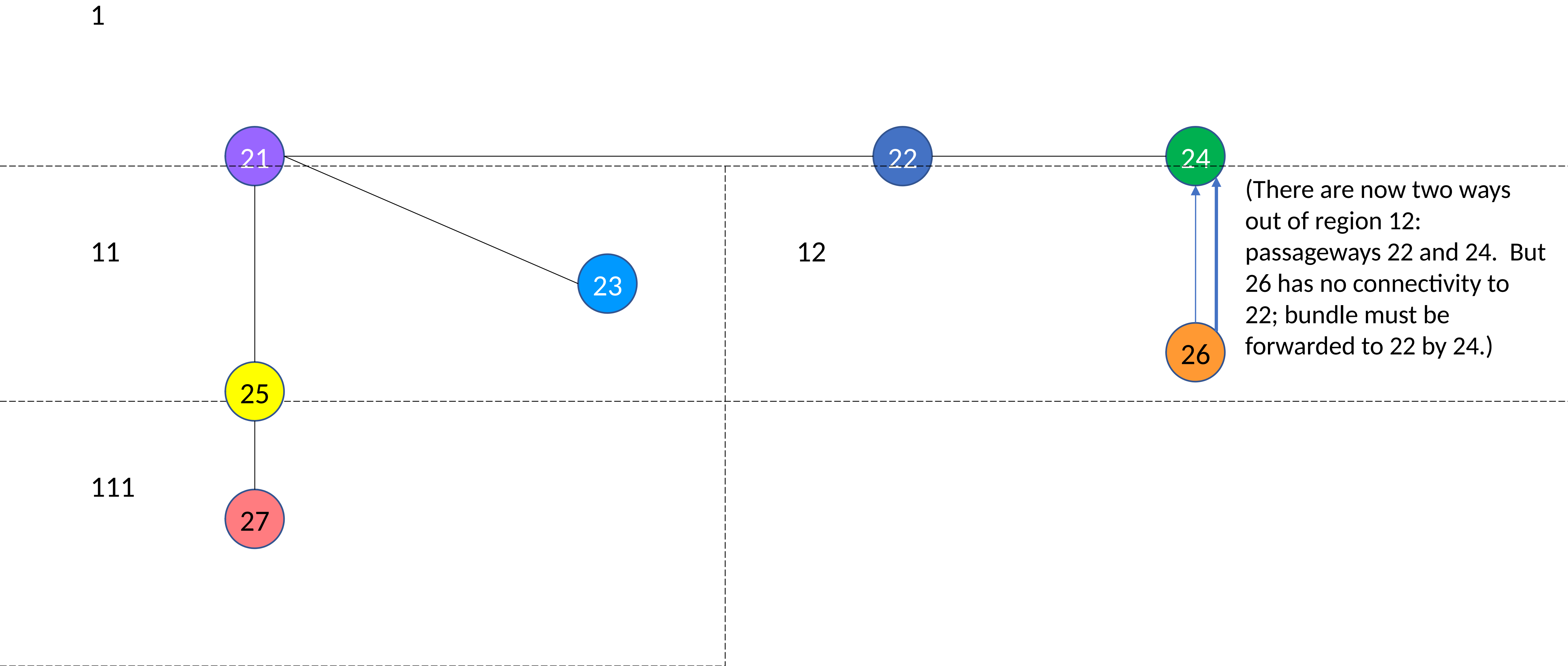

1
(Encouraging feedback
not shown)
21
22
11
12
23
24
26
25
111
27

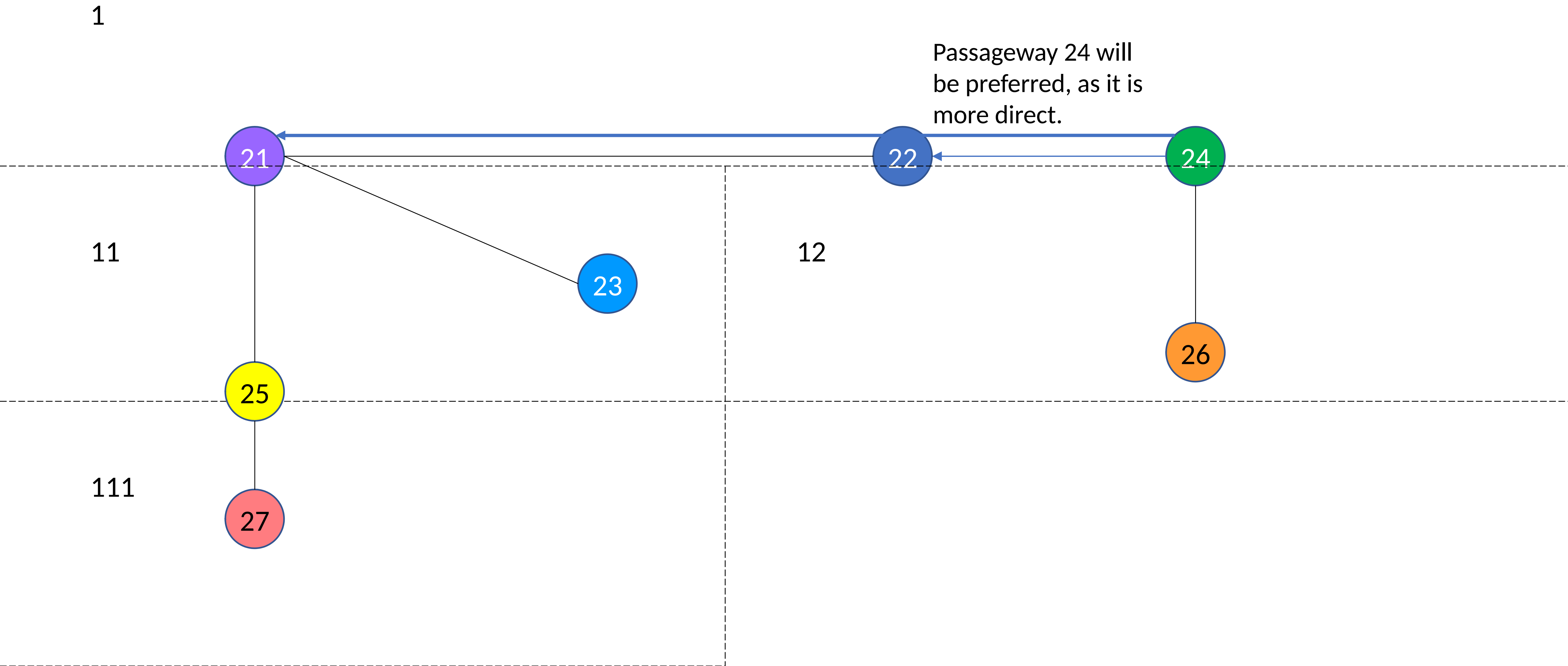

1
11
12
111
21
22
23
24
25
26
27

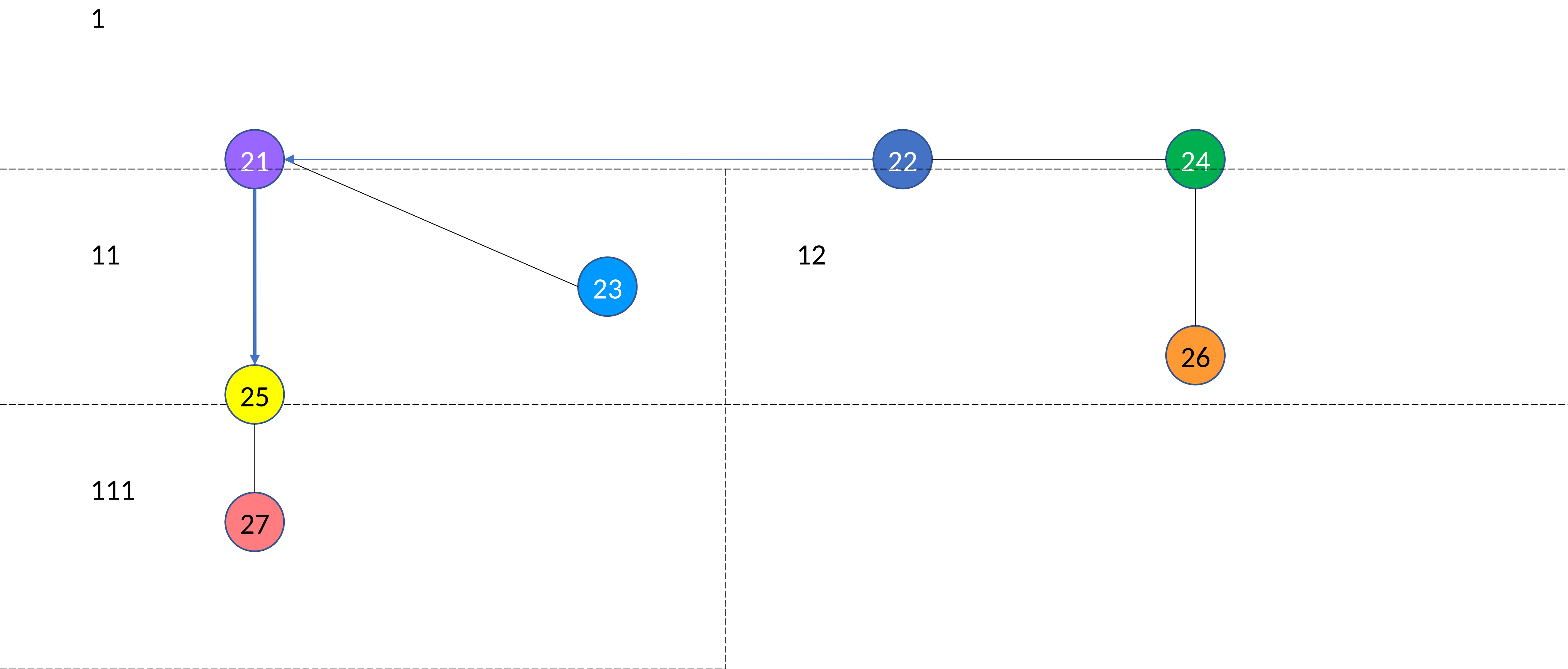
Make node 24 a passageway between 1 & 12.

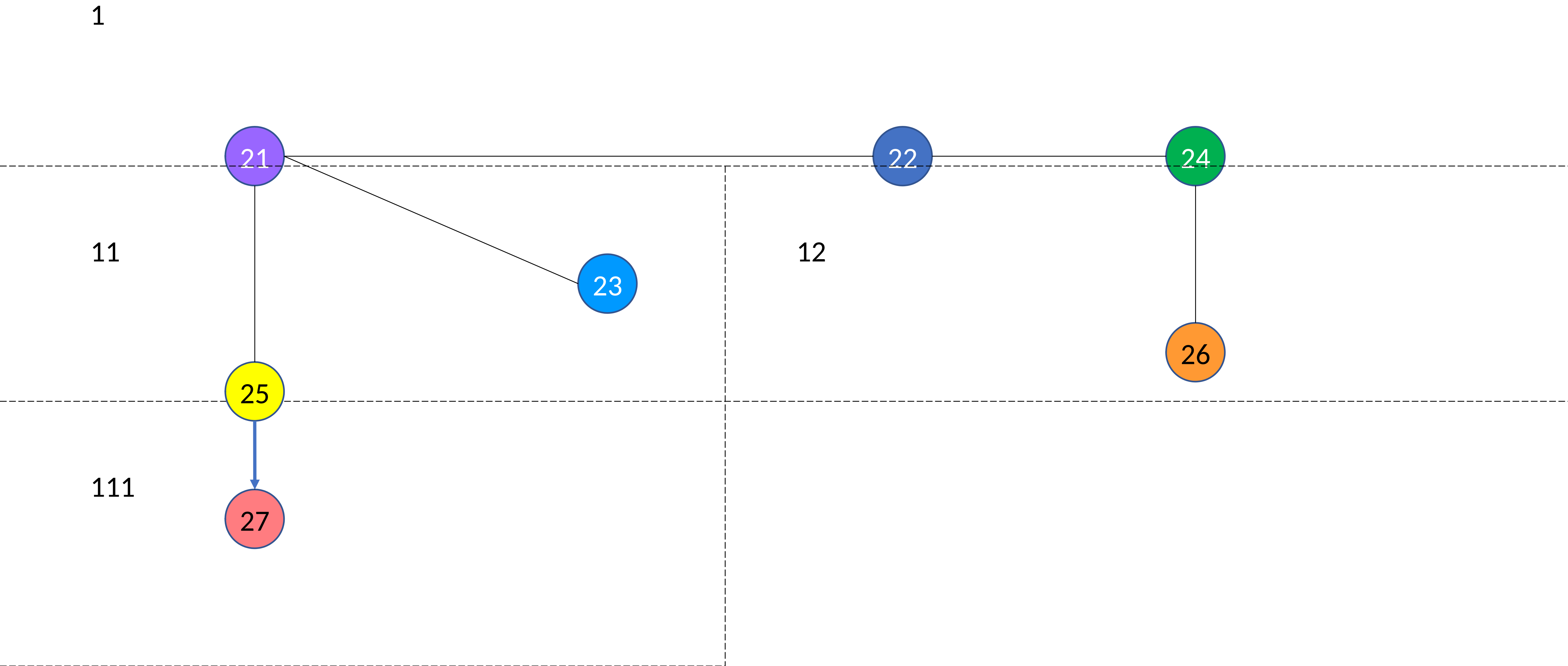


Send file from 26 to 27.

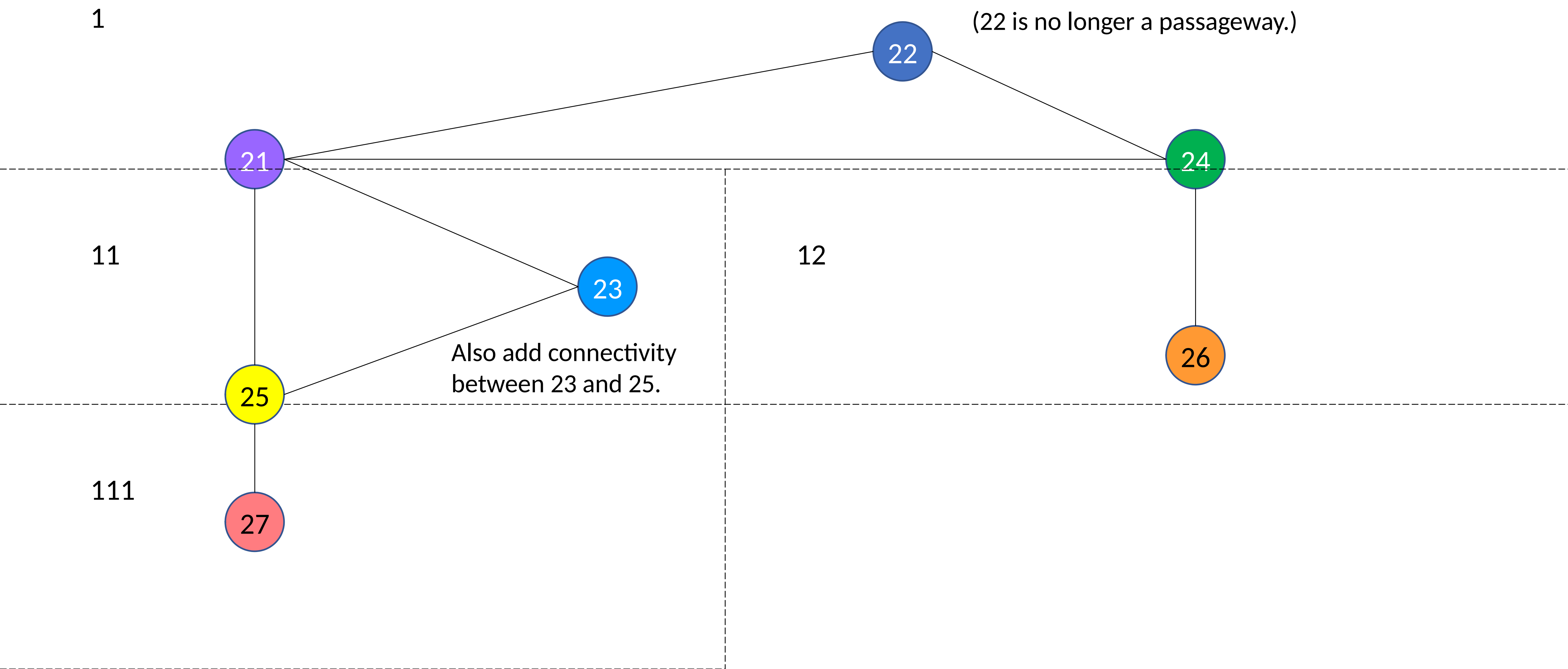



1
Passageway 24 will be preferred, as it is more direct.
21
22
24
11
23
12
26
25
111
27

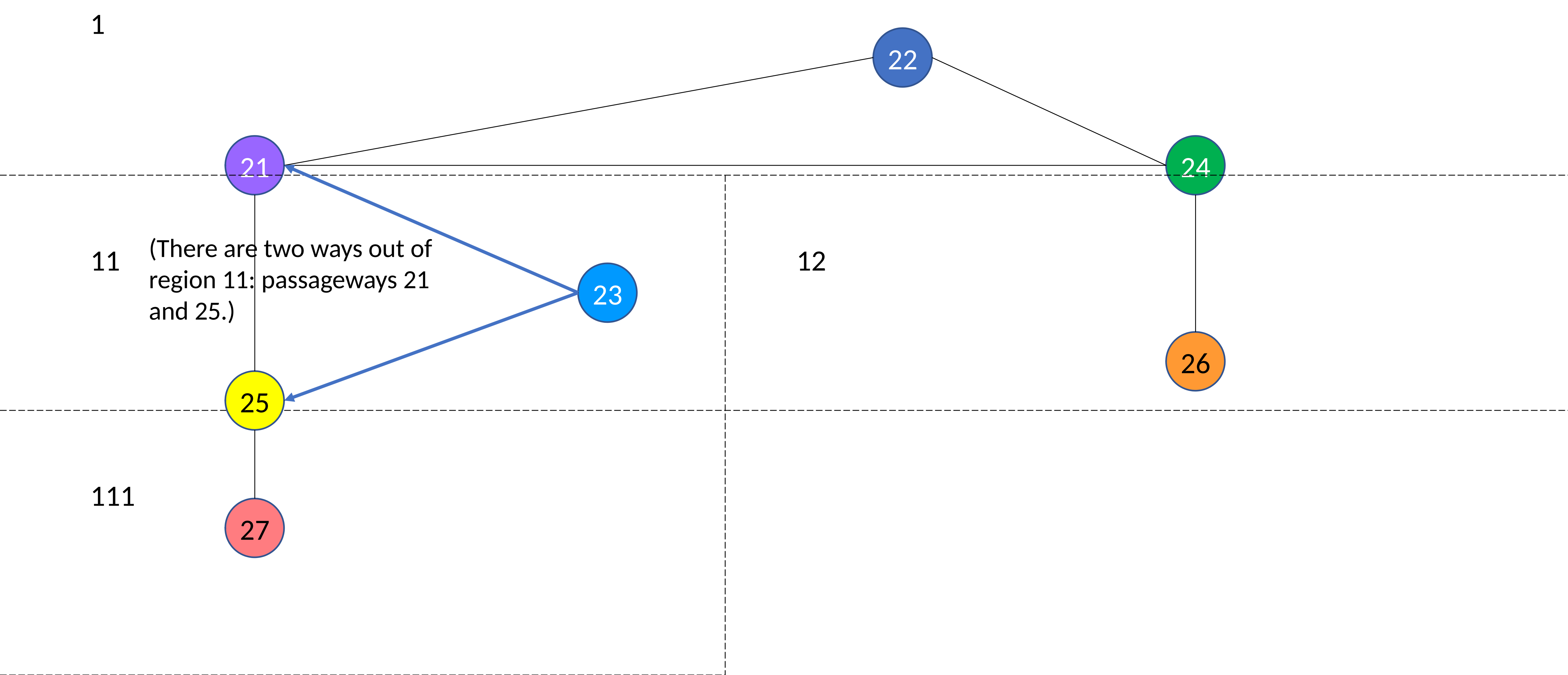

1
11
12
111
21
22
23
24
25
26
27


1
21
22
24
11
23
12
26
25
111
27

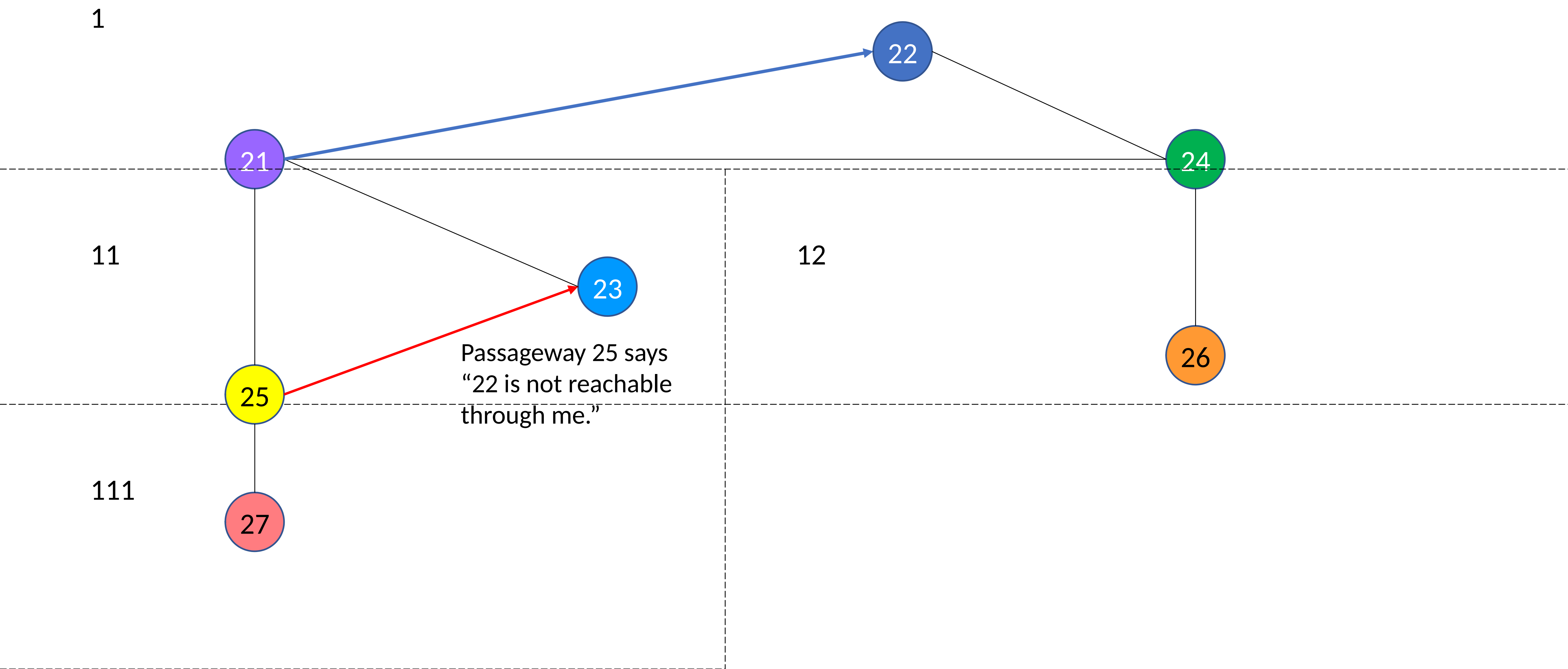
Remove 22 from region 12.



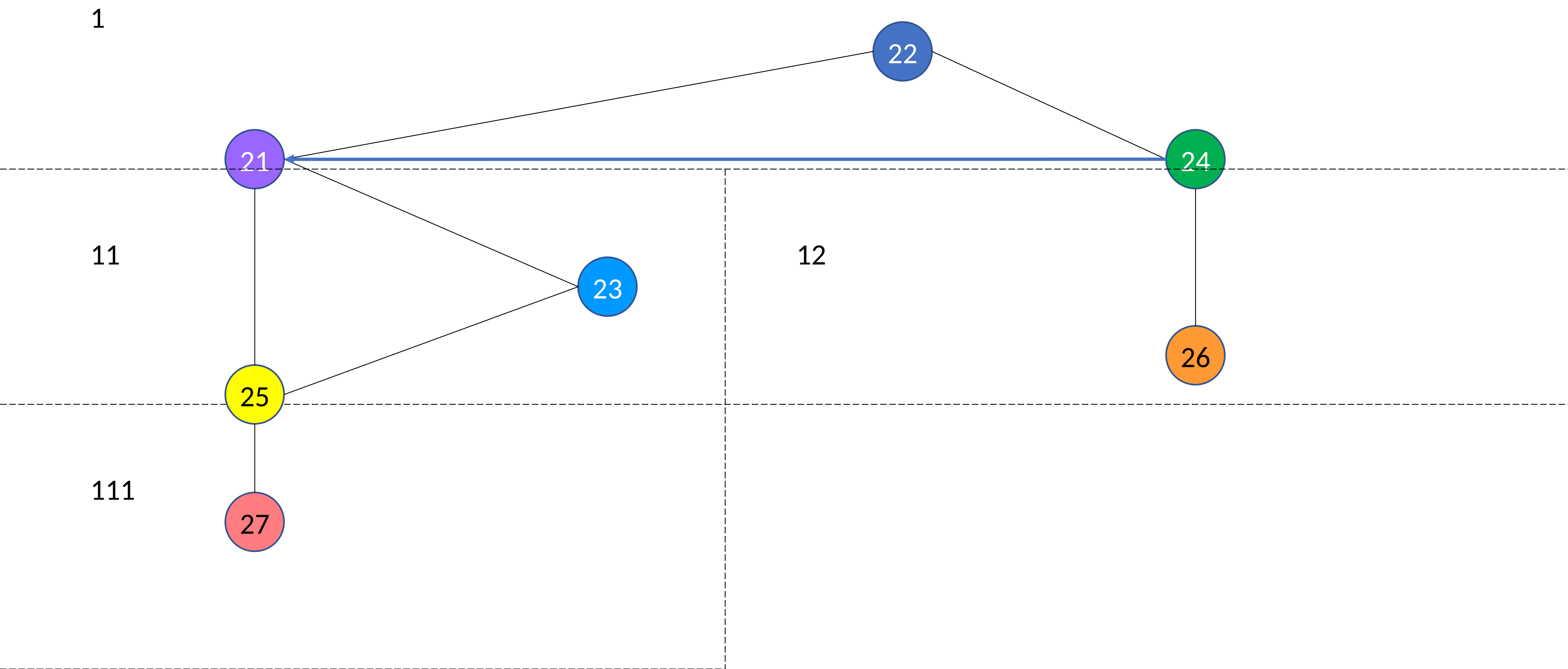
Send file from 23 to 22.

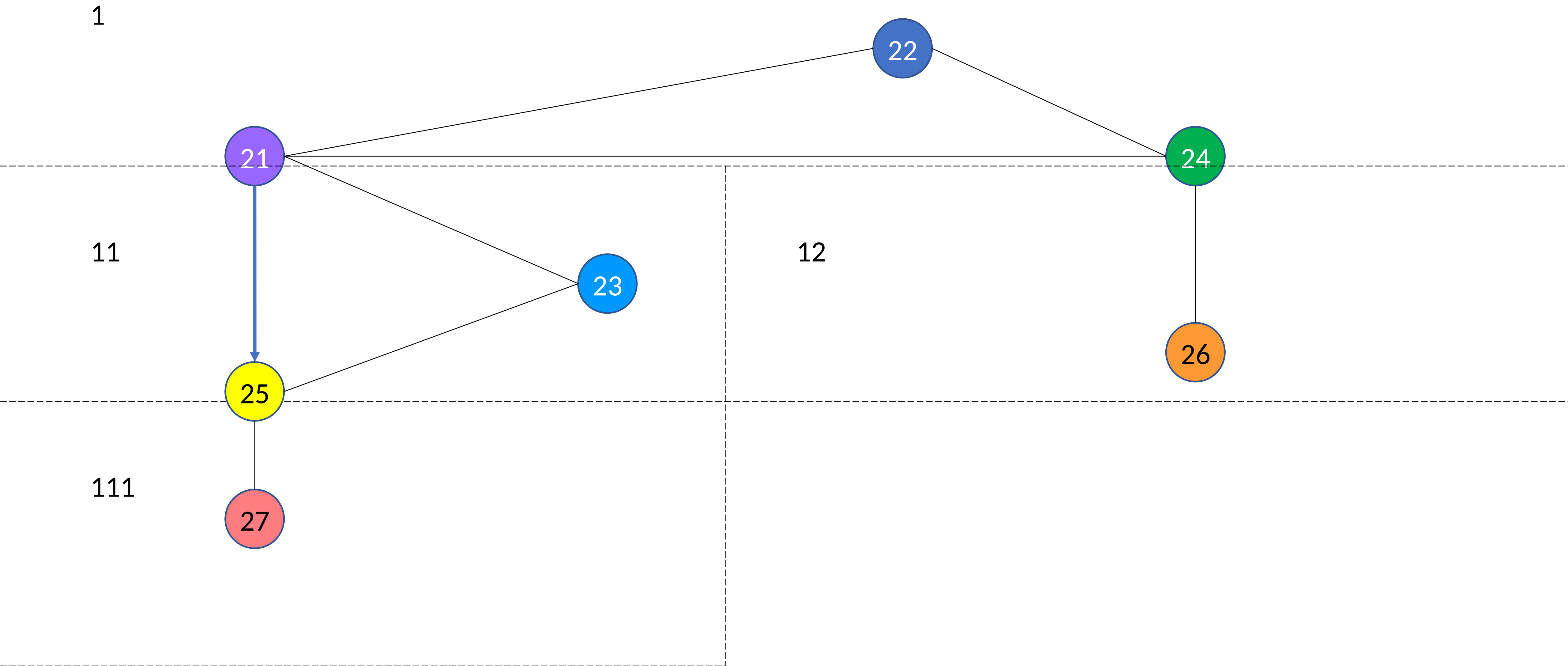


Discouraging feedback



And send file from 24 to 25.

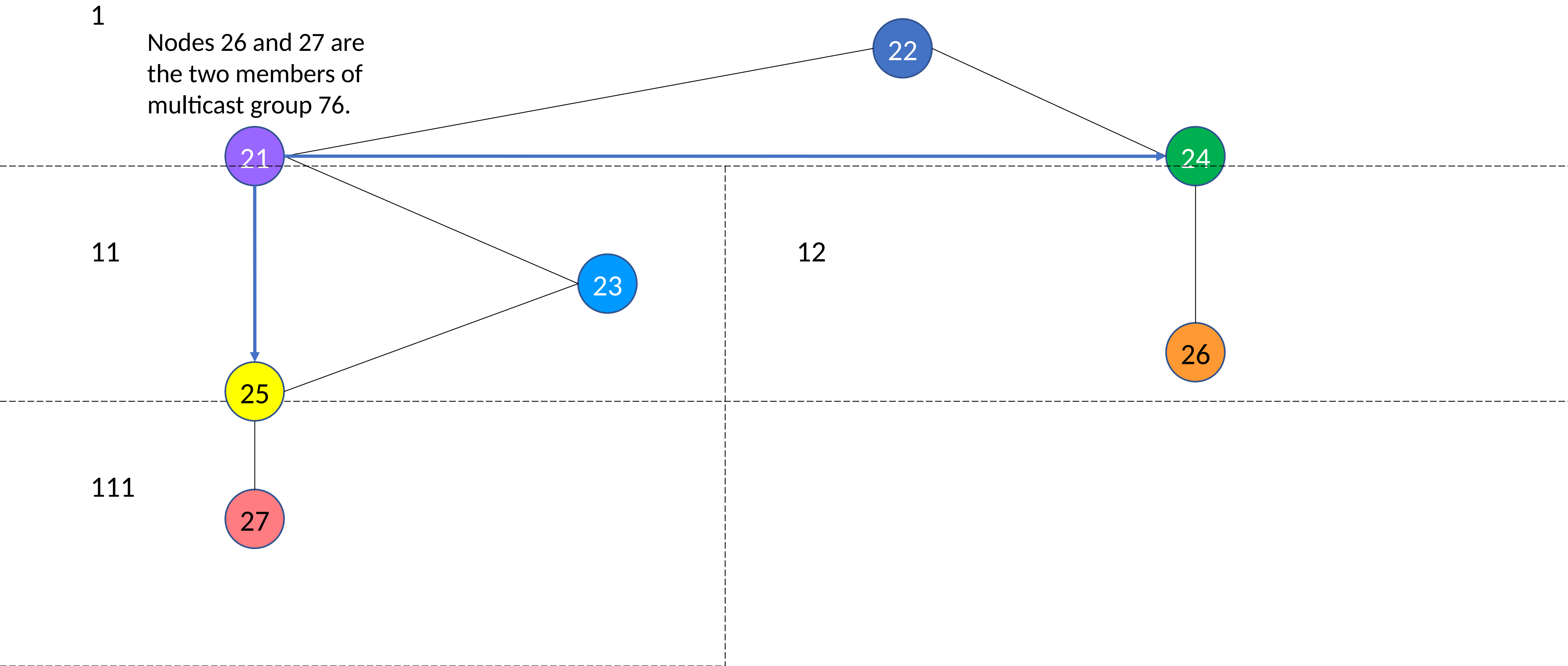


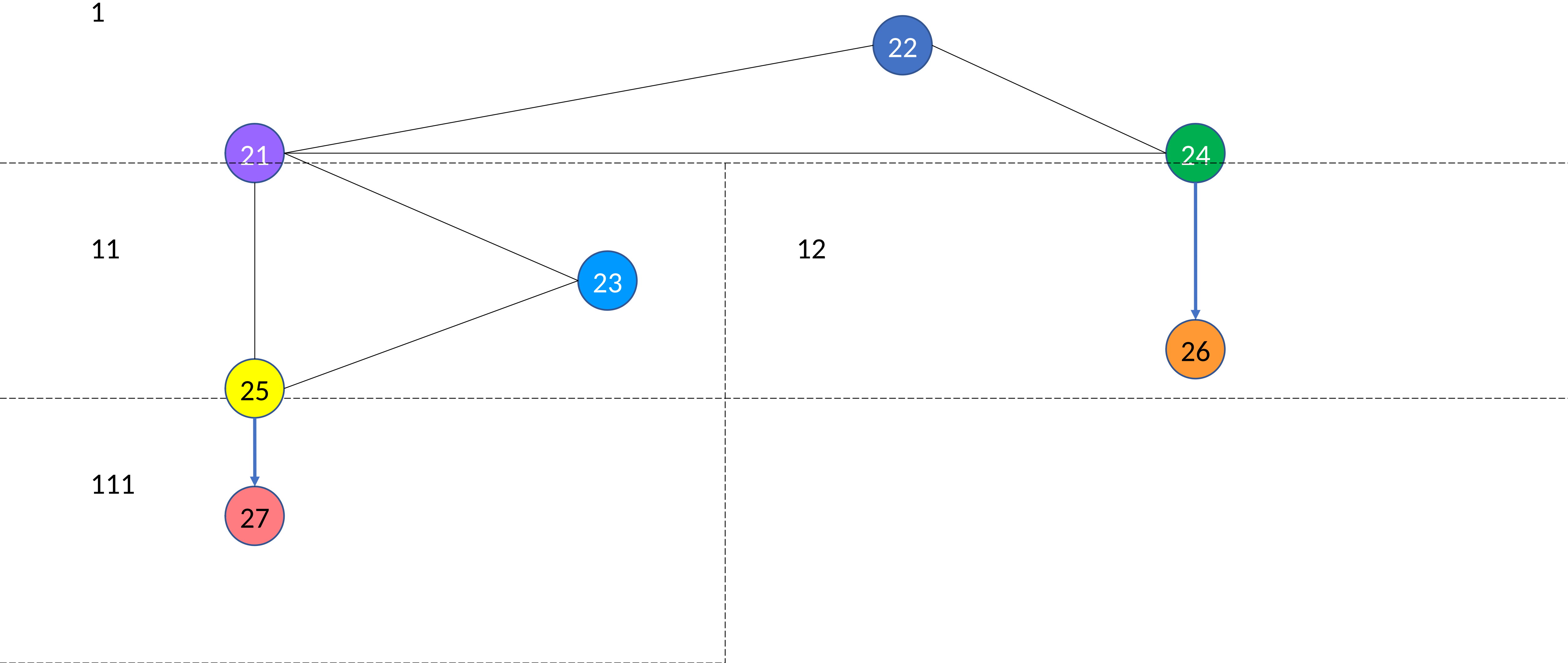


1
11
12
111
21
22
23
24
25
26
27

Last, multicast from 21 to multicast group 76.

Nodes 26 and 27 are the two members of multicast group 76.




1
22
21
24
11
12
23
26
25
111
27

A hypothetical use case (1 of 2)

- DNAC can instantiate a node in about 1 second, but transplanting the new node to its destination machine may take longer (needs time for SSH and SCP operations). Let's say 15 seconds per node.
- Assume we can have up to 33 nodes in each region.
- Assume we've identified (by host name) 29,000 machines that we want to include in our DTN network. These nodes could be instantiated in 879 regions.
- The first region will be fully populated after 8 minutes, and each of the nodes in that region can be the passageway to one of that region's sub-regions.

A hypothetical use case (2 of 2)

- Each of those 32 sub-regions will in turn be fully populated after another 8 minutes, and each of those 1024 new nodes can be a passageway to another subregion.
- An additional 846 sub-regions will be fully populated after another 8 minutes.
- So a fully interconnected, BPSec-secured delay-tolerant network of 29,000 nodes can be easily deployed in 24 minutes.
  - In this particular scenario, no node would be more than 5 forwarding hops from any other.