

Routing Algorithm

Let the packets flow ...

Dr. Kiran M
IT Dept., NITK

Previous Session

- IPv6
- How to compress the v6 addresses
- Consecutive parts of v6 address containing zeros are replaced with ::
- It applies only for zeros
- IPv4 address in IPv6
 - ::ffff : [IPv4 address](#)
 - 80 bits , 16 bits and 32 bits
- Routing Algorithms
 - Correct, Robust, Quick and Fair enough.

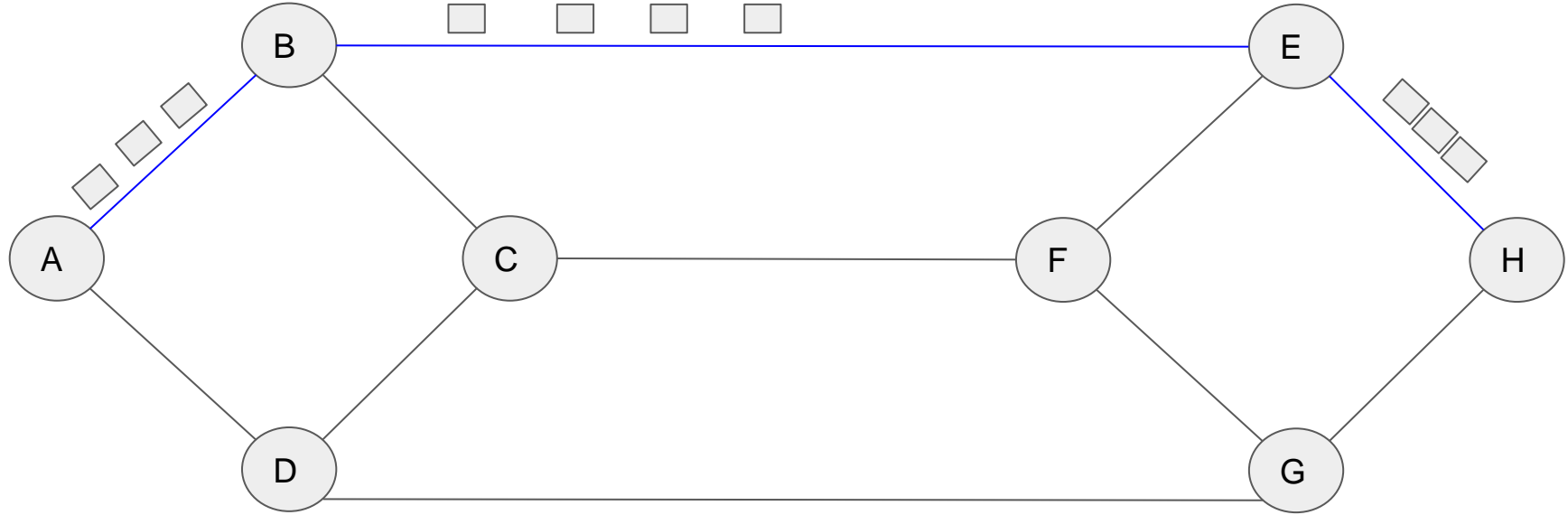
Two types of routing algorithm

- Non Adaptive
 - Static Routing algorithm
 - Once the route is setup, done.
 - If congestion happens during the transmission, path will break up, then another path will be discovered.
 - Route rediscover latency, pause in data transmission, retransmission, waste of resources, waiting time.
 - Performance degrades.

- Adaptive

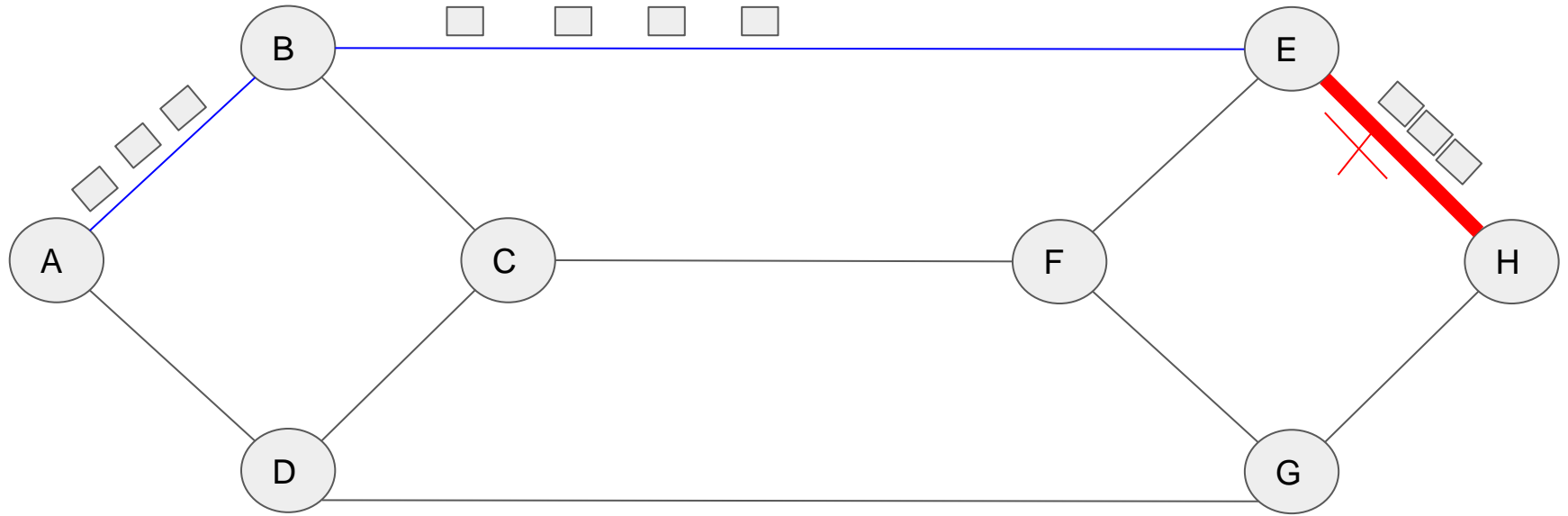
- Dynamic Routing algorithm.
- Can sense the danger, and can adapt accordingly.
- Before path breaks up due to congestion, a new path will be chosen
- No transmission hiccups.
- Better performance.

Non Adaptive

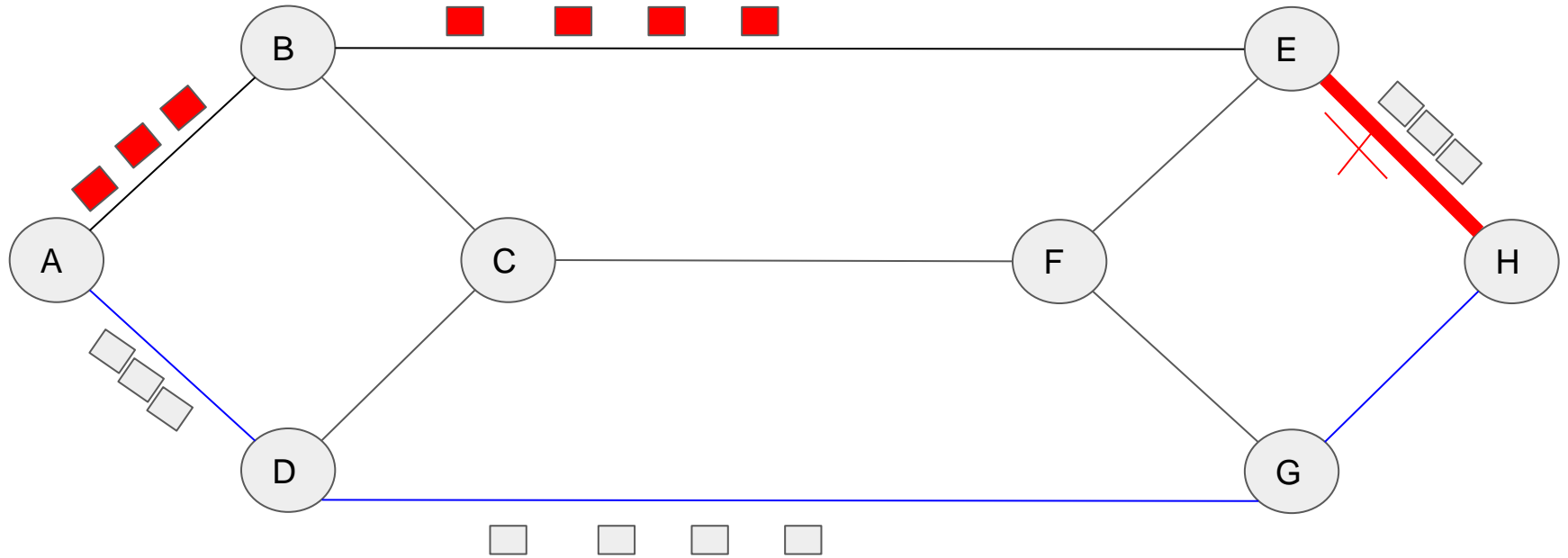


A is source and H is the destination. Start with A

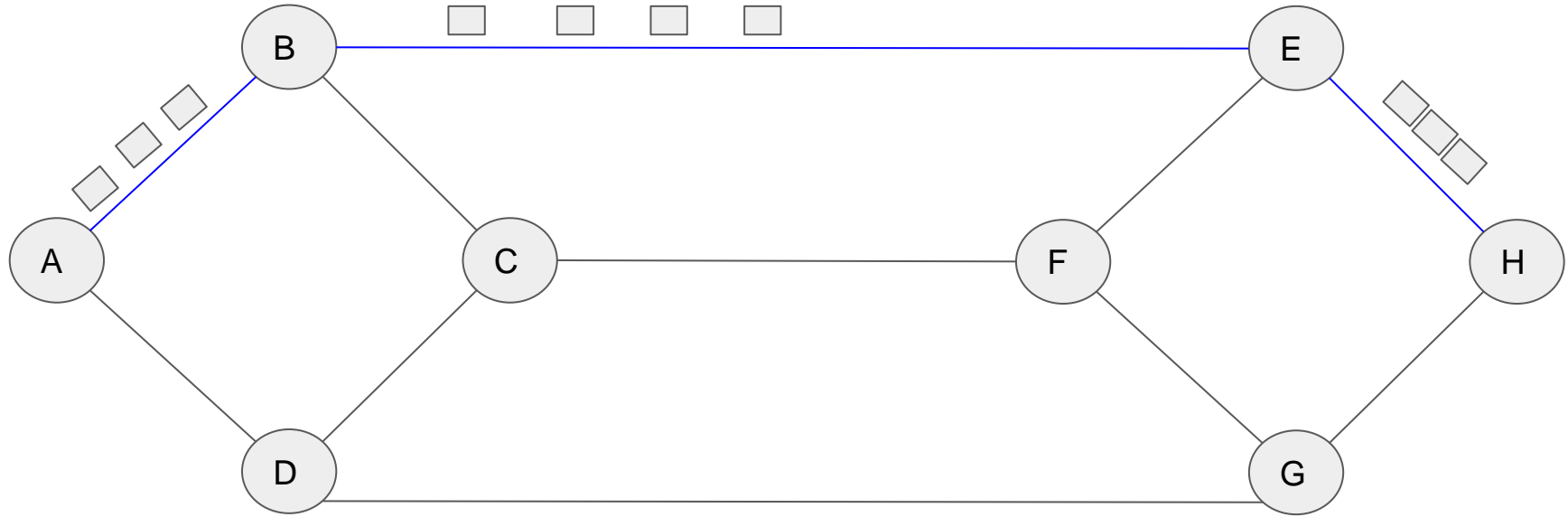
Non Adaptive



Non Adaptive

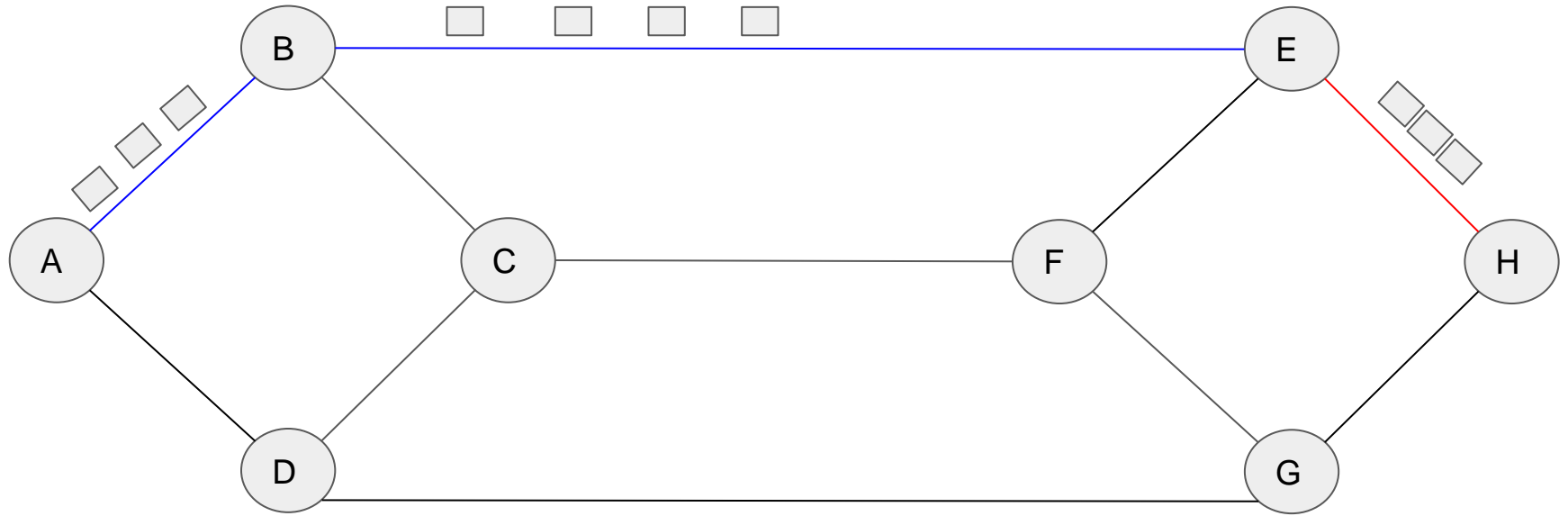


Adaptive

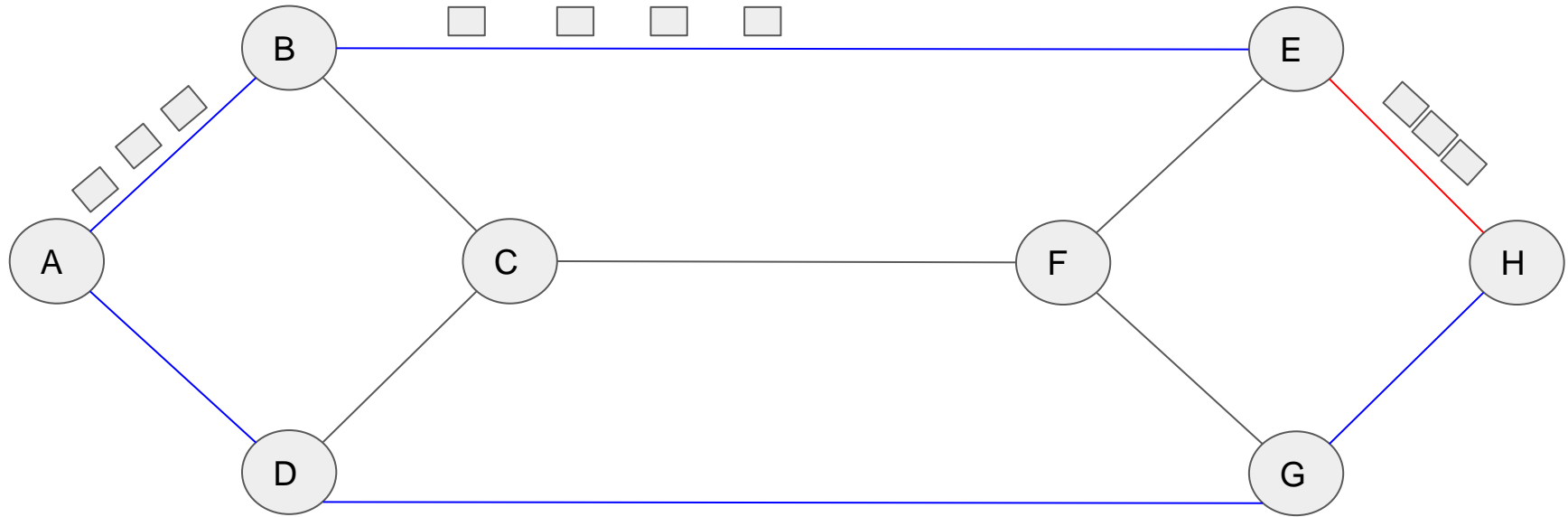


A is source and H is the destination. Start with A

Adaptive



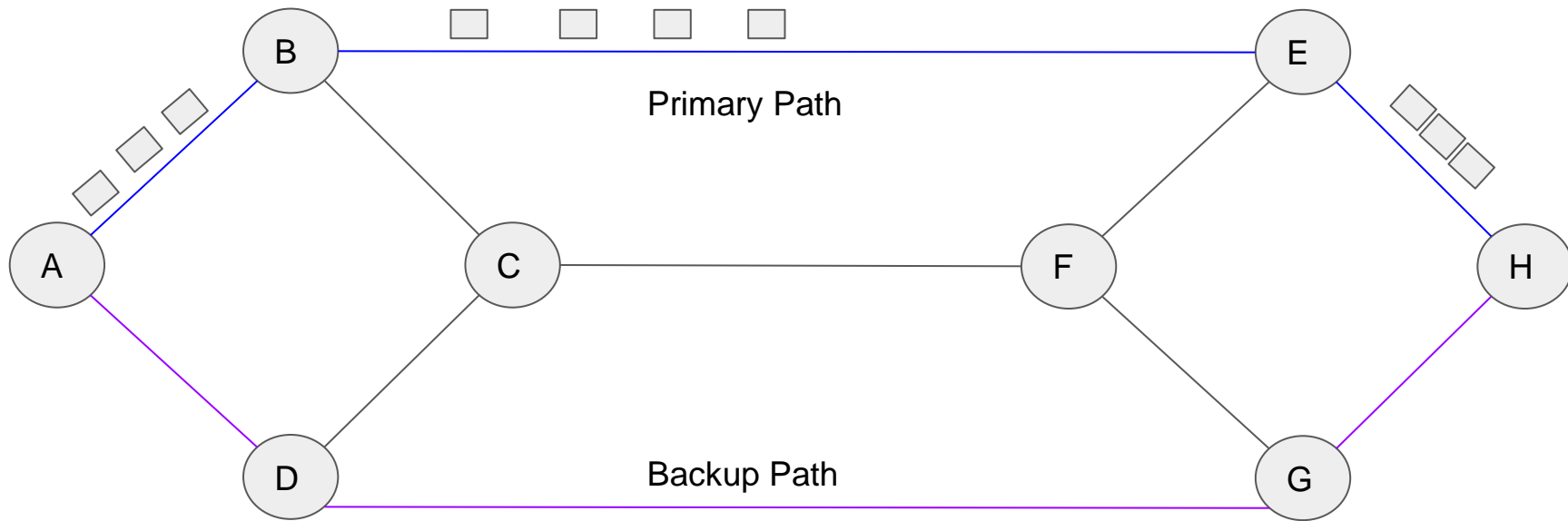
Adaptive



Before condition worsens, new path will be chosen and will be activated.

Routing Algorithms

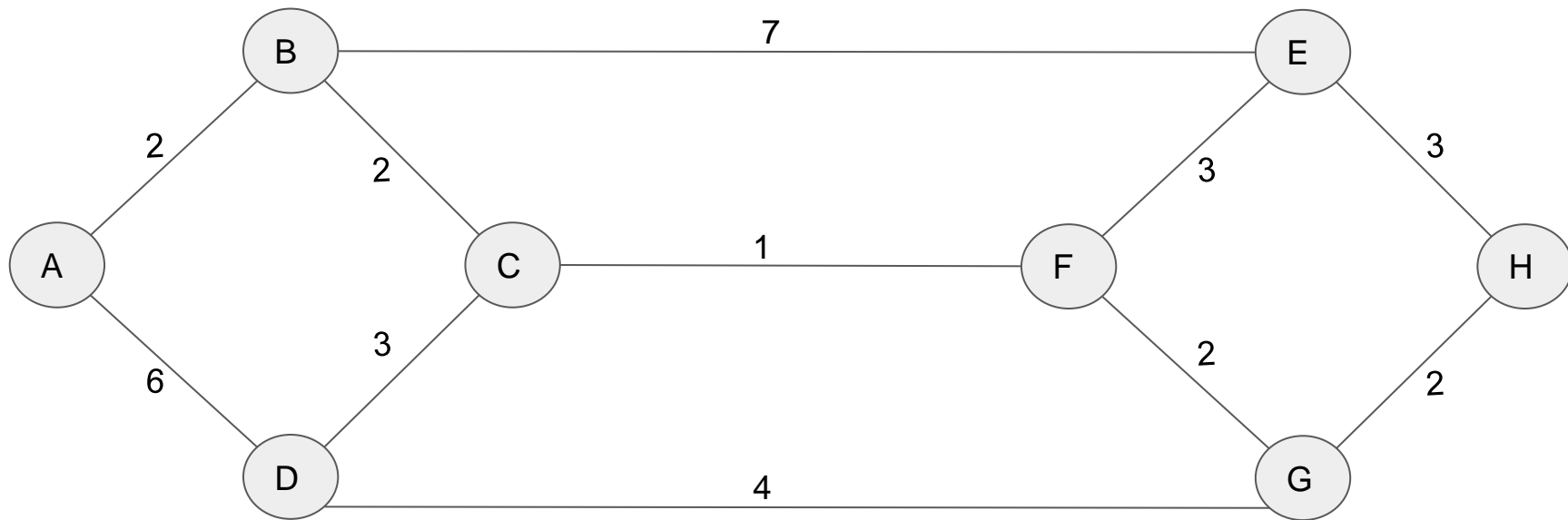
- Single Path
 - Only one path will be maintained between the source and the destination
- Multipath
 - Multiple paths will be maintained between the source and the destination.



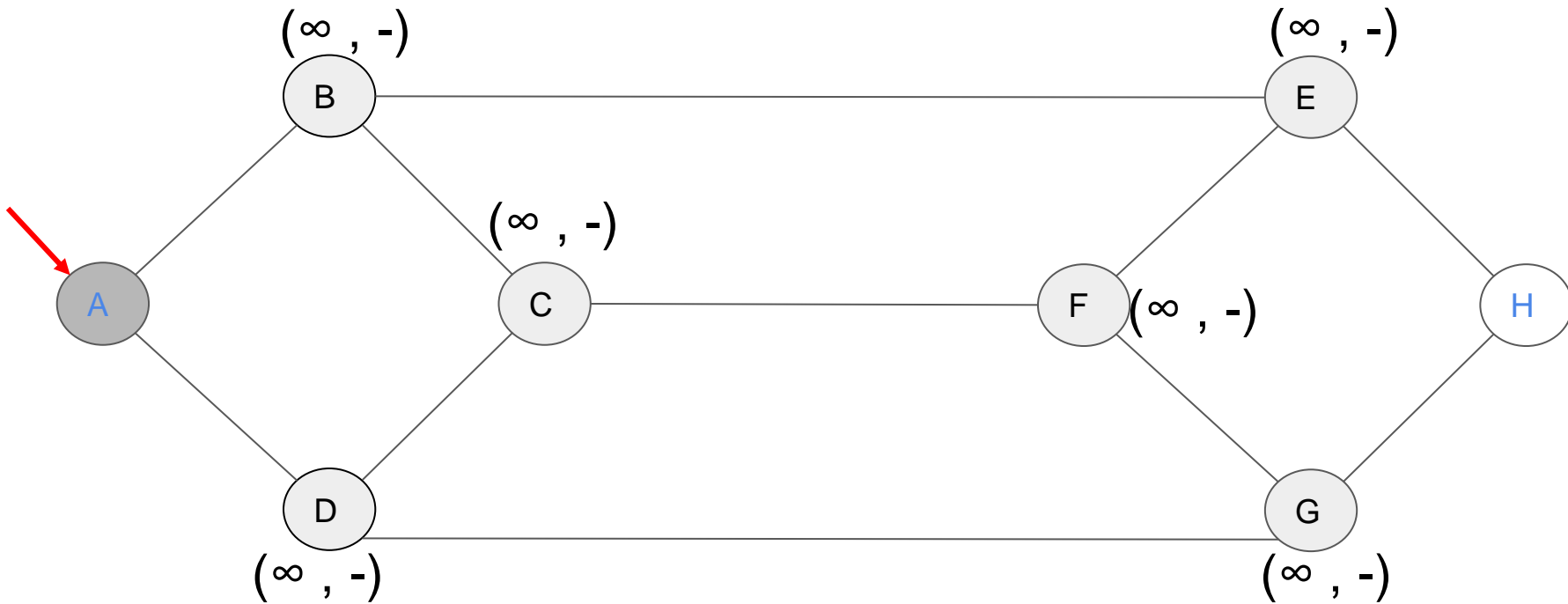
- Routing Parameters (Metrics)
 - Average Traffic
 - Bandwidth
 - E-2-E delay
 - Queue Status
- Performance Parameters.
 - Throughput, E-2-E delay, routing overhead, (re) route discovery latency,
- Shortest Path may not work always.

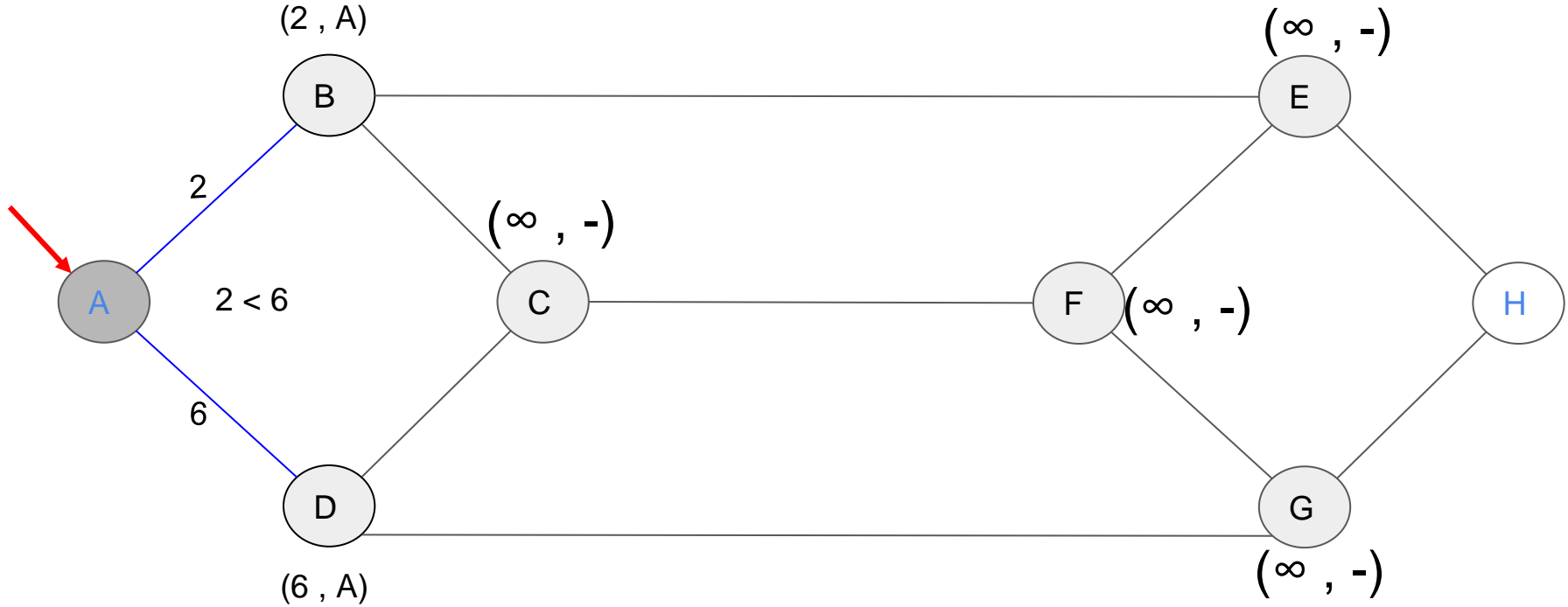
Dijkstra Algorithm (1959)

- Each link will be labelled.
- Initially No Paths will be known.
 - Hence, all nodes are labelled with infinity.
 - Label value will change over the time.



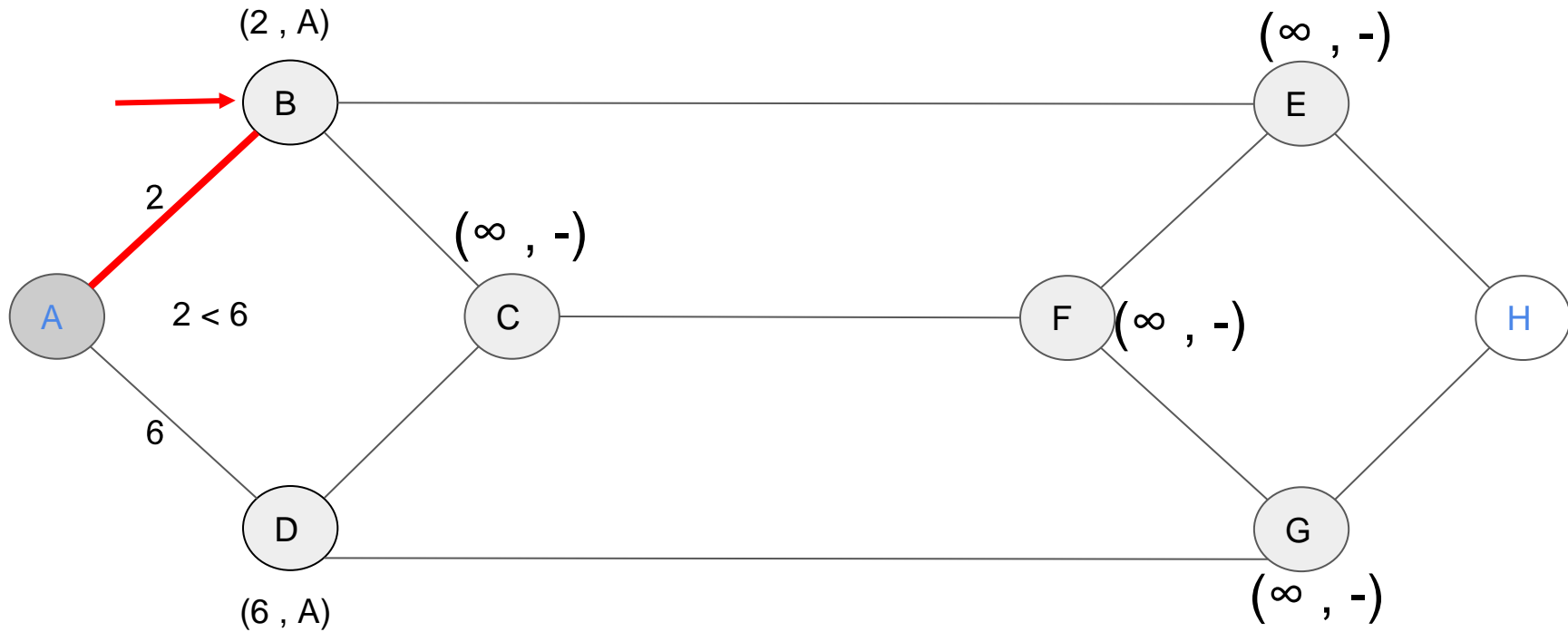
A is source and H is the destination. Start with A

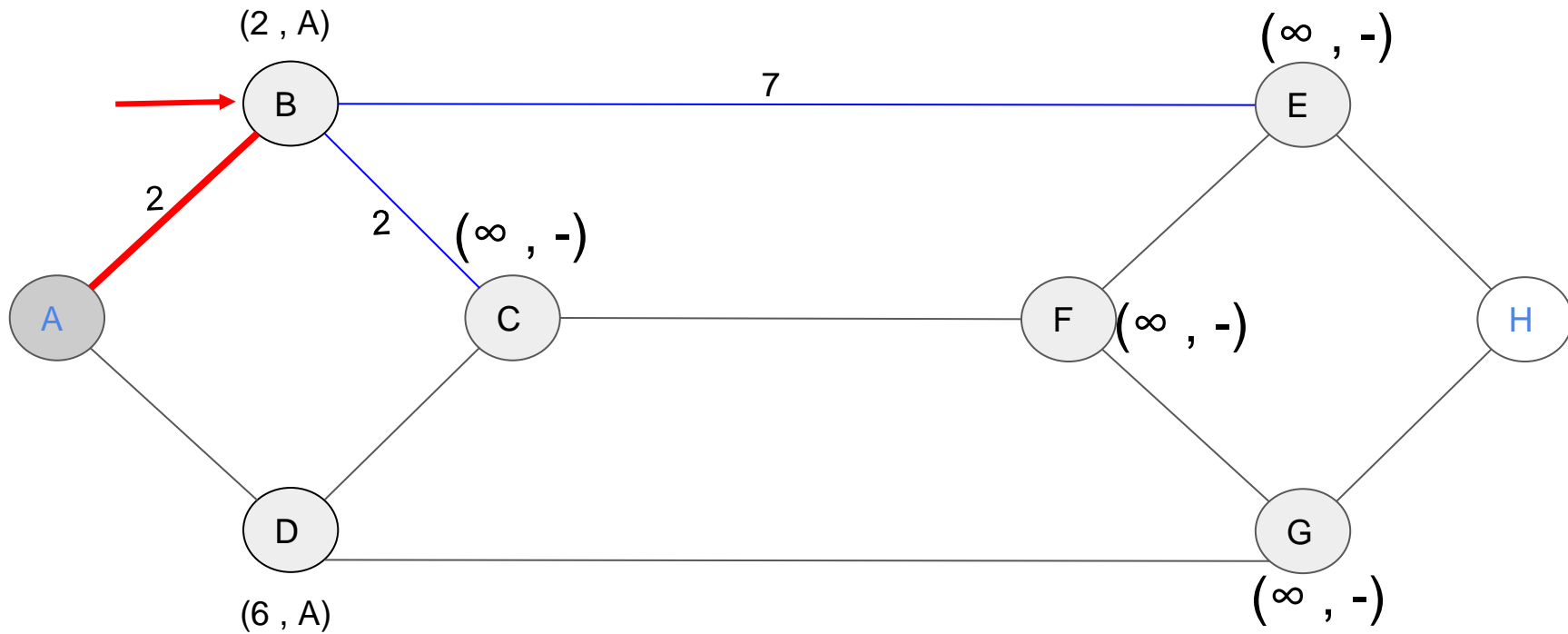




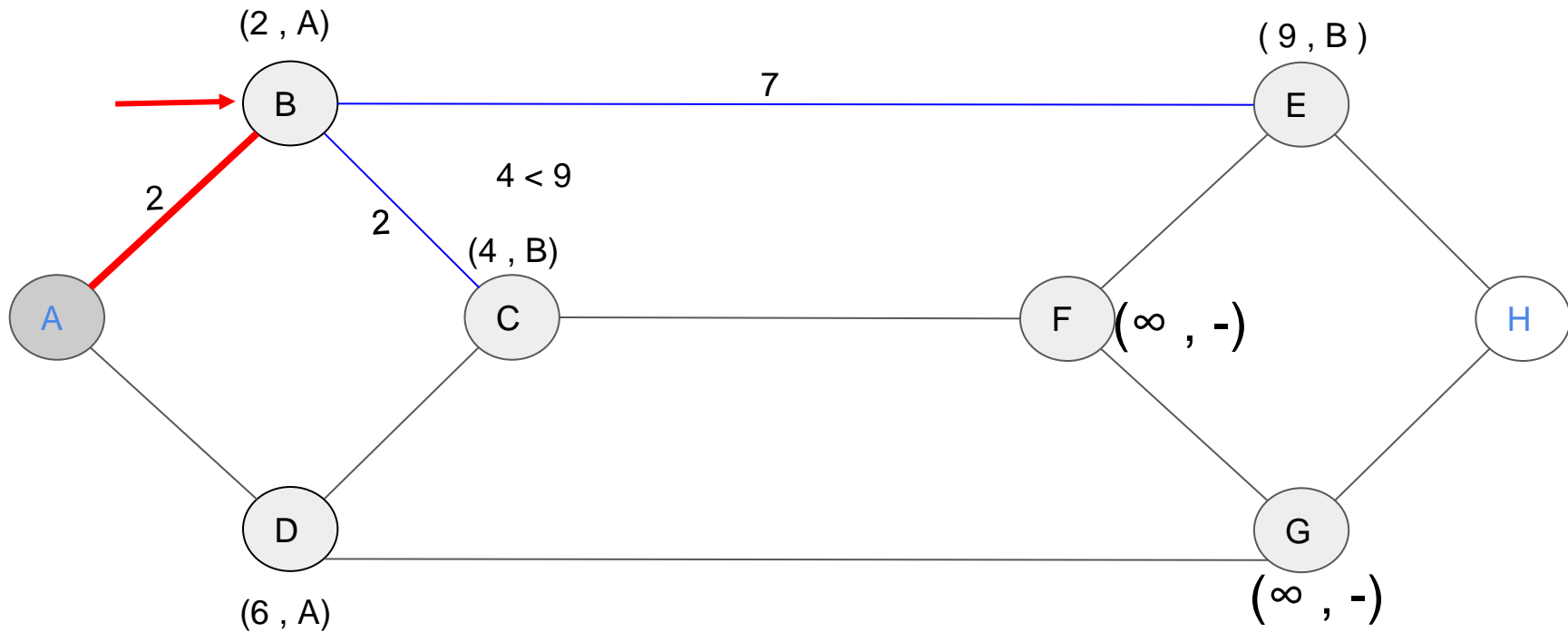
For A, neighbors (adjacent nodes) are B (2) and D(6).

Re Label Them.

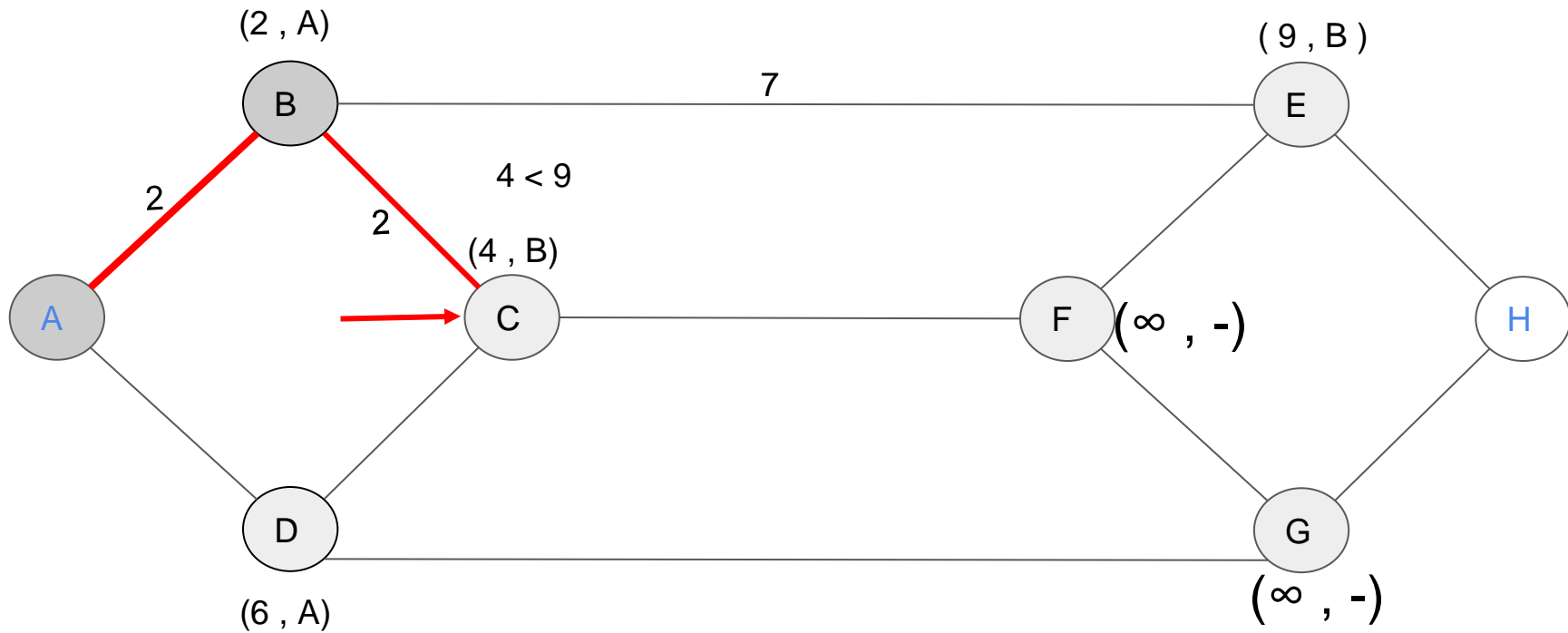




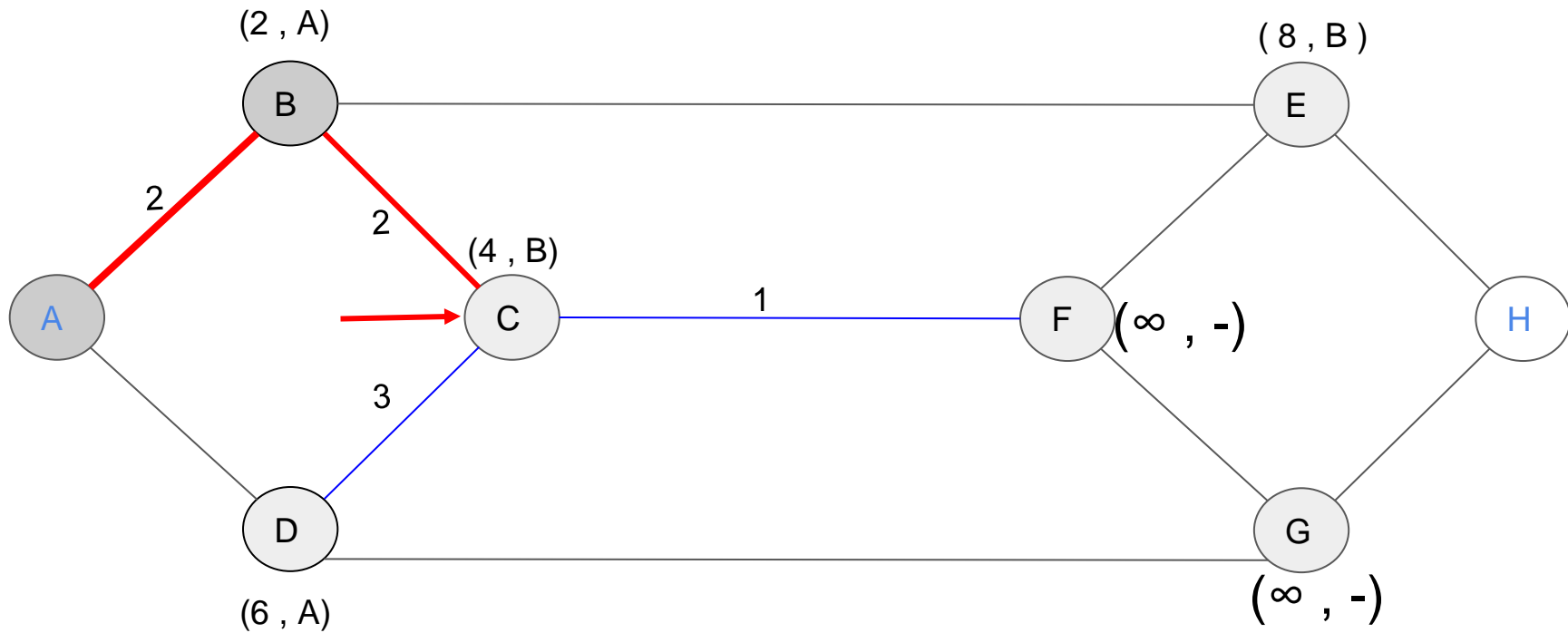
For B, neighbors (adjacent nodes) are E (7) and C(2).



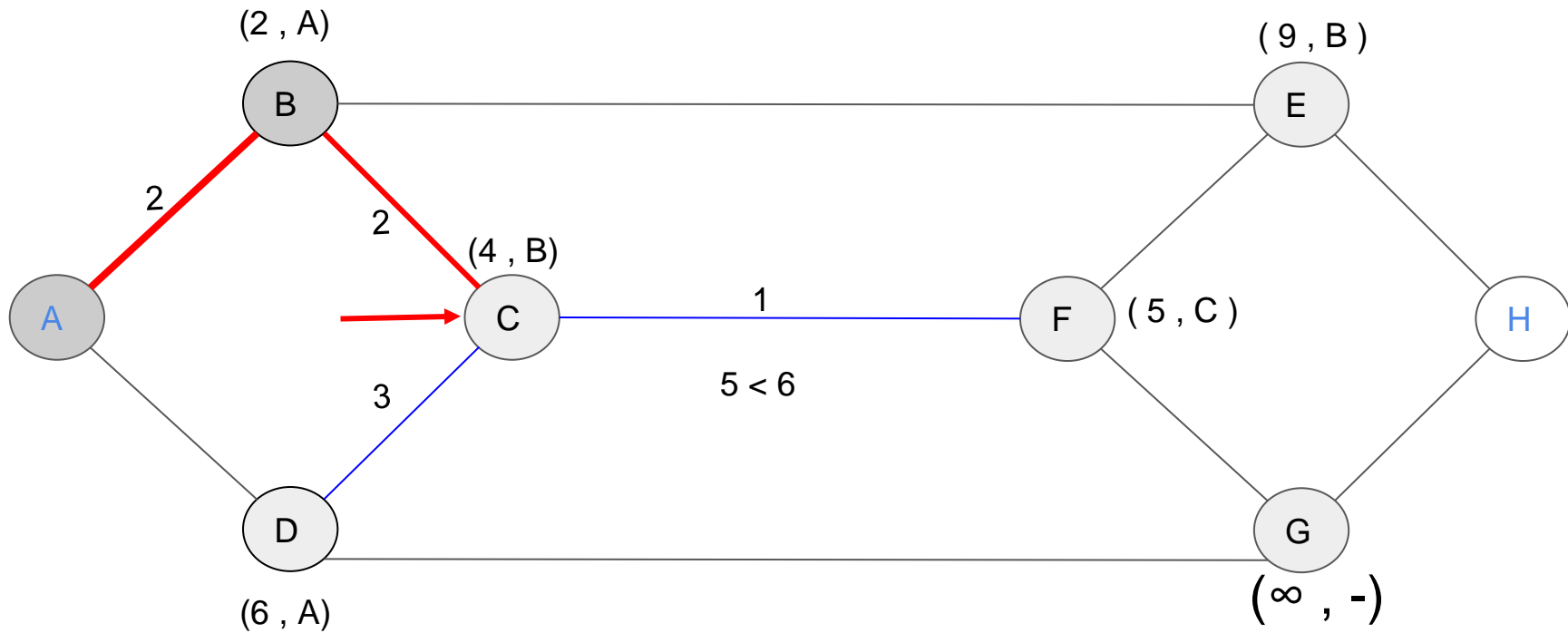
For B, neighbors (adjacent nodes) are E (7) and C(2).



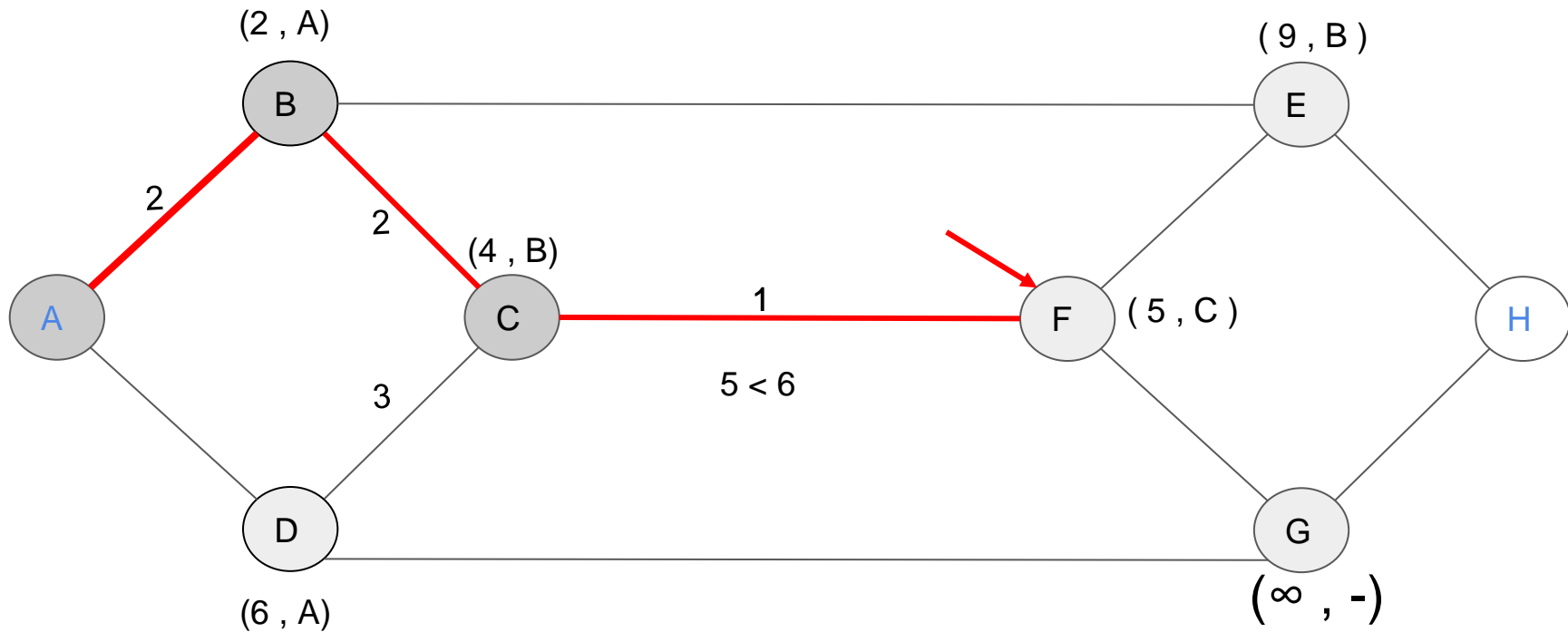
For B, neighbors (adjacent nodes) are E (7) and C(2).



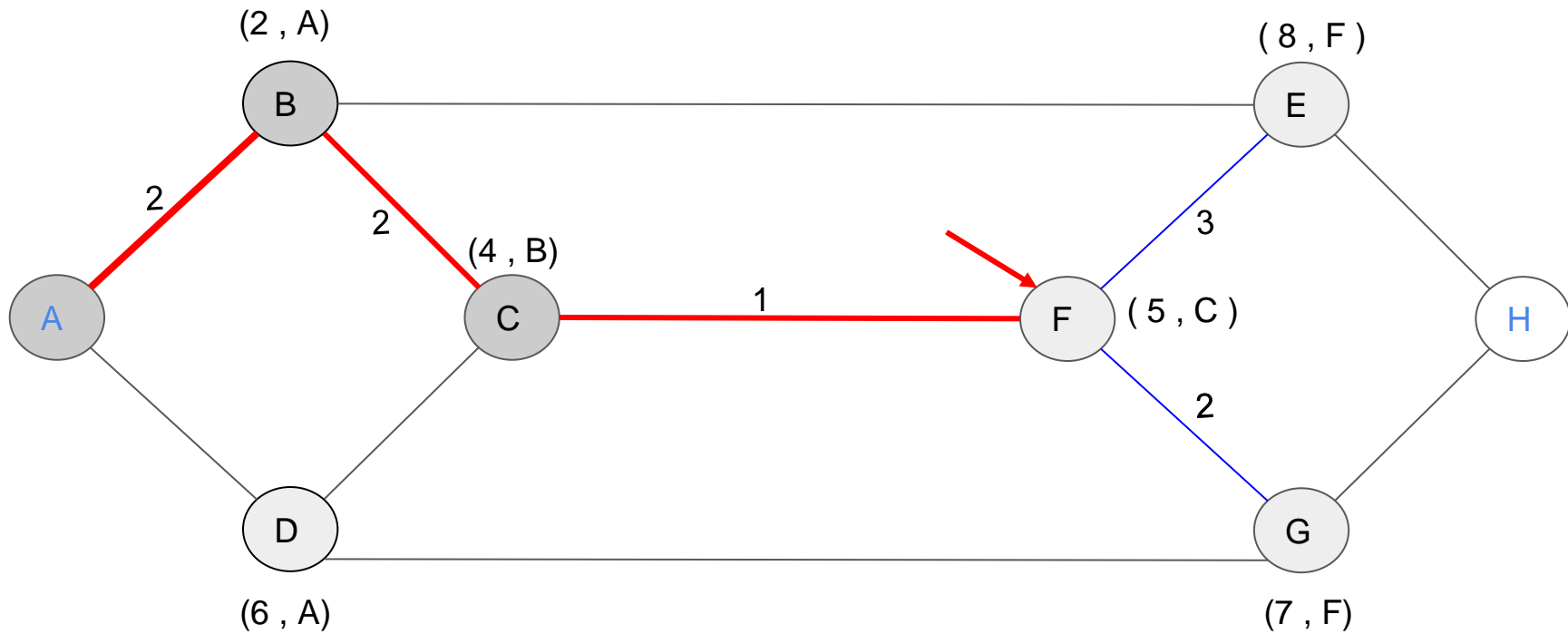
For C, neighbors (adjacent nodes) are D (3) and F(2).



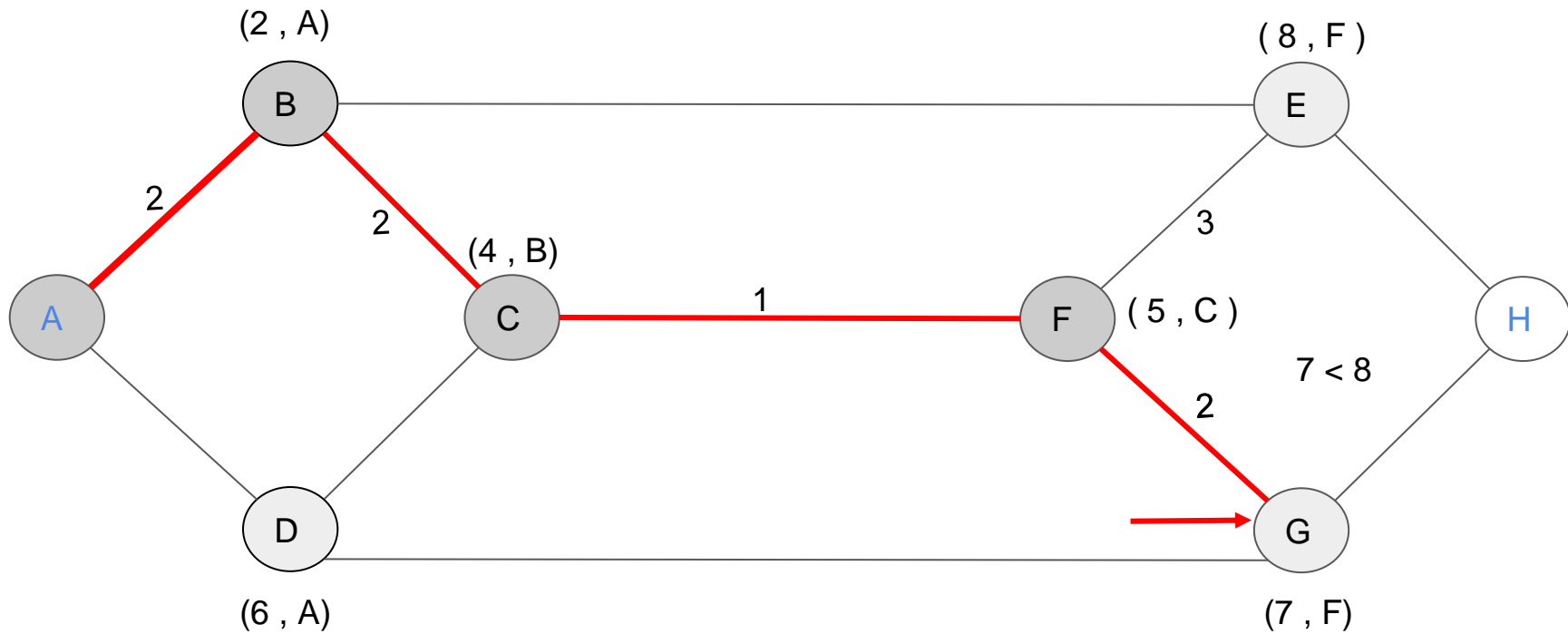
For C, neighbors (adjacent nodes) are D (3) and F(2).



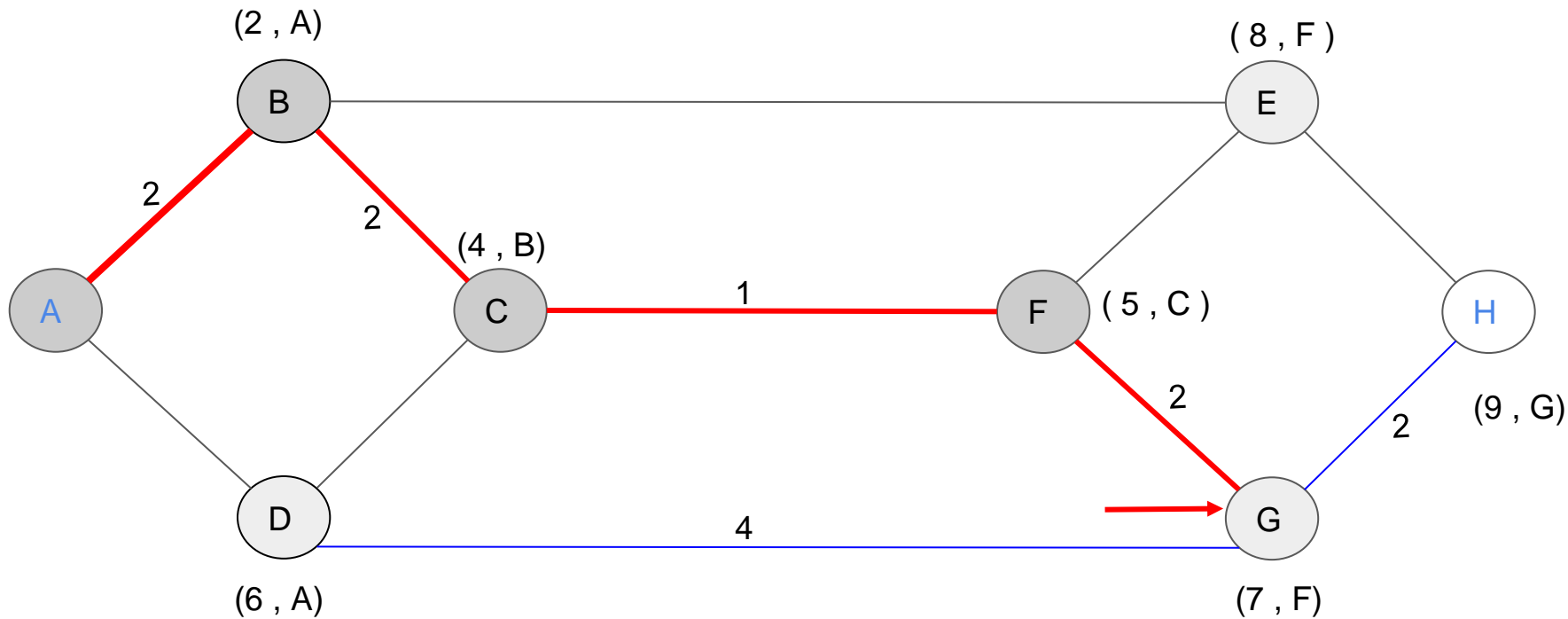
For C, neighbors (adjacent nodes) are D (3) and F(2).



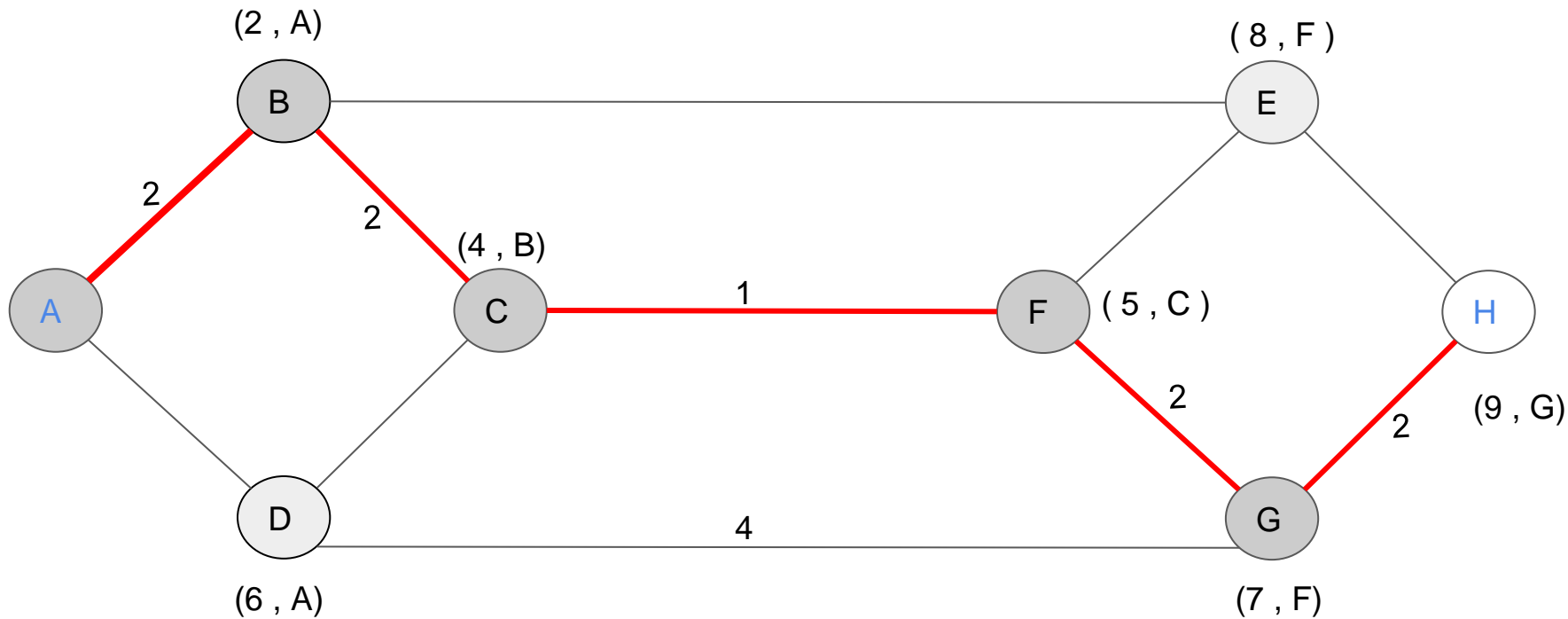
For F, neighbors (adjacent nodes) are E (3) and G(2).



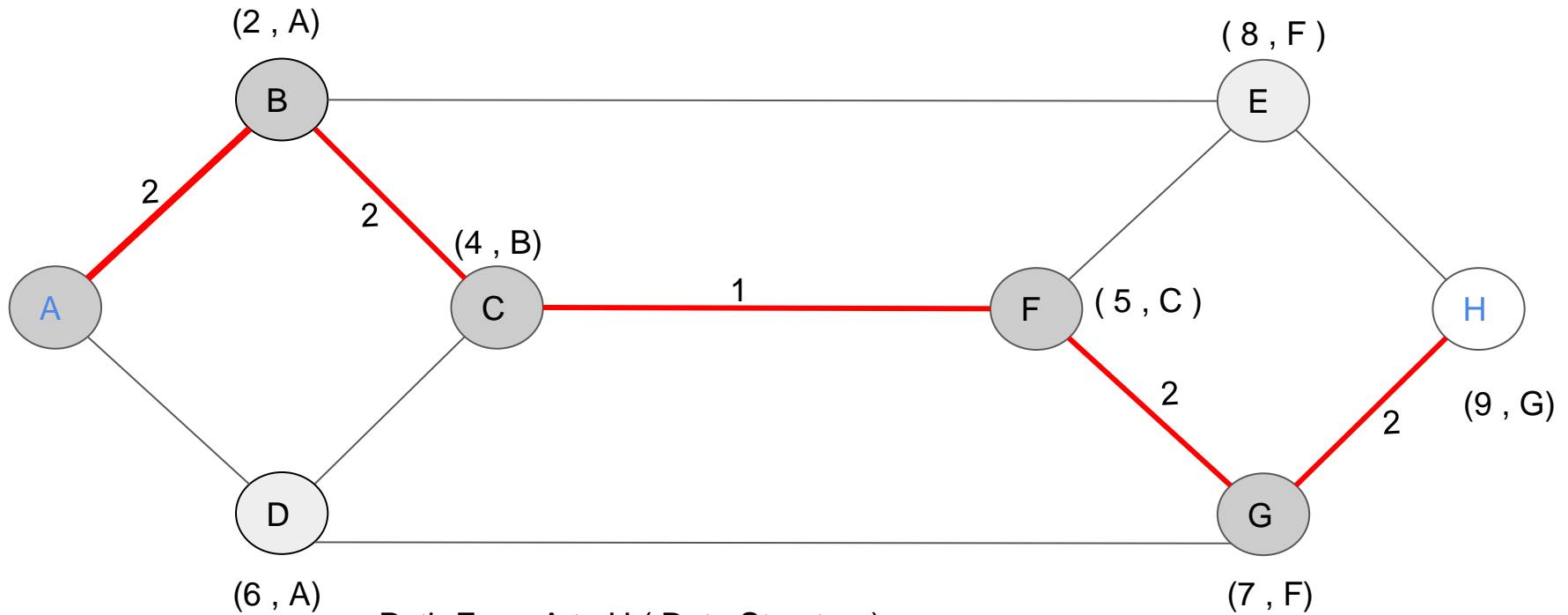
For F, neighbors (adjacent nodes) are E (3) and G(2).

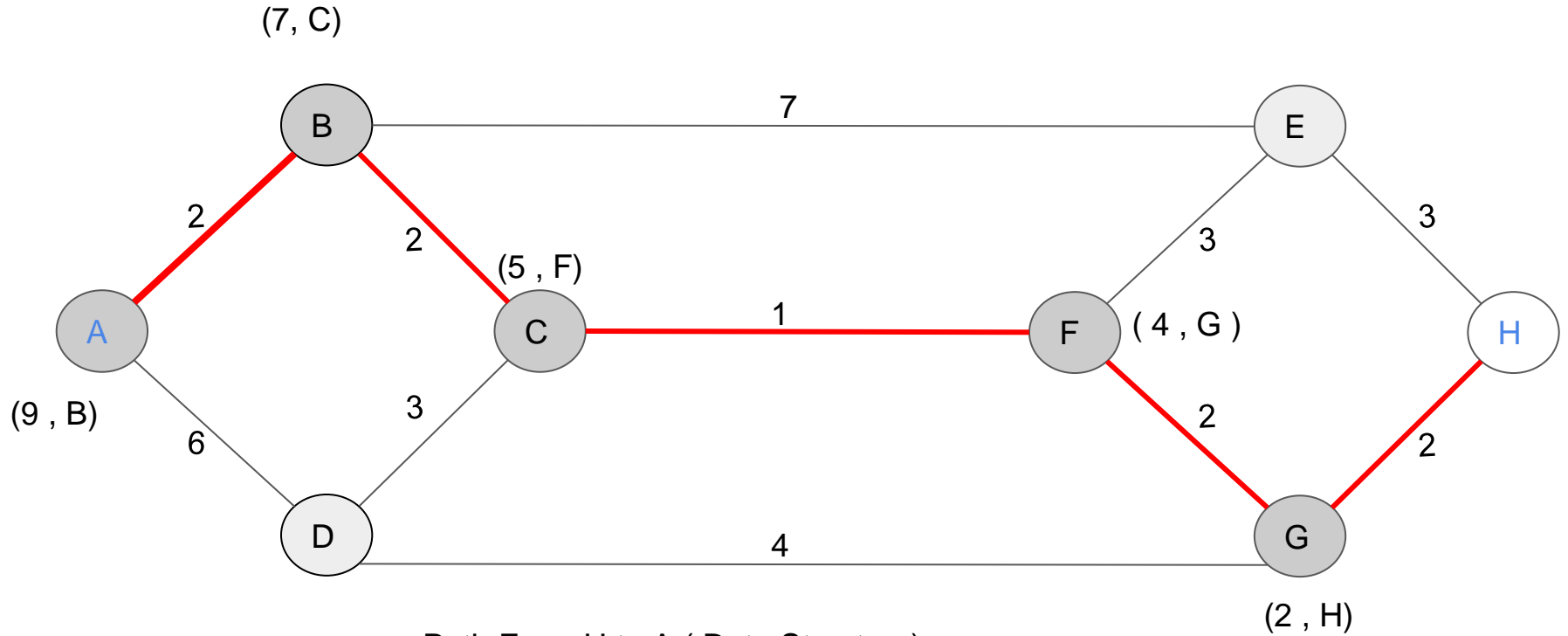


For G, neighbors (adjacent nodes) are D (4) and H (2).



For G, neighbors (adjacent nodes) are D (4) and H (2).





Path From H to A (Data Structure)

H -> G -> F -> C -> B -> A.

Total Cost $2 + 2 + 1 + 2 + 2 = 9$

Path From H to A (In CN Route is opposite)

A -> B -> C -> F -> G -> H.

Total Cost $2 + 2 + 1 + 2 + 2 = 9$