IN2010 Gruppe 4

Uke 7 - Grafer: 2-Sammenhengende grafer og Sammenhengende komponenter

Bli med :)

Dagens Plan

- → Info
- → Pensum-gjennnomgang
- → Gruppeoppgaver

Samretting

4

Innlevering 3

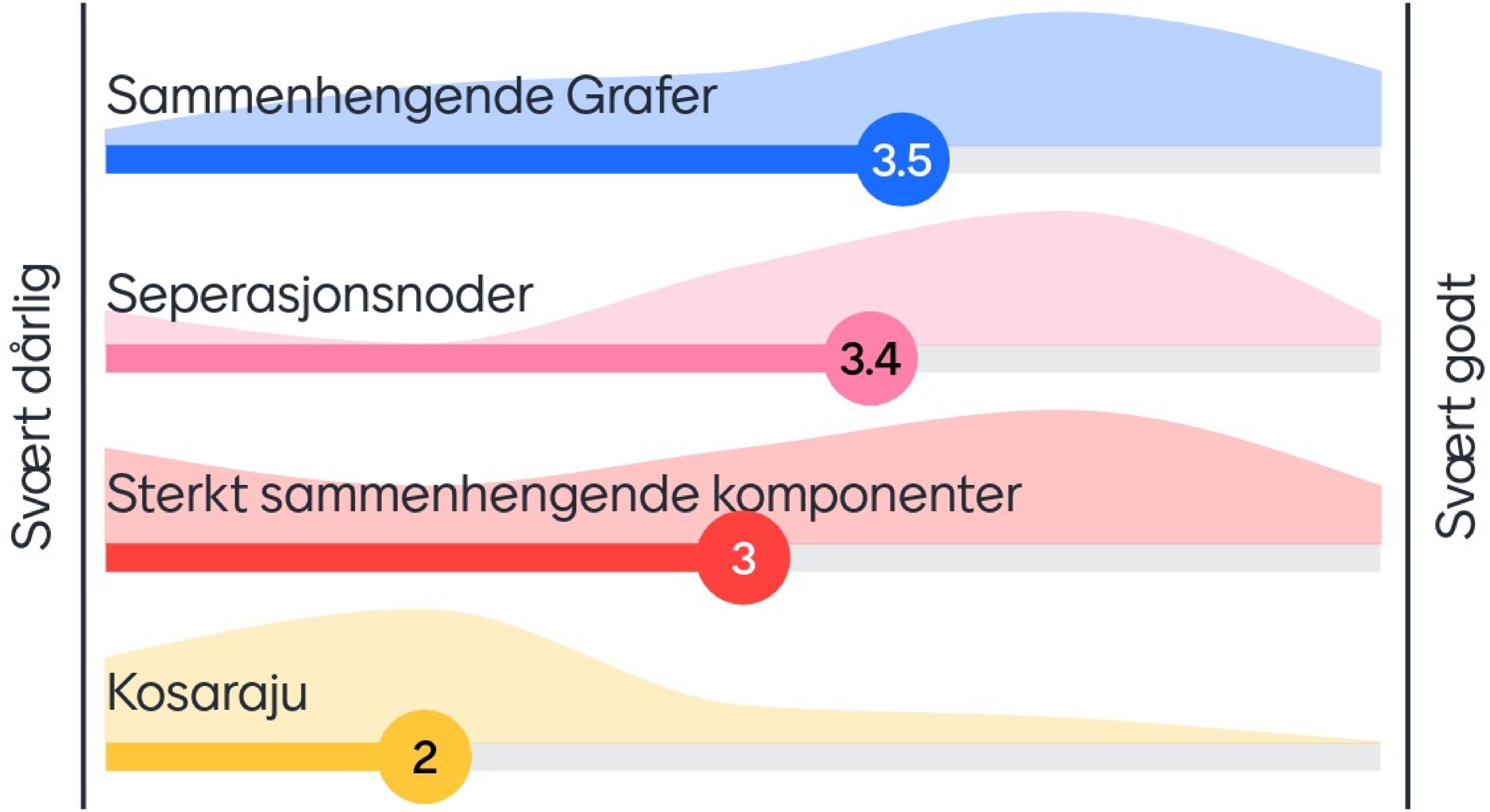
https://www.uio.no/studier/emner/matnat/ifi/IN2010/h23/innleveringer/innlevering3.pdf

Repitisjonsuke

Neste uke :)

Pensumgjennomgang

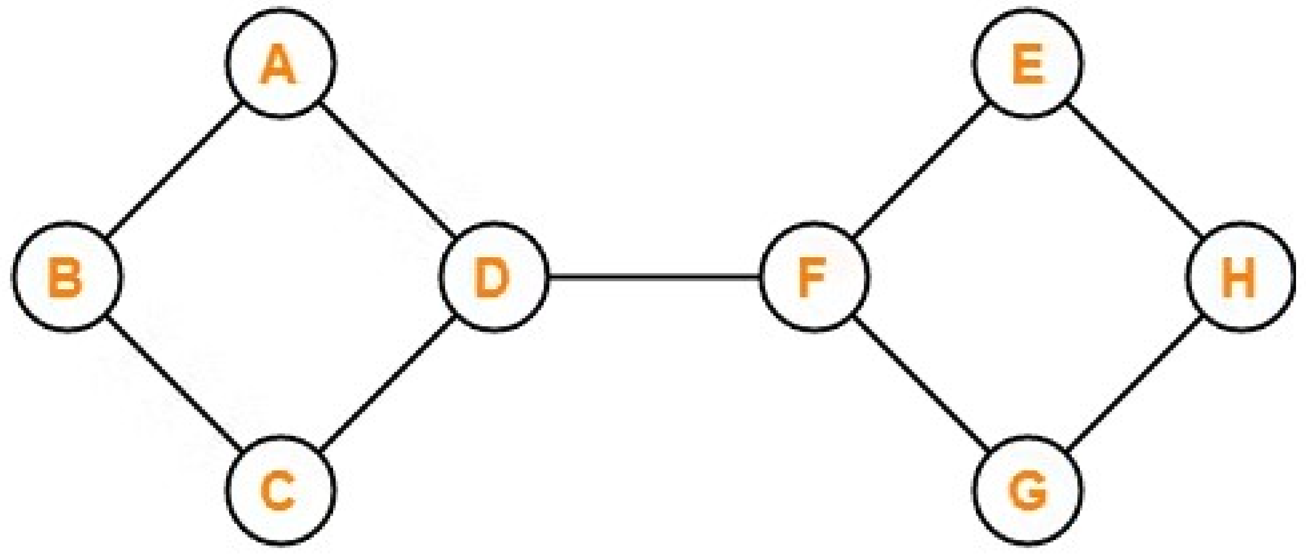
Hvor godt forsto du ukens pensum?



Bakgrunn/Recap

3

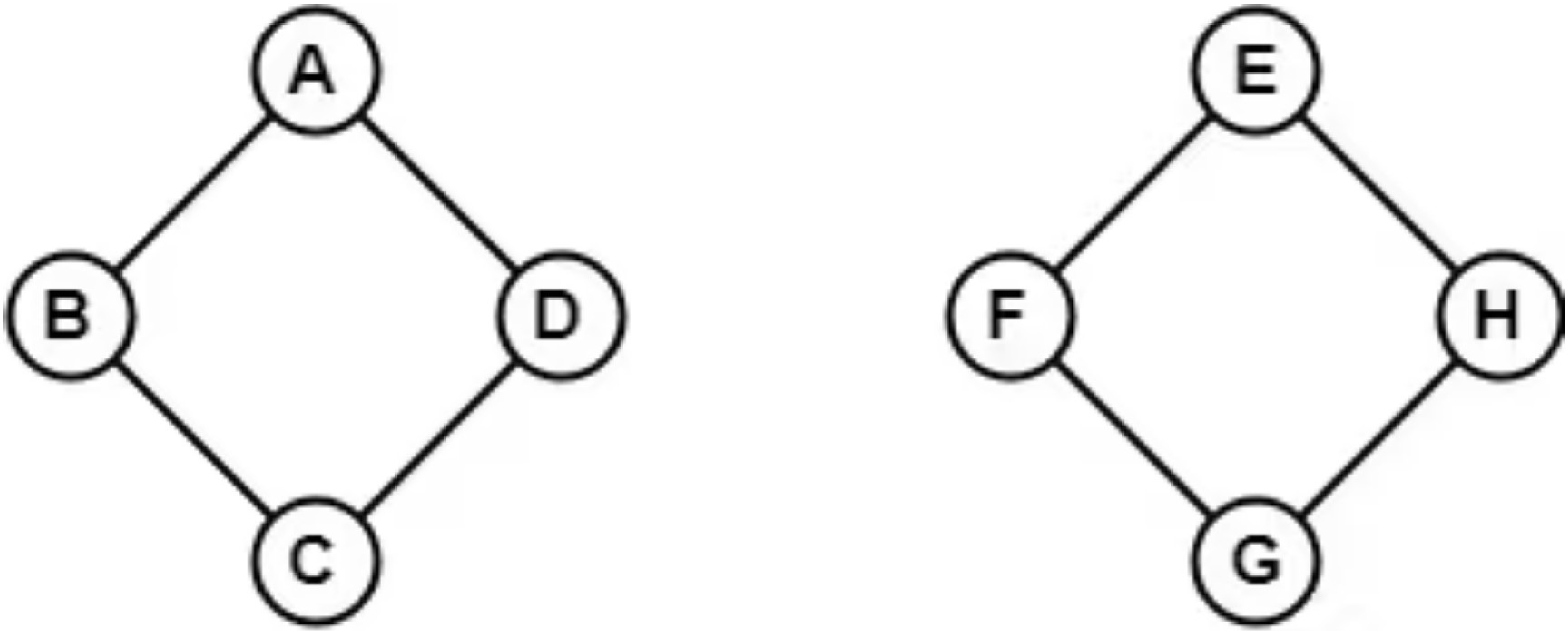
Sammenhengende grafer



Example of Connected Graph

- Rent intuitivt: En graf som henger sammen/ er graf som består av et komponent
- Definisjon: En graf er sammenhengende hvis det finnes en sti mellom hvert par av noder

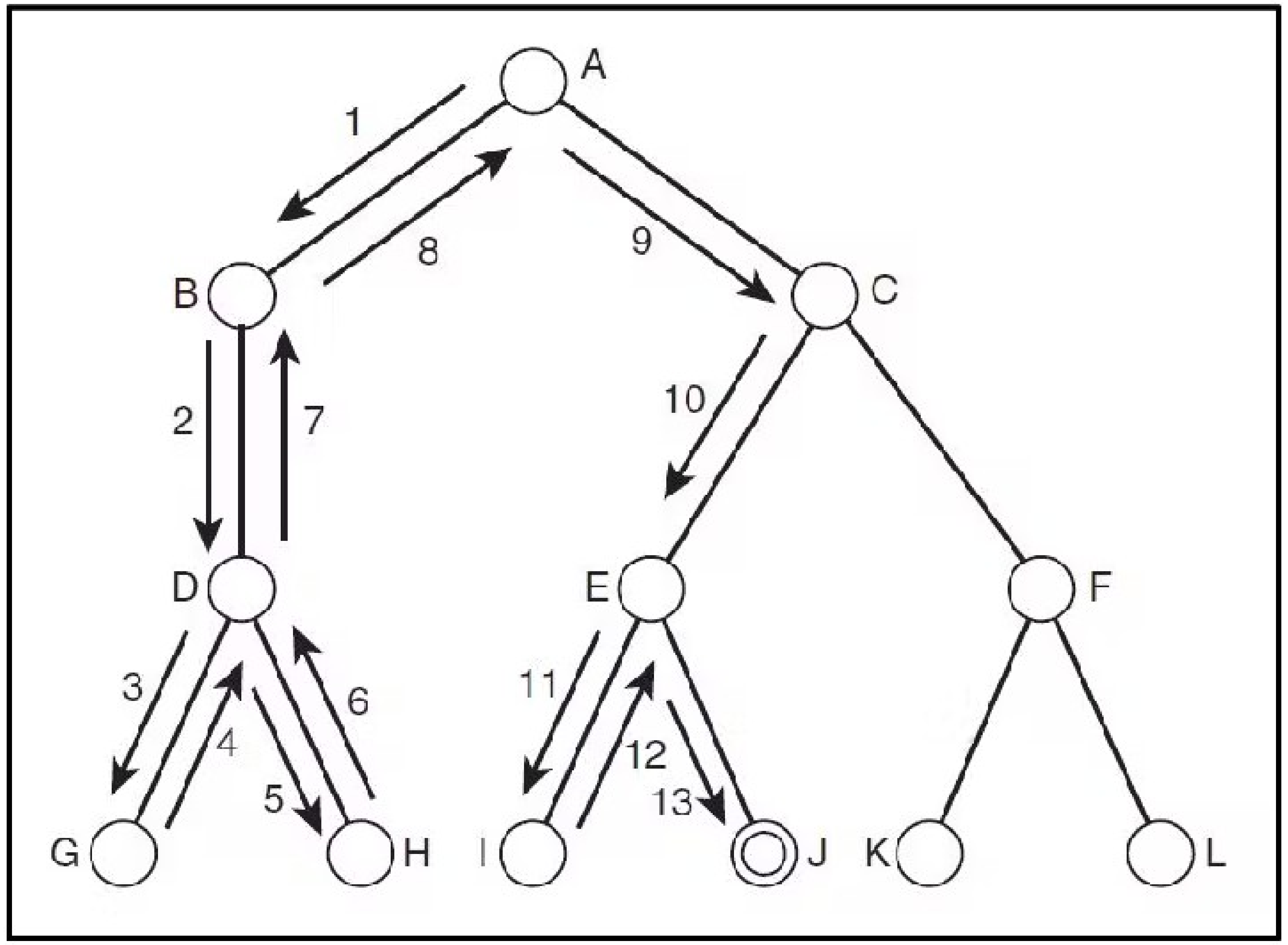
1



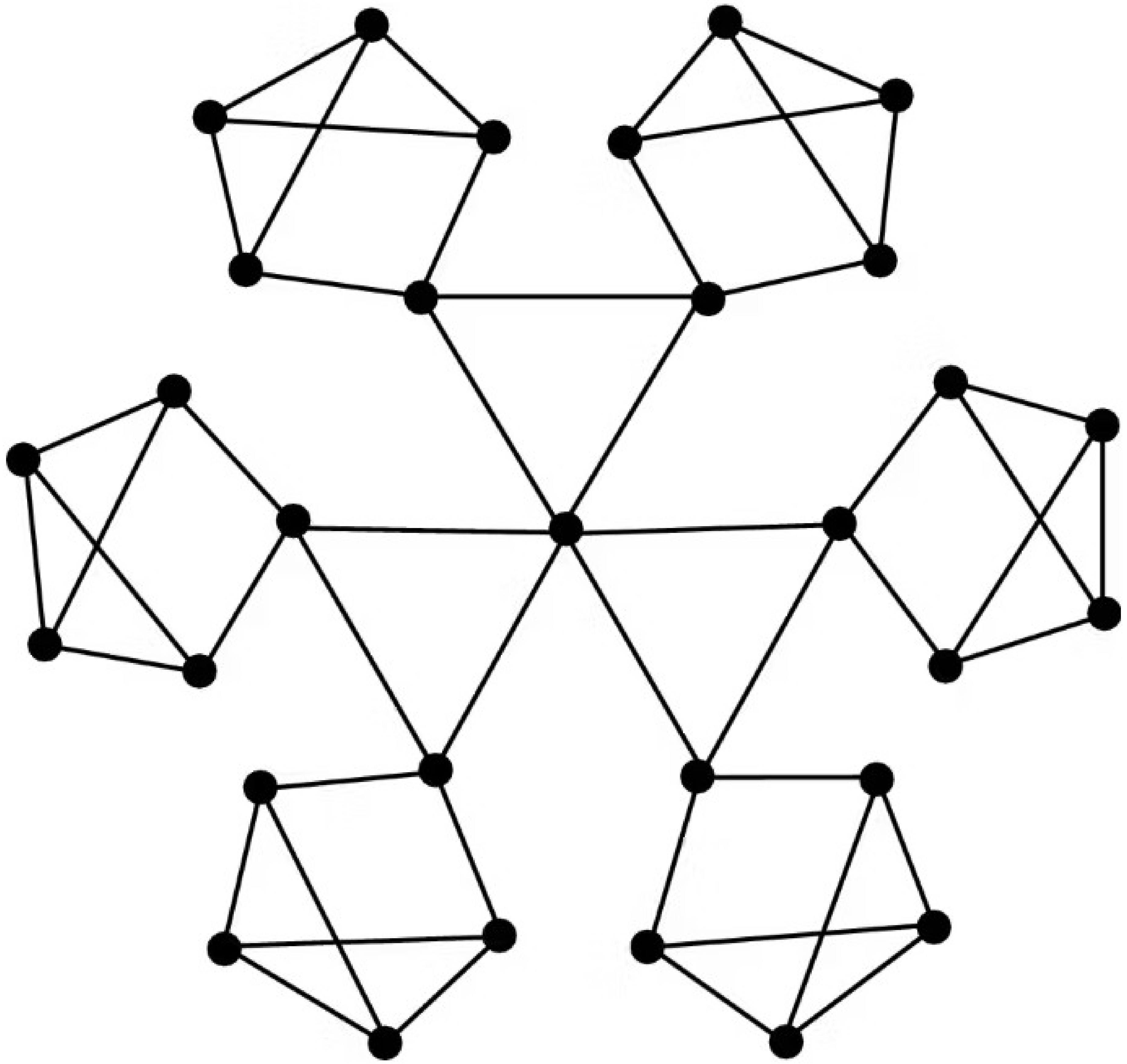
Ikke-Sammenhengende graf

Dybde Først Søk

- Ide: Gå så dypt gjennom en graf som mulig
- Går til noder som ikke er beslkt



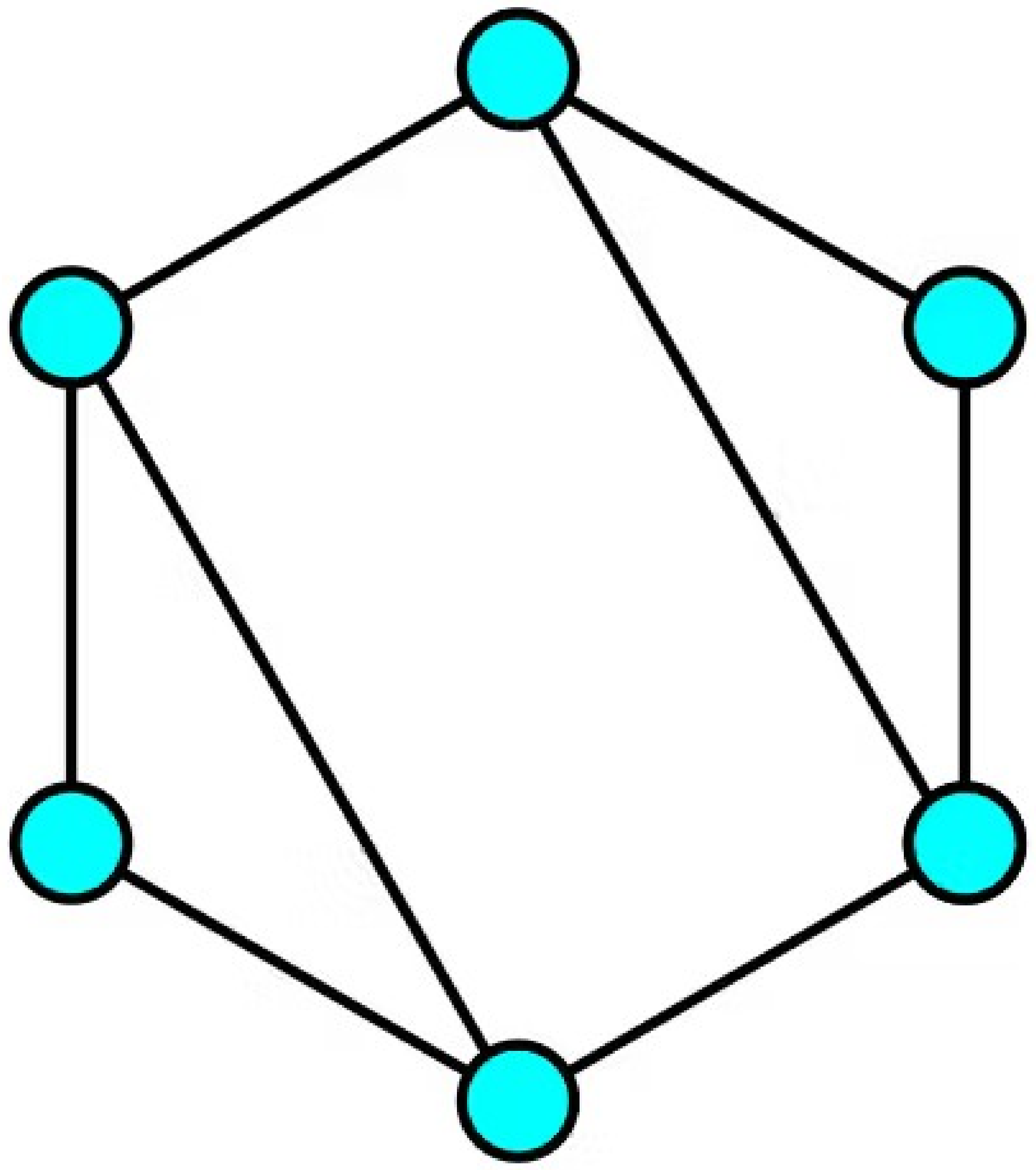
Ukens pensum



2-Sammenhengende grafer

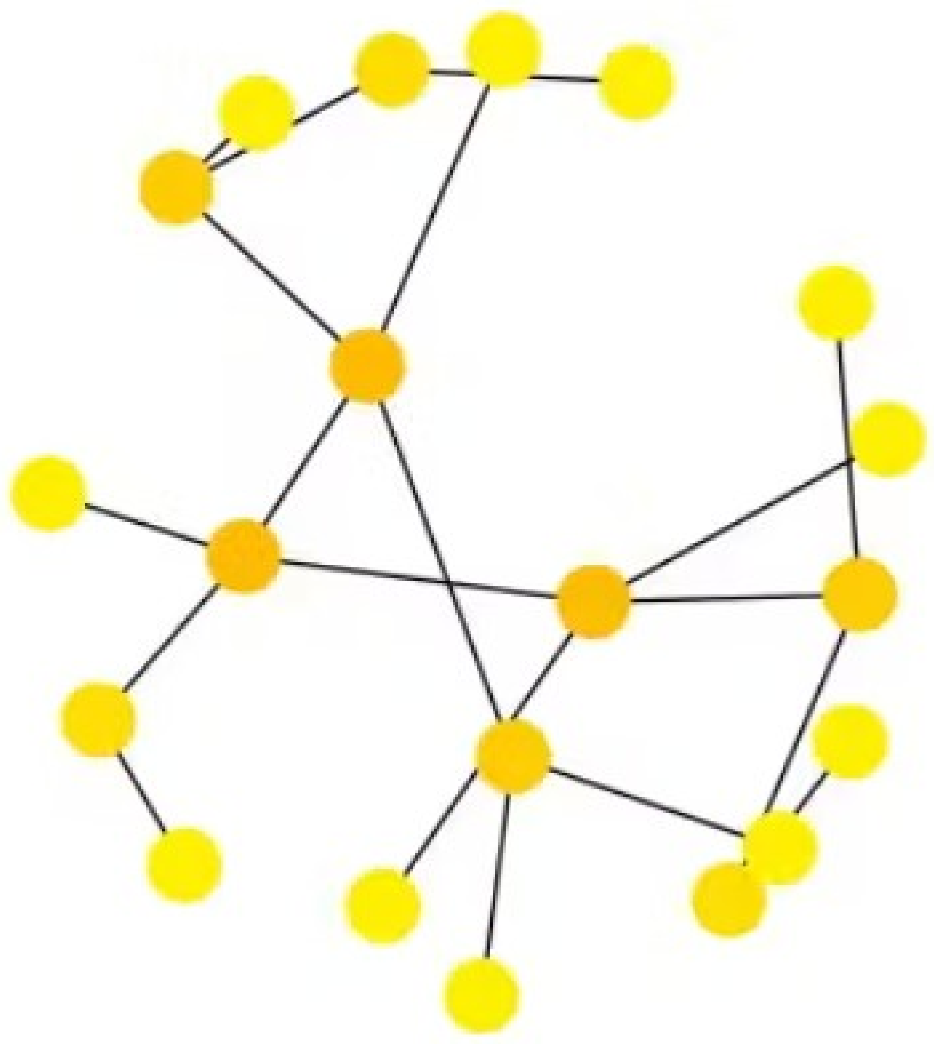
- En graf er 2-sammenhengende dersom den forblir sammenhengende etter vi har fjernet mindre enn 2 noder
- Altså at den er sammenhengende etter å ha fjernet 1 node

Mentimeter



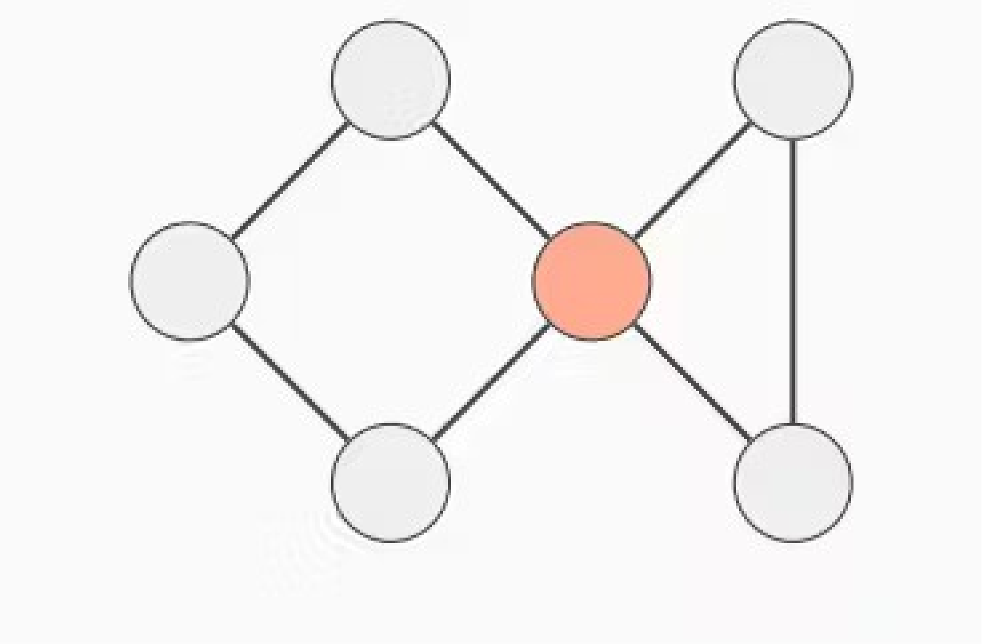
En graf kan være mer enn 2-sammennhengende

- Altså at vi må fjerne flere noder for at grafen ikke skal bli sammenhengende
- Dette kalles k-sammenhengende
- Da vil vi heller se etter seperasjonsnodene, istedenfor å gå gjennom alle noder i grafen



Seperasjonsnoder

- Dette er nodene som gjør at grafen er sammenhengene
- En 2-sammenhengende graf kan ha flere seperasjonsnoder
- Bullet 3

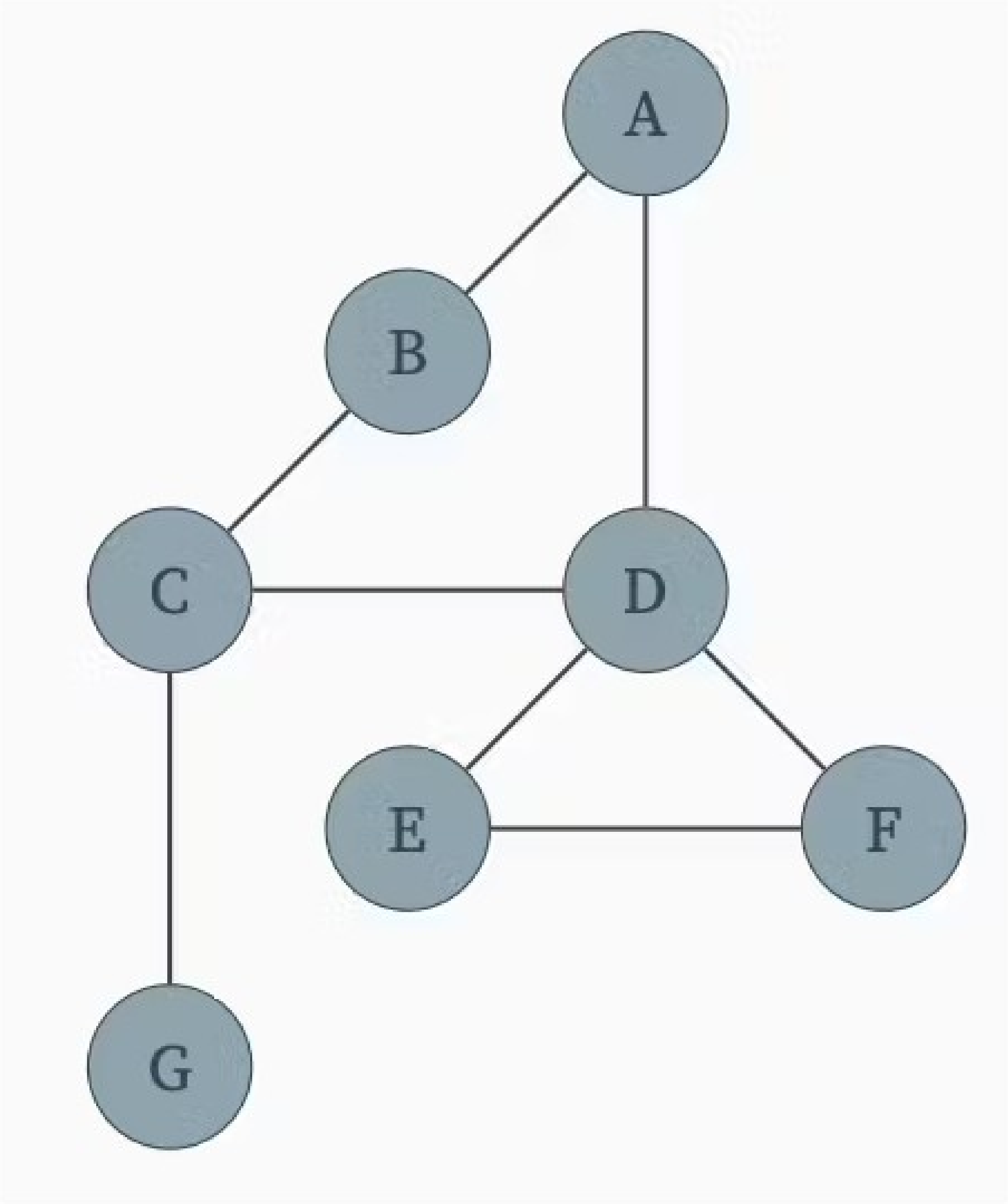


Å finne seperasjonsnoder

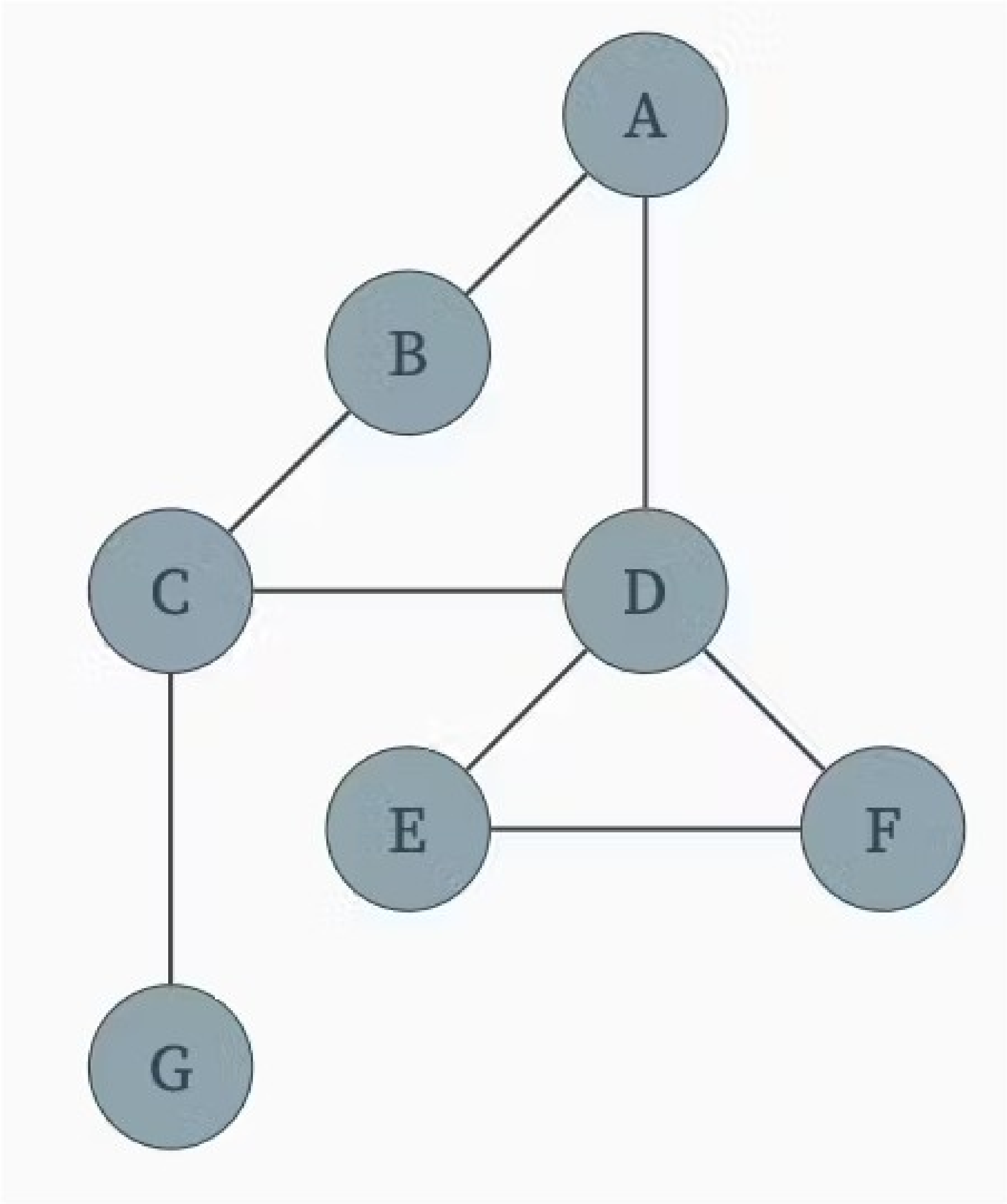
- Handler om å finne nodene som holder grafen sammenhengende
- 3 Ulike metoder(Naiv vs rask)
- Handler om å finne alle separasjonsnoder

Naiv metode

- Går gjennom alle nodene i grafen
- For hver node v så fjerner vi den noden og sine kanter
- Deretter gjør vi et Dybde først søk på den resterende grafen G'
- Hvis DFS ikke besøker alle nodene i G', så er v en separasjonsnode

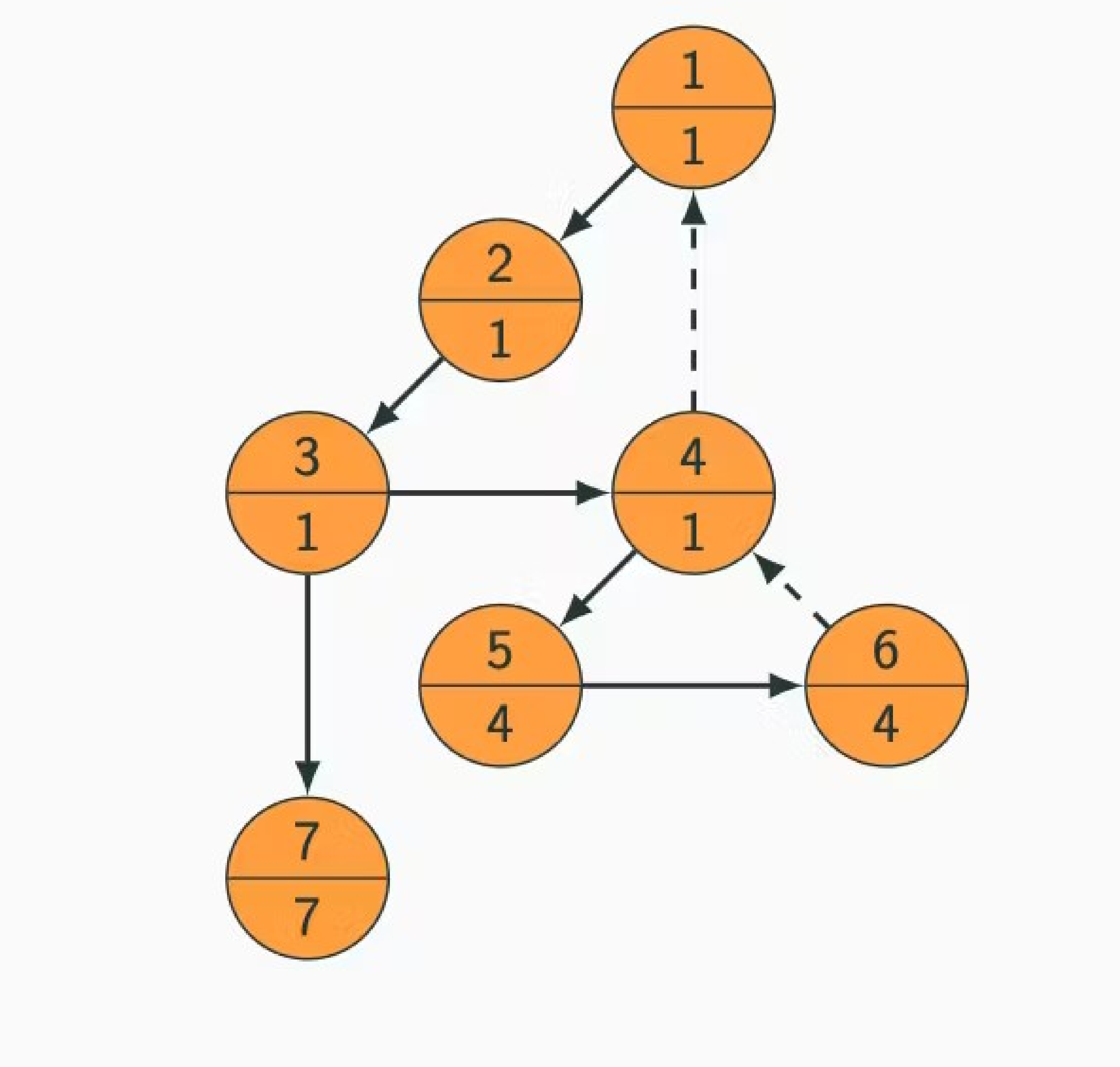


2



Rask metode(DFS)

- Gjør DFS(for å lage et spenntre)
- Vi indekserer nodene i rekkefølgen vi oppdager de
- Når vi oppdager noder lager vi en discovery edge(rettet)
- Om en node har 2 eller flere barn, så er det en seperasjonsnode



Rask Metode(Depth Low)

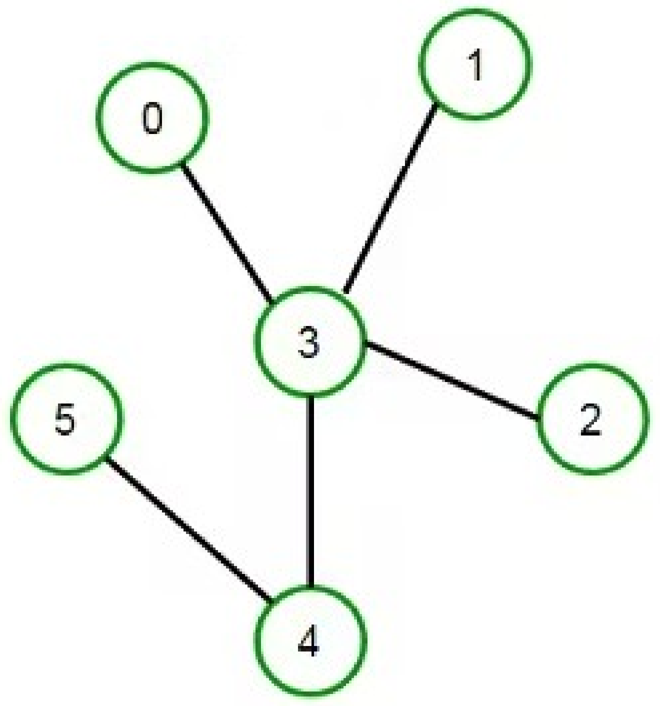
- Depht: Rekkefølgen noden ble oppdaget
- Low: Laveste depth en node kan nå ved å følge treet
- Man kan følge veien i treet, og ta med maksimalt ÉN back-edge
- En node v er en separasjonsnode dersom det finnes en node u(med kant til v) som har en low som er mindre eller lik v

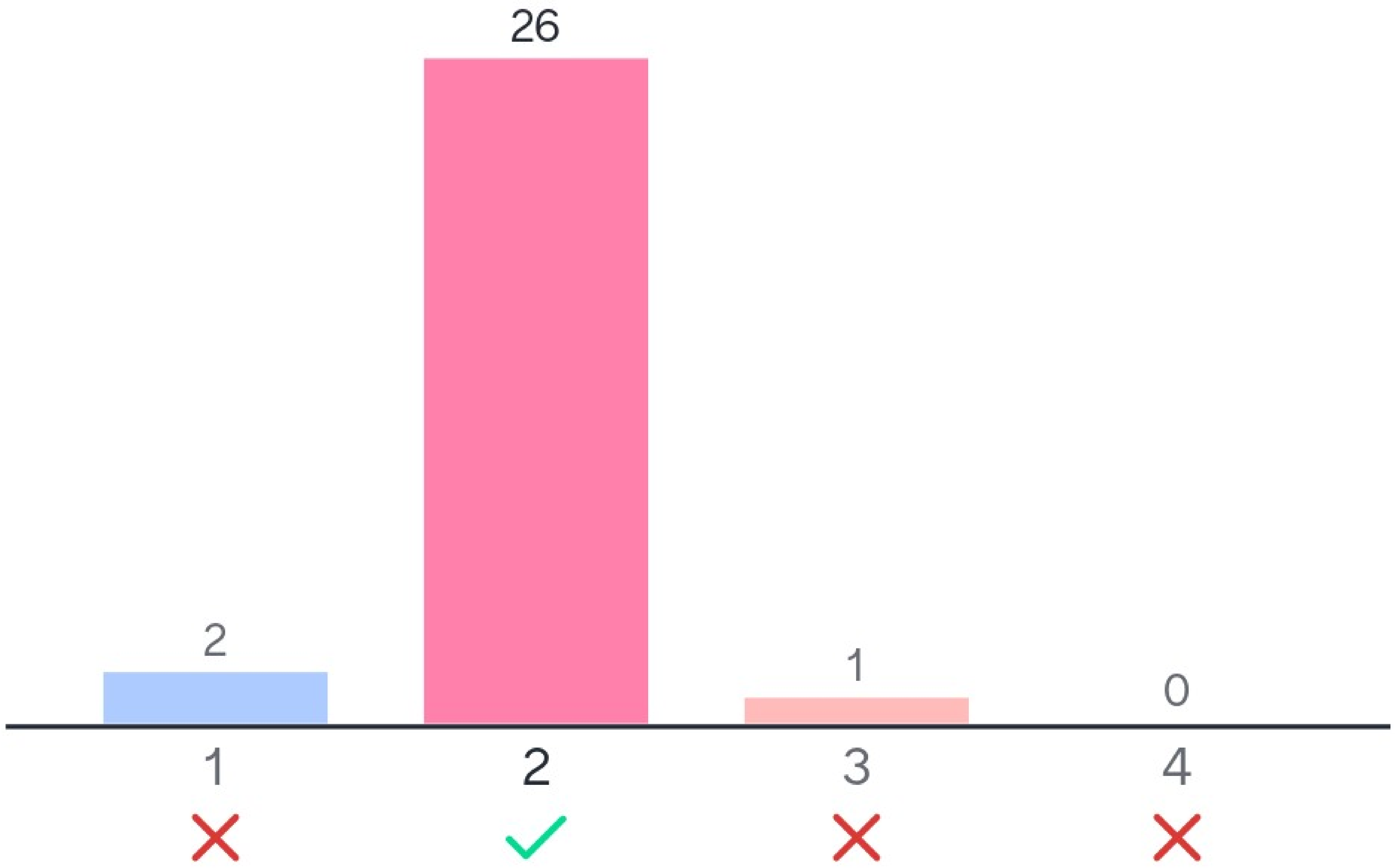
Quiz

10

Mentimeter

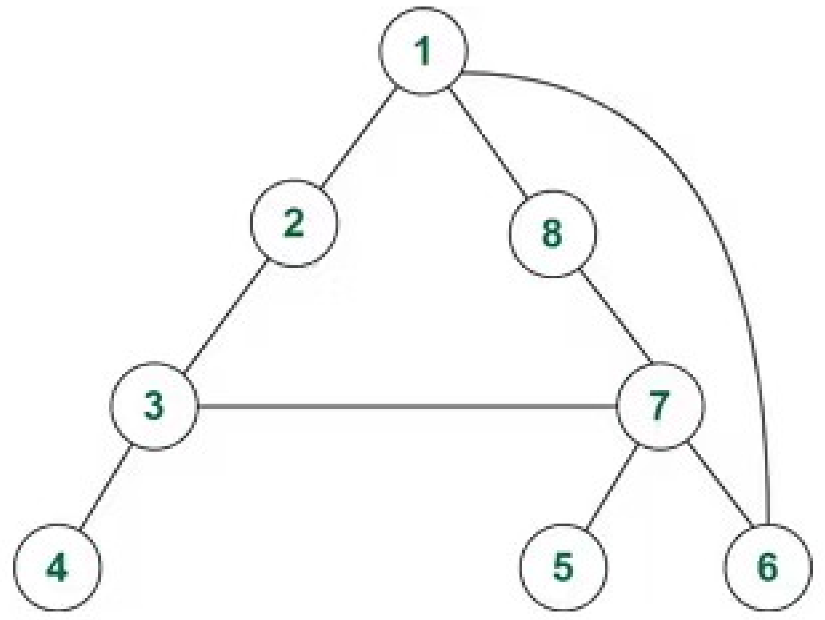
Hvor mange separasjonsnoder har denne grafen?

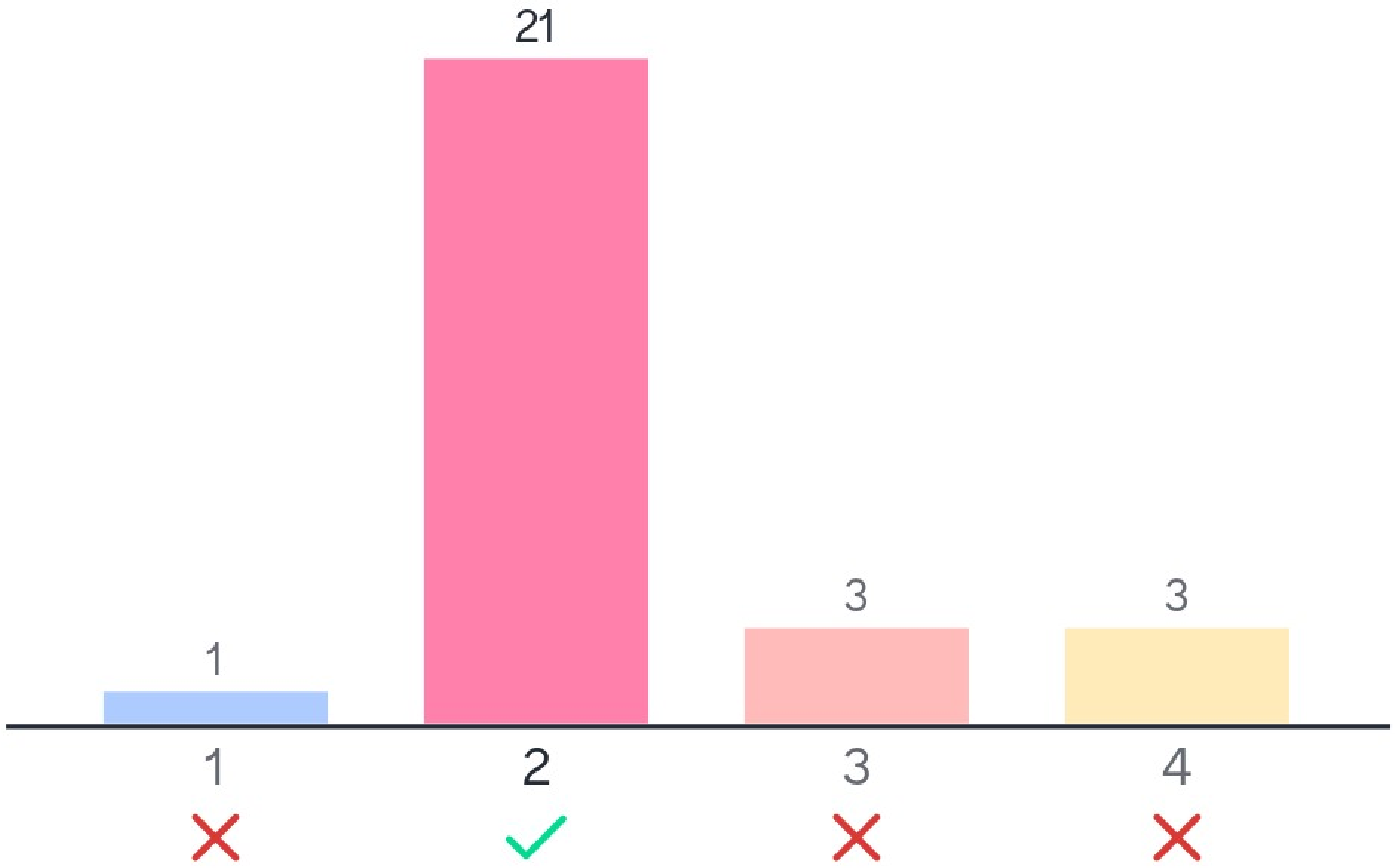




5
29

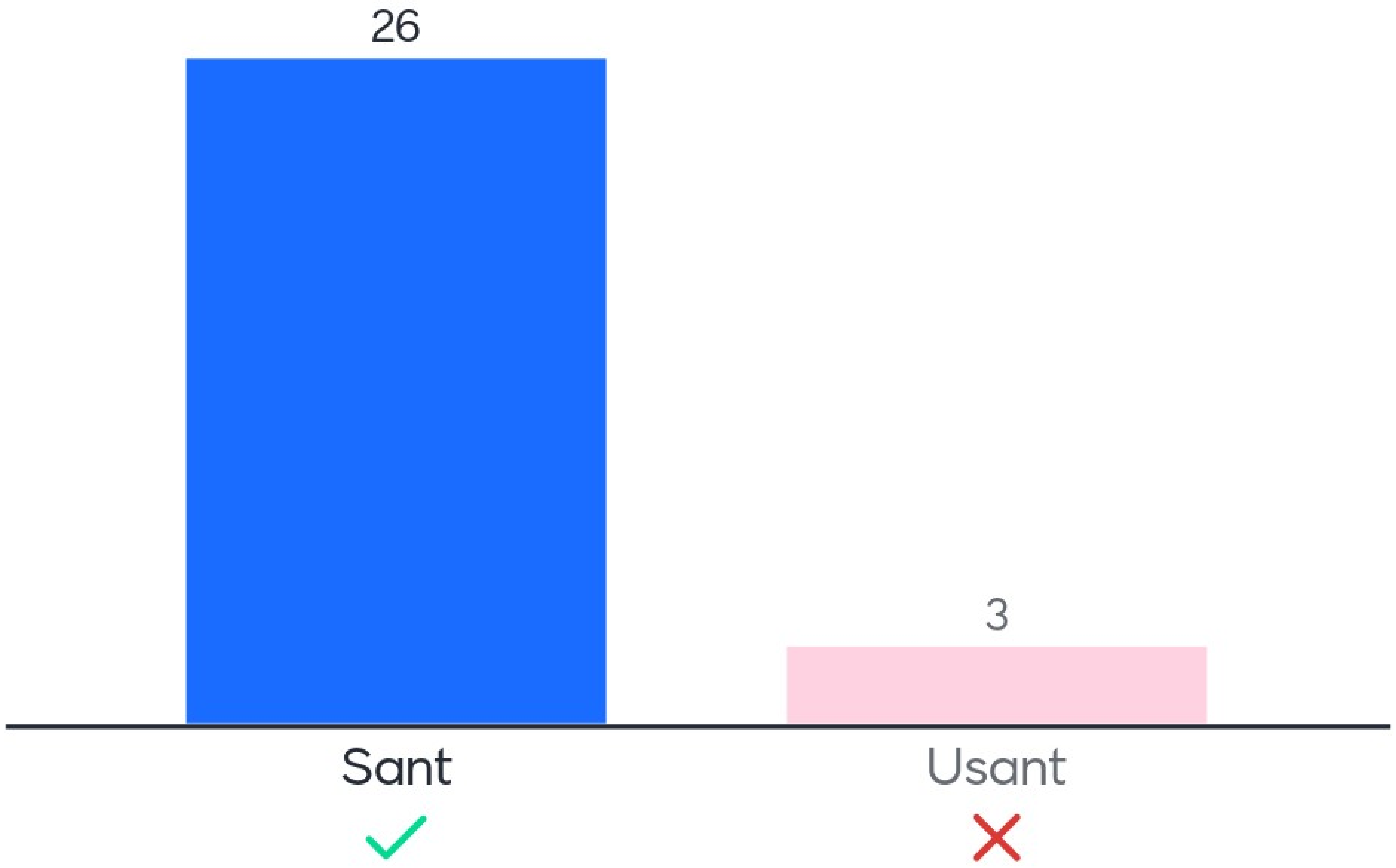
Hvor mange separasjonsnoder har denne grafen?

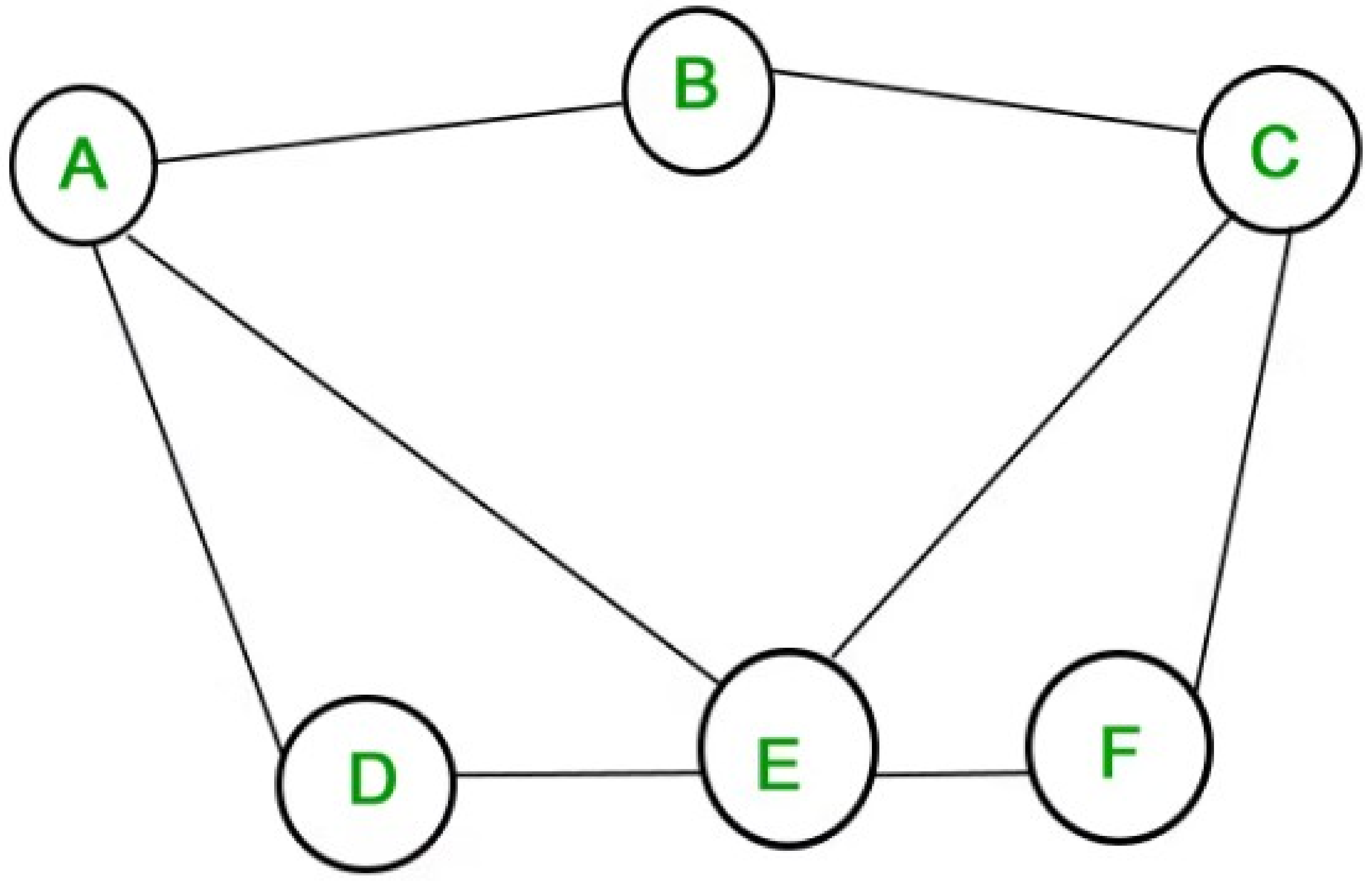




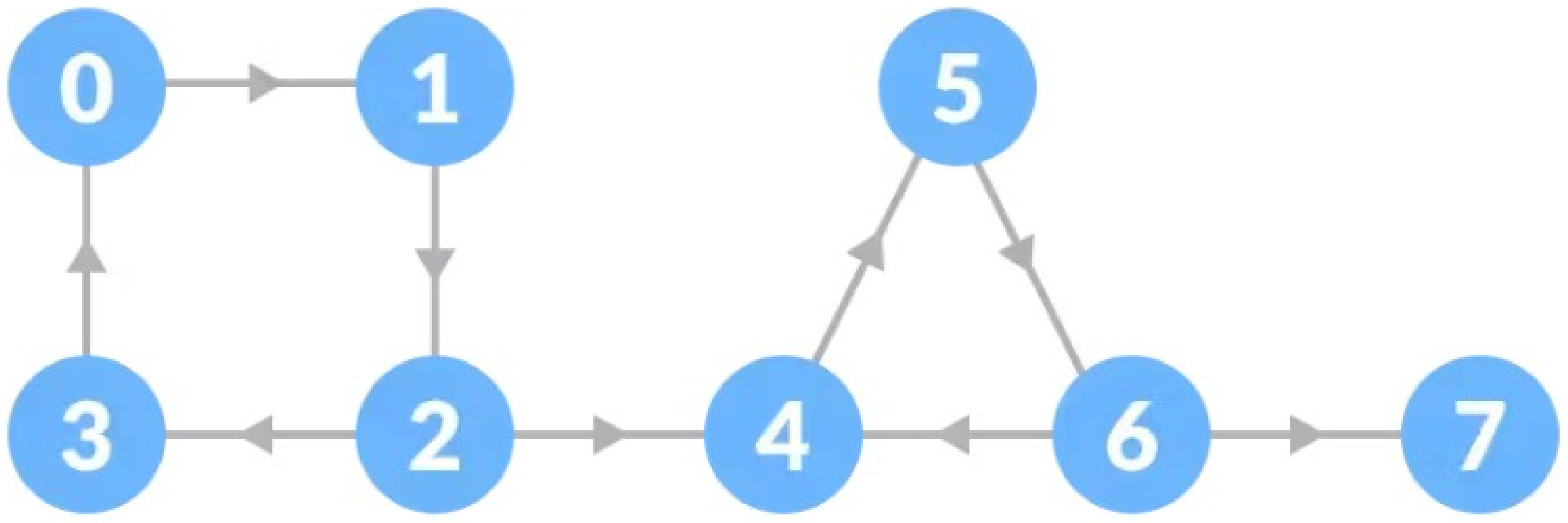
2
28

Denne grafen er 2 sammenhengende



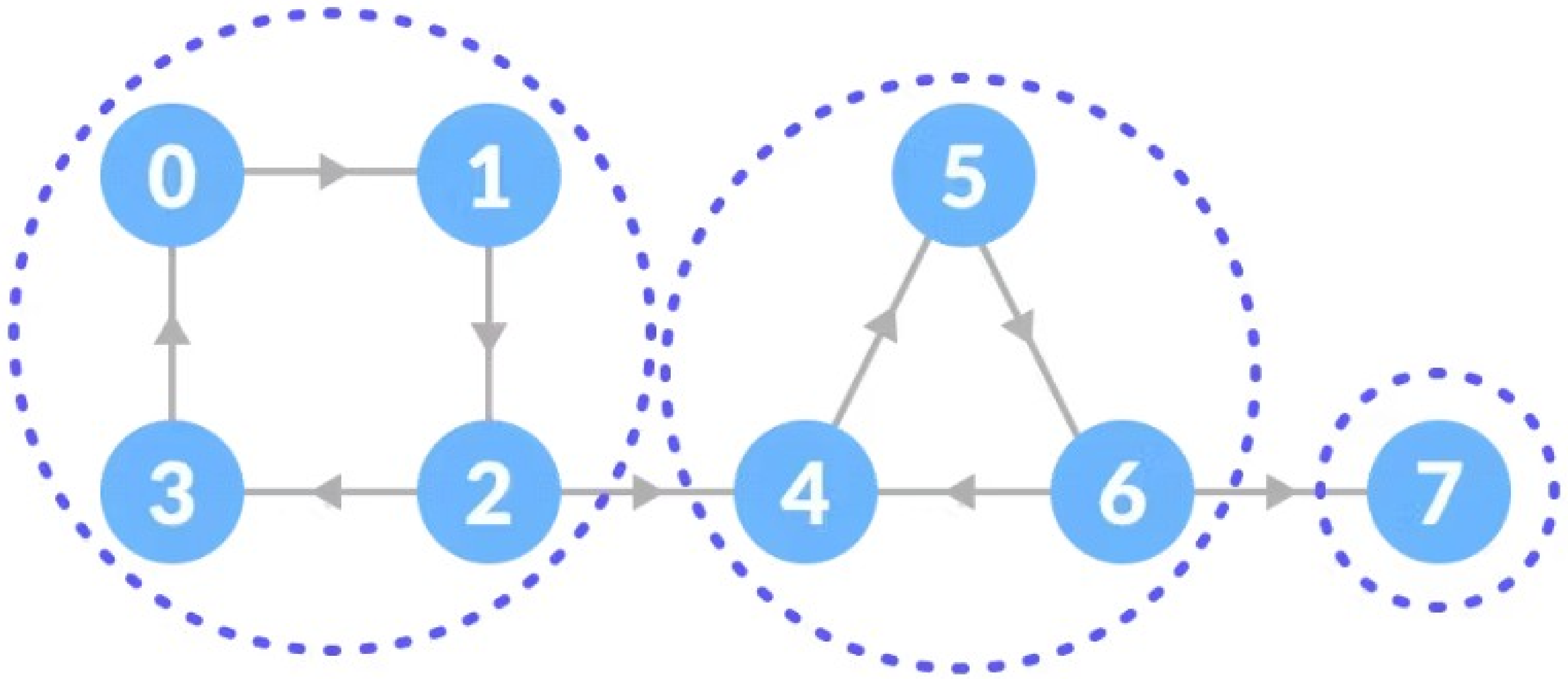


Sterk sammenhengende komponenter



Sterk sammenhengende grafer

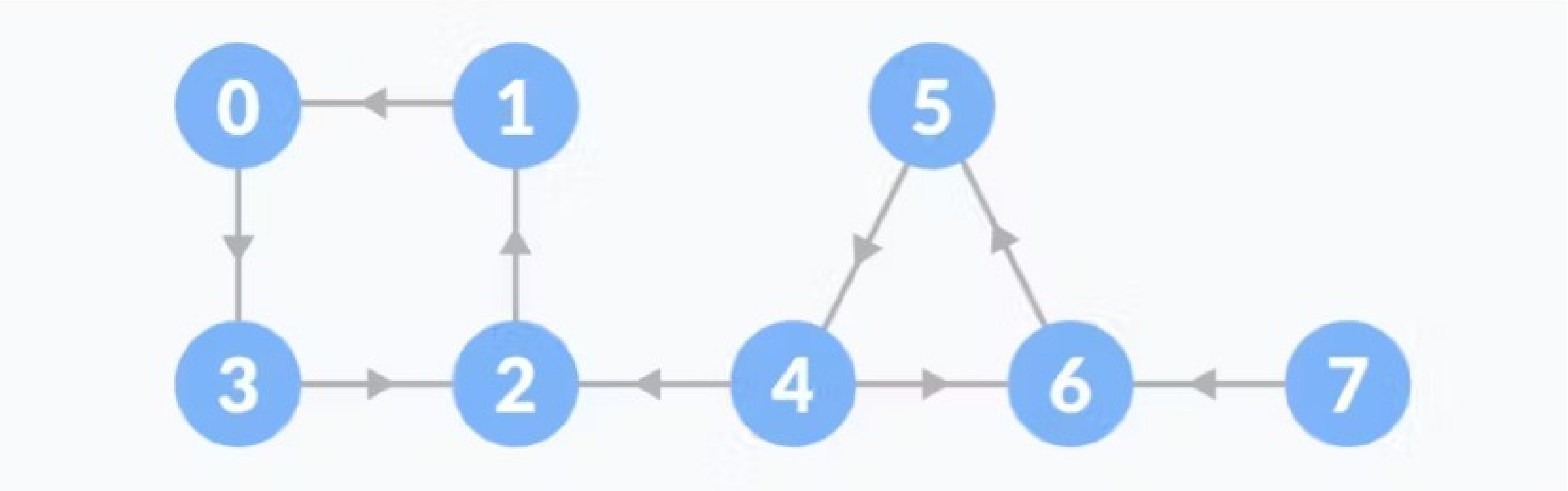
- Gjelder for rettede grafer
- Definisjon: En rettet graf er sterkt sammenhengende dersom det finnes en sti mellom alle par av noder

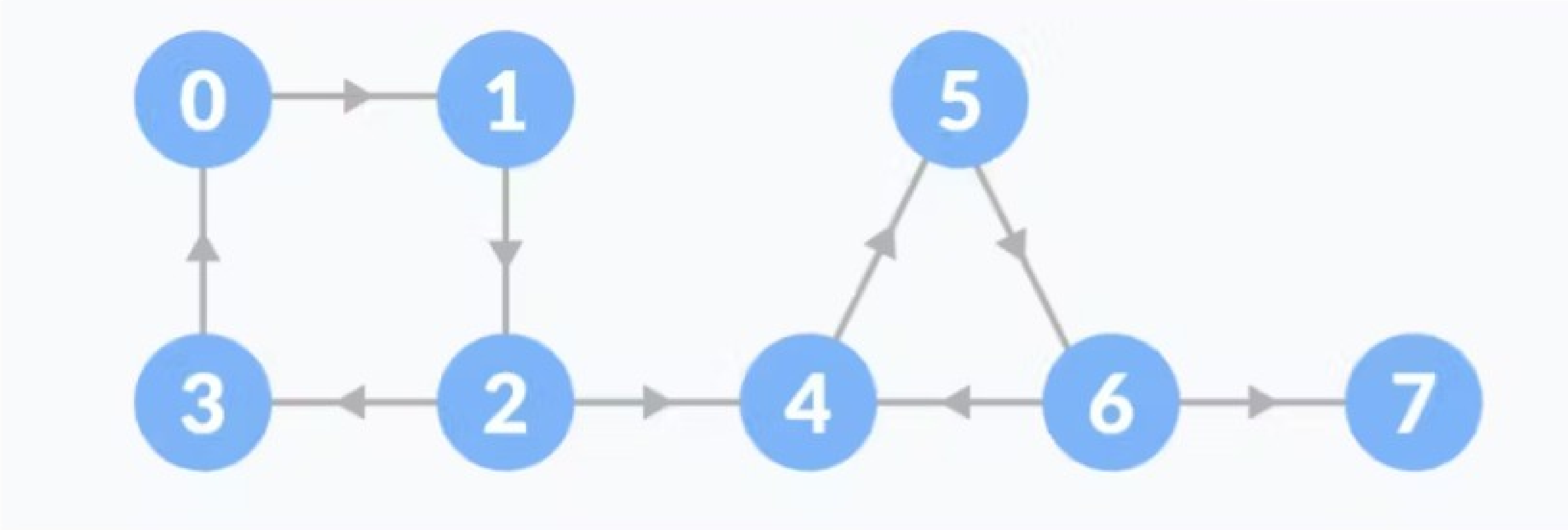


Sterk sammenhengende komponenter

- Definisjon: Dersom det finnes en sti mellom alle par av noder i et komponent, så er den sterk sammenhengende
- Basically kan man tenke at et sterkt sammenhengende komponent er en sykel

Mentimeter





Reverserte grafen

→ Den reverserte grafen er definert slik at alle kantene i en graf bytter retning

Å finne sterk sammenhengende komponenter

Rent intuitive så kjører du DFS på den original grafen G
Deretter kjører du DFS på den reverserte grafen.

Dere finner gode illustrasjoner her: https://www.programiz.com/dsa/strongly-connected-components

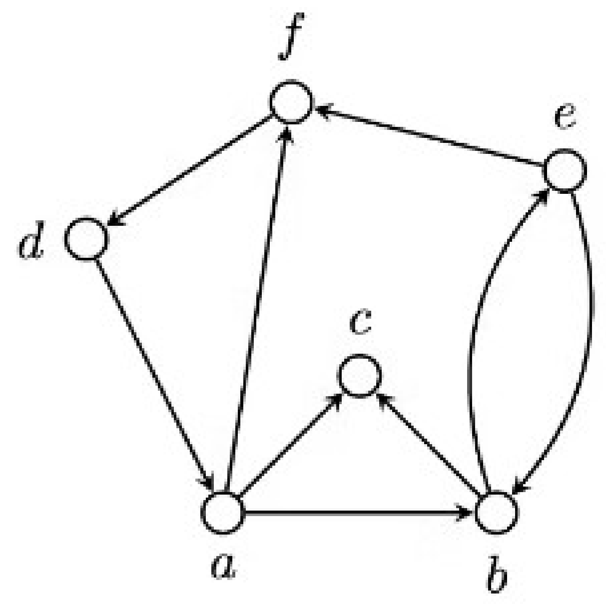
Kosaraju

- Gjør et dybde først søk
- Når en node er visited, så legges det til en stack
- På slutten av DFS søket vil vi ha en stack i rekkefølgen nodene ble visited
- Deretter reverserer vi grafen, og gjør et nytt DFS søk

Oppgaver

1b SCCs i grafer (vekt 4%)

Anta gitt den rettete grafen G med noder $a, \ldots, f$ i Figur 1.



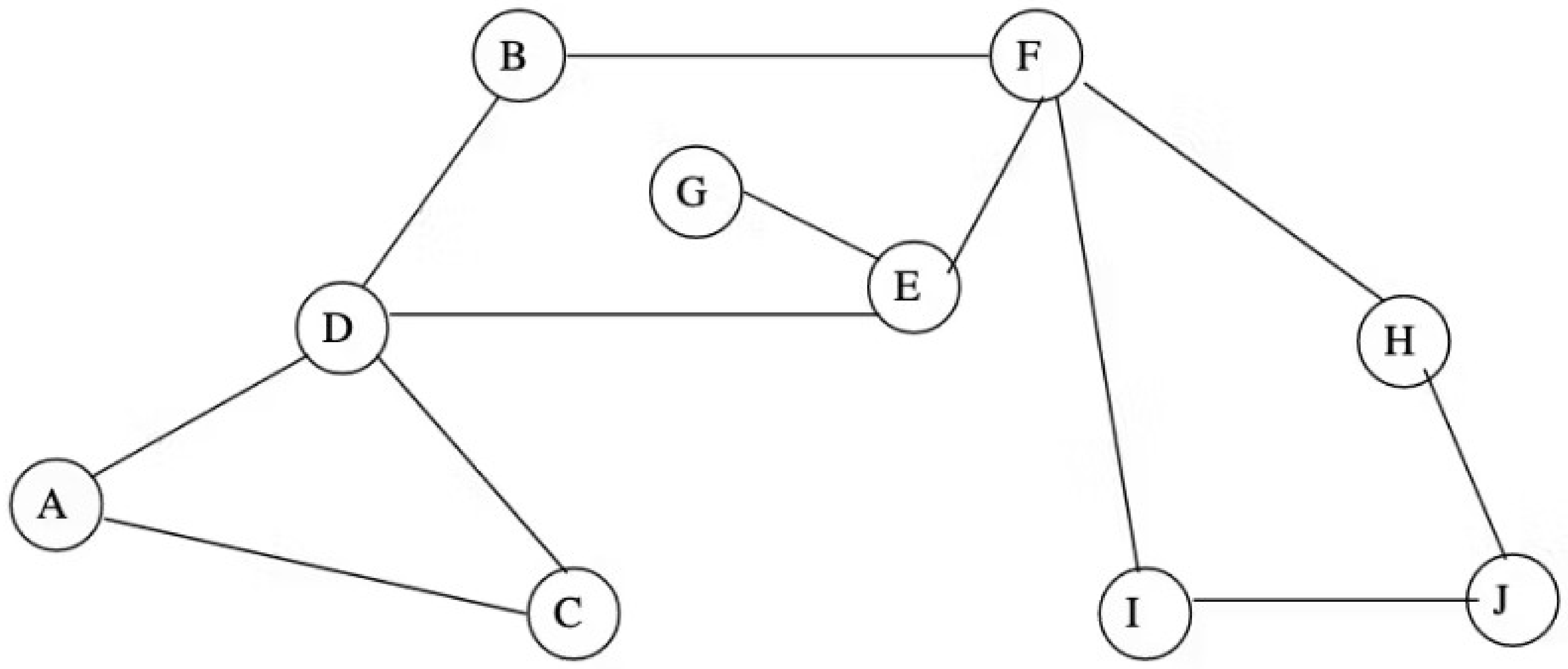
Figur 1: Rettet graf

1. Hvilke sterkt sammenhengende komponenter (SCCer, strongly-connected components på engelsk) har vi i G? Bare lag en liste.
2. Vis hvordan SCCs er bestemt algoritmisk. Gi trinnene i algoritmen. Ett trinn skal tilsvare å følge en kant i grafen mellom traversering, ikke mer detaljert enn det.
3. Anta nå en urettet graf. Beskriv (ingen kode er nødvendig) hvordan man kan bestemme SCCs av urettede grafer på en måte som er enklere enn måten for rettet grafer?[1] Innebærer denne forenklingen også en forbedring med henblikk på worst-case tidskompleksitet? Forklar kort.

Eksamen 2012

3a Biconnectivity (vekt 4%)

Finn alle *articulation points* for grafen under. Vis et dybde først-spenntre som starter fra node A samt *Num*- og *Low*- numrene for hver node.

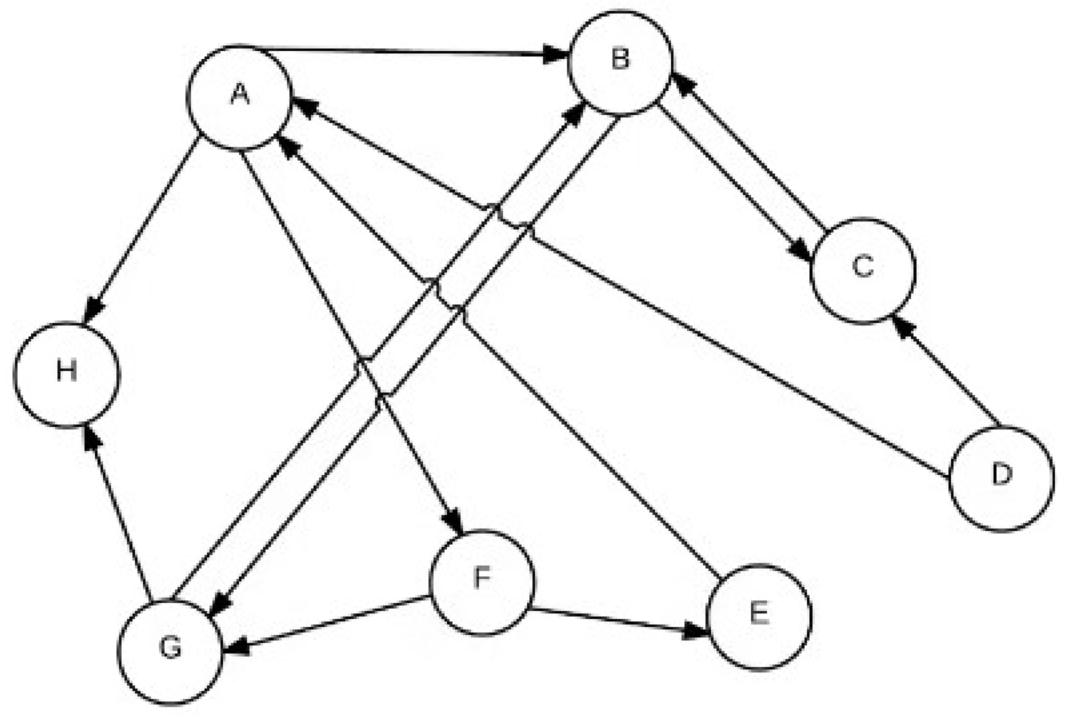


Eksamen 2013

Mentimeter

3a Sterkt sammenhengede komponenter (SCCs) (vekt 7%)

Gitt en rettet graf med noder $A, \ldots, H$ som vist i Figur 2.



Figur 2: En rettet graf

1. Hvilke sterkt sammenhengende komponenter (Strongly Connected Components (SCCs)) har grafen i Figur 2? Du skal illustrere hvordan SCCs er funnet algoritmisk ved å vise trinnene i algoritmen.
2. Gitt en vilkårlig rettet graf G, la G' være en rettet graf der hver node i G' representerer en SCC av G. For nodene u og v i G' finnes det en kant (u, v) hvis det finnes en kant i G som forbinder SCCene som tilsvarer u og v. Kan G' sorteres topologisk? Begrunn kort.

Eksamen 2014