Adeel Hussain & Rakin Ali
ADK21HT1, CINTE3

Fråga 1:

Edmonds-Karp algoritm använder BreddenFörstSökning för att hitta stigen med minsta antalet kanter med hjälp av Ford-Fulkersons metoden. Däremot hur grafen implementeras kan påverka tidskomplexitet och minneskomplexitet.

Om grafen implementeras som en grannlista blir minneskomplexiteten O(**m** + **n**) där **m** = kant och **n** = hörn. Detta eftersom man har en array med storleken **n** och där varje arrayplats är **m** lång i en länkad lista. Worst case blir när alla hörn är kopplade med varandra då minneskomplexiteten blir $O(\mathbf{n}^2)$. Tidskomplexiteten blir O(**n + m**) ifall man kör BFS på en grannlista. Detta då hörnet måste hittas vilket tar O(**n**) tid sedan alla dess kanter tas fram från listan vilket tar O(**m**). Eftersom innehållet i alla listor inte kommer itereras igenom tar det O(**n + m**) tid. Ifall grafen är gles blir m = O(**n**) vilket ger att tidskomplexiteten O(**n + n**) = O(**2n**) = O(**n**).

Ifall grafen implementeras som en grannmatris blir minneskomplexitet som krävs $O(\mathbf{n^2})$. För att kontrollera om en kant är befintligt tar det O(**1**) då man slå upp det i matrisen. Däremot för att hitta alla kanter från en hörn tar det $O(\mathbf{n^2})$ tid, då matrisen, som är av $\mathbf{n}^2$ storlek.

Sammanfattningsvis väljs grannlista vid implementation av en gles graf.

Fråga 2:

Vid konstruktion av en flödesgraf från en bipartit graf kan det leda till att stigarnas riktning ändras på grund av flödet. Detta kan leda till att vägarna från s till t inte alltid blir 3 då en riktning från ett hörn kan leda "tillbaka" och bli en omväg. Detta kan ge att BFS inte alltid returnerar en längd på 3. Worst case den returnerar stigen som passerar alla hörn på den sida som har minst hörn. Det blir 1 + 2 * min(s,t) kanter

Fråga 3:

Ford-Fulkerson kommer alltid generera heltals-flöde om kantkapacitet är heltal. Detta kan man se genom en sats och en formel.

Sats

Om $c(u,v) \in \mathbb{N}$ (heltalskapaciteter) så producerar algoritmen ett maximalt flöde med heltalsflöden i varje kant

$$c_f(u,v) = c(u,v) - f(u,v), \text{ där } f(u,v) = -f(v,u).$$

Genom formeln ser vi att om flödet i en kant är heltal så måste c(u,v) och f(u,v) vara två heltal. **Alltså måste kant kapacitet c(u,v) vara heltal.**

Flödet längs en kant kan sättas till antingen 0 eller 1, vilket endast representerar ett binärt värde för sant/falskt. Detta menas med att 1:an kan ändras och sättas till vilket heltalsvärde som helst så länge alla kanter har samma heltalsvärde. Alltså kan man sätta alla kanter till 0/n där n kan vara vilket heltalsvärde som helst. **Alltså borde lösningen för flödesalgoritmen fungera även om man ändrar reduktionen till 2 istället för 1.**