Øving 7 Algoritmer og datastrukturer IDATT2101

# *Vektede grafer: Edmond-Karp algoritme*

# Implementasjon

For å implementere Edmond-Karp algoritmen i denne øvingen tok jeg bruk av bredde-først søk algoritmen fra tidligere øvinger. Denne metoden ble modifisert med hensyn på Edmond-Karps algoritme. I algoritmen må det på et gitt tidspunkt ikke være mulig å finne sluttnoden.

Et bilde som inneholder tekst, skjermbilde, Font, nummer

Automatisk generert beskrivelse

Figur 1: Bredde-først søk algoritmen fra tidligere øvinger. Merk at den eneste forandringen er at denne metoden returnerer en boolsk verdi avhengig om sluttnoden er funnet i søket.

Nå som bredde-først søk algoritmen er implementert, må vi få tak i den korteste veien fra startnoden til sluttnoden. Denne informasjonen blir viktig ettersom vi er nødt til å endre på kantene før vi i neste omgang utfører ett nytt bredde-først søk. For å få til dette ble metoden under implementert:

Et bilde som inneholder tekst, Font, line, skjermbilde

Automatisk generert beskrivelse

Figur 2: Utsnitt av findNodePath metoden. Denne metoden bruker sluttnodens forgjengere til å finne veien fra startnoden til sluttnoden.

Det neste steget blir dermed å finne hva flyt økningen blir gjennom denne veien. For å få til dette ble følgende metode implementert (vist på neste side):

Et bilde som inneholder tekst, Font, line, skjermbilde

Automatisk generert beskrivelse

Figur 3: Utsnitt av findFlowIncrease metoden. Denne metoden bruker resultatet fra findNodePath-metoden til å finne den laveste flyt verdien gjennom veien fra startnoden til sluttnoden. Dermed finner vi ut hva flyt økningen blir.

Dermed har vi det vi trenger for å oppdatere grafen slik at vi kan utføre et nytt bredde-først søk. Metodene under viser hvordan denne oppdateringen blir gjort:

Et bilde som inneholder tekst, skjermbilde, Font, dokument

Automatisk generert beskrivelse

Figur 4: Utsnitt av metodene updateEdge og updateGraoh. Disse metodene er ansvarlige for å oppdatere/fjerne kanter slik at den nye grafen er klar for et nytt bredde-først søk.

Merk at denne løsningen for å håndtere kanter ikke tar bruk av tipsene beskrevet i oppgavebeskrivelsen. Her anbefales det å bruke to felt i kantstrukturen: en for flyt og en for kapasitet. Jeg har derimot valgt å bruke å bare bruke ett felt for dette: flowValue. Grunnen til dette er fordi jeg syns denne løsningen virket mer intuitiv. På denne måten trengte jeg bare å forholde meg til en variabel.

I tillegg anbefales det å opprette motstående kanter etter innlesning fra fil. Denne tilnærmingen har jeg heller ikke tatt i bruk i denne løsningen. For meg virker det mer praktisk å fjerne, redigere og legge til kanter det trengs, slik at ikke grafen holder på flere kanter enn det den trenger. Utgangspunktet mitt for implementeringen av disse metodene var å oppnå nøyaktig samme restgraf som ble vist under demoen av Edmond-Karps algoritme i forelesningen 10.10.2023.

Dermed har vi alt det som trengs for å implementere Edmond-Karps algoritme. Det ble gjort på følgende måte:

Et bilde som inneholder tekst, skjermbilde, Font, nummer

Automatisk generert beskrivelse

Figur 5: Utsnitt av Edmond-Karps algortmen.

I denne metoden som representerer Edmond-Karp algoritmen, returneres det en streng som representerer resultatet fra denne algoritmen. Her opprettes int-variabelen maxFlow for å finne den maksimale flyten til en graf. Videre kjøres et bredde-først søk i en while-loop så lenge søket finner sluttnoden vi er interessert i. Det er derfor endringen i bredde-først søk algoritmen ble gjort i utgangspunktet. I while-loopen finner vi veien fra startnoden til sluttnoden, og flyt økningen gjennom denne veien. Disse verdiene bruker vi igjen for å oppdatere grafen slik at den er klar til et nytt bredde-først søk. På et tidspunkt kommer ikke bredde-først søket til å finne sluttnoden, og metoden returnerer resultatet.

# Resultat

Fra oppgavebeskrivelsen for denne øvingen fikk vi oppgitt denne tabellen for resultatene av denne algoritmen:

Et bilde som inneholder tekst, skjermbilde, Font, nummer

Automatisk generert beskrivelse

Der maksflyten til flytgraf3 og flytgraf5 er ukjent.

Gjennom å kjøre Edmond-Karp algoritmen med disse verdiene finner vi ut at maksflyten til flytgraf3 er 42, og maksflyten til flytgraf5 er 90. Resultatet av kjøringen blir følgende (vist på neste side):

Et bilde som inneholder tekst, skjermbilde, Font, nummer

Automatisk generert beskrivelse

Figur 6: Utskrift av resultatet fra Edmond-Karp algoritmen i flytgraf3 med startnode 0 og sluttnode 1.

Et bilde som inneholder tekst, skjermbilde, Font, nummer

Automatisk generert beskrivelse

Figur 7: Utskrift av resultatet fra Edmond-Karp algoritmen i flytgraf5 med startnode 0 og sluttnode 7.

Da kan vi til slutt fullføre tabellen:

Et bilde som inneholder tekst, Font, skjermbilde, nummer

Automatisk generert beskrivelse