

IN2010 – Obligatorisk Innlevering 3

Programmet kan kjøres enkelt med kommando `§ python3 innleveringsoppgave3.py`.

Det skal ikke være feil eller mangler i oppgaven. Jeg ser dog at jeg får litt variable resultater på hvilke skuespillere oppgave 2 går igjennom for å komme til den vi søker korteste sti til. Derimot er stiene som kommer ut alltid lik i størrelse som eksempelet i oppgaveteksten, så da går jeg ut i fra at denne oppgaven har flere riktige løsninger. Det er en liten «lazy shortcut» implementert i oppgave 2 da jeg printer filmen som ligger først i listen min av filmer to skuespillere deler (linje 170 i graph.py). Her kan det jo være flere filmer som linker de to samme skuespillerne, men dette blir ikke direkte feil mtp. Oppgaveteksten i min oppfatning.

Jeg vil si oppgaven gjør det den er bedt om å gjøre, men kanskje ikke helt optimalisert med tanke på tidsbruk. Som beskrevet under kjøring av main programmet så bruker programmet ca. 2 minutter på å kjøre alle oppgavene. Satt opp mot løsningsforslaget på 3 sekunder så er dette en ganske stor deviation. Programmet mitt har flere forløkker som går igjennom hele actors listen på 119000 objekter. Jeg mener kjøretiden kunne vært forbedret om jeg hadde klart å implementere andre algoritmer slik at jeg fikk bort noen av disse løkkene. Samt hadde jeg kanskje hatt ett bedre utgangspunkt om jeg implementerte grafen med en egen nodeklasse for å holde styr på kanter osv. Jeg oppdaget dette ganske sent og valgte derfor ikke å skrive om oppbygningen av grafen samt oppdatere alle metodene. I tillegg er programmet skrevet i Python fremfor Java, som vi vet er en del tregere til grunns pga. Dynamiske typer osv.

Jeg har tatt i bruk algoritmene fra forelesningene, bredde-først søk for både en og to noder. Dijkstra for en og to noder, samt enda en bredde-først søk for å implementere den siste og 4 oppgaven. Ser i retrospekt at jeg kanskje kunne fjernet denne ved å endre litt på den tidligere bredde-først algoritmen.