

DM574: INTRODUKTION TIL PROGRAMMERING

Gruppeprojekt — Efterår 2024 — Fase III

Afleveringsfrist: d. 20. december 2024, kl.18

Oversigt

I denne fase af projektet udvikles én eneste funktion i modul `prune.py`, hvis formål er at gøre programmet hurtigere.

Modul `prune.py`

Dette modul indeholder en funktion `prune(w: list[Filter], n: int) -> list[Filter]`, som tager en liste af filtre `w` og et antal kanaller `n`, og returnerer en ny liste som kun indeholder nogle af de oprindelige filtre fra `w`.

Der forventes som minimum to forskellige implementeringer af `prune()`. For at gøre det nemt at skifte imellem dem, skal de defineres som separate (hjælpe)funktioner, hvor den ene kaldes fra selve `prune()`. Det betyder, at modulets struktur skal ligne den nedenstående:

```
def prune(w: list[Filter], n: int) -> list[Filter]:  
    """Removes some filters on n channels from w."""  
    return boring_prune(w,n)
```

```
def boring_prune(w,n):  
    """Does something boring."""  
    (...)
```

```
def cool_prune(w,n):  
    """Does something cool."""  
    (...)
```

Den første implementering af `prune()` er en, der fjerner semantiske dubletter – dvs, netværk som har de samme outputs. Funktionen beholder så mange som mulige filtre fra `w`, uden at der er to filtre med den samme mængde af binære outputs. Bemærk, at denne funktion ikke har brug for input `n`.

Den anden implementering reducerer antallet af filtre ved at "gætte" hvilke af dem er "tættest på" at være et sorteringsnetværk. Dvs at funktionen skal anvende et evalueringskriterium (også kaldt en *heuristik*) for at give hvert netværk en værdi, og bagefter kun beholde de netværk, der har højest værdi. Kernen i opgaven er at finde ud af, hvad en god heuristik kan være.

Derudover kan der være flere implementeringer af `prune`, i tilfældet fx. at I vil eksperimentere med forskellige heuristikker og sammeligne dem – eller hvis I har andre idéer, som I vil teste.

Det er vigtigt, at rapporten beskriver jeres overvejelser ifm. definitionen af heuristikker, og inkluderer en evaluering af, hvordan de virker.

Aflevering

Hver gruppe skal aflevere en zipfil med navn `groupXX.zip` (hvor `XX` er gruppens nummer), som indeholder:

- en fil `prune.py`, som implementerer den kontrakt beskrevet tidligere;
- en fil `reportXX.pdf`, hvor XX er gruppens nummer.

Rapporten skal bl.a. beskrive, hvordan modulerne er designet – de valg, der er blevet taget ifm. definitionen af datatyper, samt alle relevante overvejelser ifm. implementering af funktioner. Det er vigtigt at inkludere eksempler og forklare, hvordan modulerne er blevet testede. Rapporten skal også inkludere kildekode for de tre moduler som bilag. Grupper skal vælge en fasekoordinator, som skal identificeres tydeligt på rapportens forside.

Vigtige pointer.

- Hvis instruktioner om formatet til aflevering ikke følges, kan projektet blive afvist.
- Rapporten er basis for evalueringen.
- I takt med kursets formål er det mening at alle funktionalitet implementeres af gruppen. Det betyder at det ikke er tilladt at importere nogen af Pythons biblioteker, undtaget `functools` og `dataclasses`.