

Innlevering #5

Lever oppgave

Forfall 1. des. av 23.59 **Poeng** 10 **Må leveres** en filopplasting
Filtyper py og ipynb

Denne innleveringen utgjør en del av grunnlaget for å kunne være kvalifisert til å få karakter i faget.

Totalt 6 av 8 innleveringer må gjøres og bestås før du som student er kvalifisert til å få karakter i faget.

Innleveringen inngår ikke som en del av slutt karakter i faget.

Lykke til!

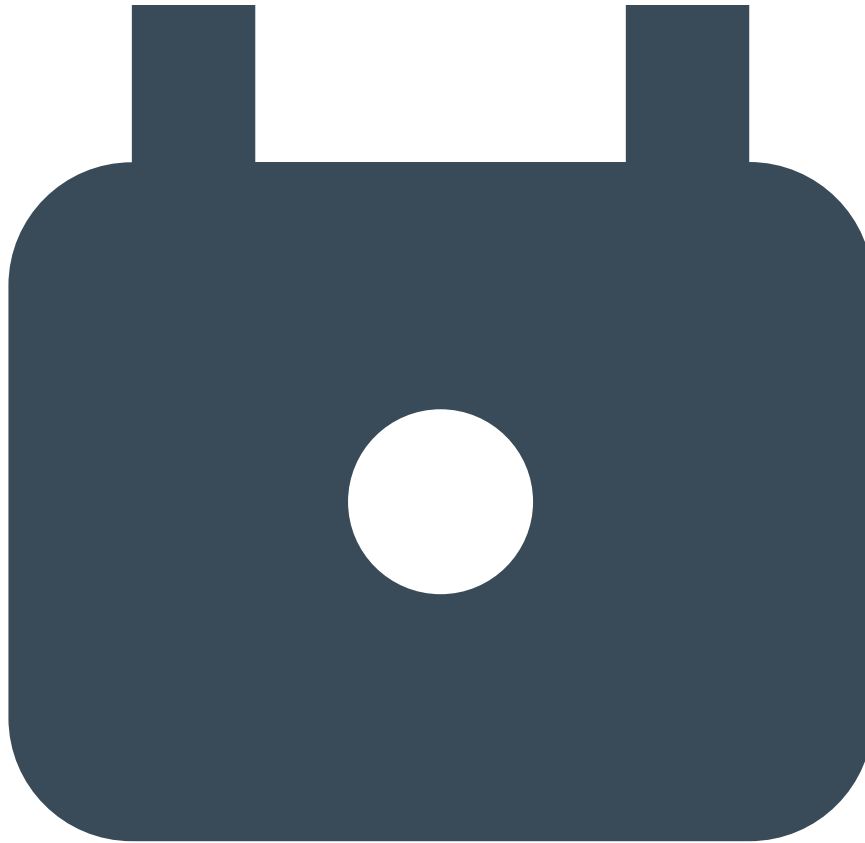
Denne innleveringen fokuserer på Pandas i Python og kombinere dette med Binære trær. Vi skal bruke namedtuple-varianten som dataene som skal brukes i treet under Unit-testingen.

Du får utlevert 2 kildekodefiler som sammen representerer et binært søketre, henholdsvis [BinaryTreeNode.py](#) og [BinaryTree.py](#). Kildekodene inneholder noen få endringer fra Innlevering 4, og denne gangen skal du endre deler av implementasjonen du får utlevert.

Du skal i denne innleveringen skal "[Personer.dta](#)" leses inn i og legges i det binære søketreet. Filen inneholder 100 000 rader med fiktive personalia på formatet "etternavn;fornavn;adresse;postnummer;poststed".

Oppgaven løses mest naturlig i en IDE, men du kan også gjøre dette i Jupyter Notebook. I denne sammenheng kan du laste ned alt via en egen notebook som du finner [her](#). En kombinasjon er nok det mest fornuftige siden du skal gjøre flere endringer. Jupyter Notebook ser ut som følger:





Oppgaven består av tre deler:

1. Innlesingsdel:

- A. Ved bruk av Pandas - les inn datafilen og legg dette i en collection av namedtuple
- B. Insert dataene (namedtuples) som du hentet i det binære søketreet
- C. Hver node i treet har en egen "level" knyttet til seg - altså nivå i treet. Skriv ut nivået og verdien for rad nr 1, 10, 100, 1000, 10000, 100000 fra filen Personer.dta.
Legg til side denne listen av namedtuples og nivået i treet

2. Endringsdel:

- A. I kildekoden [BinaryTree.py](#) er de fleste rutinene skrevet som rekursive implementasjoner, med unntak av deleteMin(), deleteMax(), og delete(). Skriv egen kode i [BinaryTree.py](#) for delete() slik at du bruker rekursjon i slettingen.
- B. Slett rad nr. 1000 og 10000 fra Personer.dta i det binære søketreet, og kontroller implementasjonen din mot resultatet fra oppgave 1C

3. Innleveringsdel:

Lever inn din Jupyter Lab Notebook, med alle dine endringer, som også inkluderer din egen kode med endringene i BinaryTree.py

Basert på dine endringer i BinaryTree.py, en utskrift av nivået for følgende rader i

Personer.dta: 8, 50, 400, 2300, 8000, 49999, 75000

Hvor mange noder befinner seg på hvert nivå/level i treet?

