

Obligatorisk oppgave nr.2

IDATG2102 – Algoritmiske metoder, høsten 2025

Innleveringsfrist: 23.september 2025 kl.11:00 (*må overholdes*)
via Blackboard

Bruk og se filen OBLIG2.cpp/java for hovedrisset til oblig'en.

NB: Det skal *ikke* innføres flere globale data eller datamedlemmer enn angitt i de to filene.
Det skal heller *ikke* brukes andre hjelpestrukturer - som array, stakk, kø eller liste.

Det skal lages/kodes to *helt* uavhengige (men mye lignende) funksjoner:

a) Lag den rekursive funksjonen

```
bool erTrerneIdentiske(Node* t1, Node* t2)
```

Funksjonen sjekker og returnerer om (sub)trærne er *helt* identiske. Dvs. at begge trærne har *eksakt* samme struktur (tilsvarende noder har samme forekomst av venstre og høyre barn) og ID i de tilsvarende nodene.

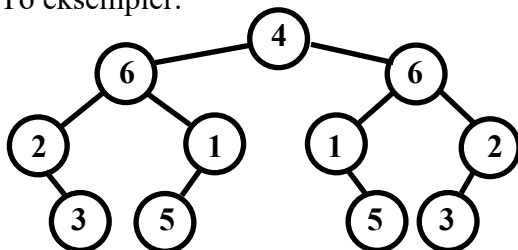
Funksjonen kalles initielt fra main ved: `erTrerneIdentiske(root1, root2)`

b) Lag den rekursive funksjonen

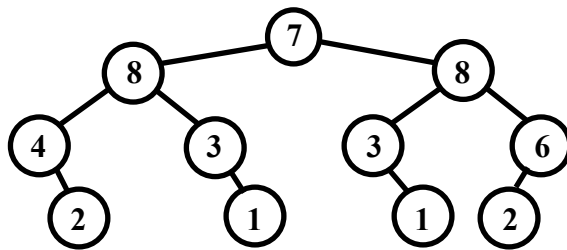
```
bool erTreetSymmetrisk(Node* t1, Node* t2)
```

Funksjonen sjekker og returnerer om treet er *totalt* symmetrisk/speilvendt (både i struktur og node-verdier) om aksene som går gjennom rota og loddrett ned (om treet er tegnet opp). Funksjonen kalles initielt fra main ved: `erTreetSymmetrisk(root1, root1)`
Legg altså merke til at i det første/initielle kallet er begge subtrærne hele treet selv (`root1`).

To eksempler:



Symmetrisk



Ikke symmetrisk

(4 og 6 er forskjellige. 1 er begge høyre barn.)

Kun koden (C++ eller Java) for de to funksjonene leveres som PDF i Blackboard