

Øving 2, Algoritmer og datastrukturer

Rekursiv programmering

1. Oppgave 2.1–1 på side 28 i boka (n er et heltall, x er et desimaltall)
2. Oppgave 2.2–3 på side 38 i boka
3. Beregn samme eksponent med programmeringsspråkets egen metode. (For Java: `Math.pow(x,n)`, for C: `pow(x, n)`). Sjekk at metodene deres regner riktig. Ta tiden her også, for å se hva som er raskest.

Tips:

- Ikke glem å forklare forskjellene i tidsforbruk mellom 2.1–1 og 2.1–3. Det kan f.eks. gjøres ved å finne kompleksiteten for begge programmene.
- I 2.2–3 beregner dere x^2 som $x*x$, for å unngå uendelig rekursjon.
- Både 2.1–1 og 2.2–3 skal løses med rekursive kall. Andre løsninger er mulig, men blir ikke godkjent fordi dette skal være trening i rekursjon!
- For å ta tiden på kjøring med stor n , kan x^n lett bli et veldig stort tall. F.eks. er 2^{64} større enn det største heltallet, men 64 er ikke noen stor eksponent. Derfor, la x være et desimaltall av typen `double`. Da kan programmet deres f.eks. beregne 1.001^{5000} uten å få problemer. Men n må være et heltall.
- Eksponenter går uansett fort å beregne, tidtakermetoden jeg viste dere i den første forelesningen kan komme godt med.