Søk

Anta at vi har tre prosedyrer:

- `Search` – som gjør et rett frem søk på lineær tid
- `BinarySearch` – som gjør et binærsøk
- `Sort` – sorterer i $\mathcal{O}(n \cdot \log(n))$

Prosedyrene nedenfor tar et array `A` med heltall av størrelse n og et array `B` av størrelse k som input. Alle skal printe tallene som er i både `A` og `B`.

```
1 Procedure Intersection1(A, B)
2     for x in B do
3         if Search(A, x) then
4             print(x)
```

```
1 Procedure Intersection2(A, B)
2     for x in B do
3         if BinarySearch(A, x) then
4             print(x)
```

```
1 Procedure Intersection3(A, B)
2     Sort(A)
3     for x in B do
4         if BinarySearch(A, x) then
5             print(x)
```

```
1 Procedure Intersection4(A, B)
2     for x in B do
3         Sort(A)
4         if BinarySearch(A, x) then
5             print(x)
```

Tre av disse prosedyrene gir alltid riktig svar, og én kan gi feil svar. Vi lurer på når man burde bruke hvilken variant av `Intersection` avhengig av hvor stor k (altså lengden av `B`) er i forhold til n (altså lengden til `A`). Du trenger bare å oppgi ett riktig svar hvis flere prosedyrer gir opphav til samme kjøretidskompleksitet.

(a) Hvis `B` inneholder ett element, altså $k = 1$, hvilken prosedyre gir riktig svar med

minimal kjøretidskompleksitet?

(b) Hvis `B` inneholder $\log(n)$ elementer, altså $k = \log(n)$, hvilken prosedyre gir riktig svar med minimal kjøretidskompleksitet?

(c) Hvis `B` inneholder n elementer, altså $k = n$, hvilken prosedyre gir riktig svar med minimal kjøretidskompleksitet?

(d) Hvis `B` inneholder n^2 elementer, altså $k = n^2$, hvilken prosedyre gir riktig svar med minimal kjøretidskompleksitet?