Sort()-algoritmen og dens effektivitet

For vår Sort()-algoritme har vi valgt å benytte oss av QuickSort med InsertionSort implementert. Vi definerer i algoritmen at dersom listen inneholder 5 elementer eller færre så benyttes InsertionSort, da denne er mer effektiv i mindre lister. Men fra Facebook og lignende ser man at vennelister gjerne kan inneholde opp til 5000 venner, så det ble vurdert viktig å bruke en mer effektiv algoritme enn InsertionSort for større vennelister.

Quicksort-algoritmen vi benytter oss av er rekursiv, og når de oppdelte listene kommer til 5 elementer tas InsertionSort automatisk i bruk. Derfor kan vi si at vi bruker en Big-O notasjon mellom InsertionSort og QuickSort, der QuickSort benyttes for større lister.

For InsertionSort er effektiviteten mellom O(n) og O(n2), der mengden av sortering reduserer arbeidet som er nødvendig.

For QuickSort er effektiviteten mellom O(n log n) til O(n2) avhengig av pivot som velges. Da vi bruker en metode der vi sammenligner 3 elementer spredd ut over listen til å velge pivot kan vi i de fleste tilfeller unngå verste utfall av denne strukturen.

Prosjektet vårt bruker derfor en QuickSort-algoritme som faller tilbake på InsertionSort ved svært små lister. Effektiviteten er mellom O(n log n) og O(n2) der flere virkemidler er tatt i bruk for å unngå en O(n2)-situasjon.