

## Rapport til øving 2

Quicksort er en den raskeste sorteringsalgoritmen man kjenner til. Den plukker ut en delingsverdi, hvor alt som er mindre plasseres under, mens alt som er større plasseres over. De to sidene sorteres rekursivt med quicksort. I dual pivot quicksort derimot benyttes to delingstall, og dette er fordelaktig i den forstand at man har tre delingsgrupper i motsetning til to. På den måten fordeles de tallene som er mindre enn det første delingstallet, de som er i midten av delingstallene og de som er større enn det største delingstallet. Denne algoritmen skal derfor i praksis være raskere siden det er færre sammenligninger og mindre rekursjon. Samtidig kan det være utfordrende å finne ut hvilken sorteringsalgoritme som fungerer best siden det kan avhenge av faktorer som pc og programmeringsspråk.

```
--- Test arrays of random numbers ---
```

Sort algorithm:	Time in ms:	Sum before:	Sum after:	Sum is equal:	Is sorted:	Array length:
Single pivot	0.2512	48439	48439	yes	yes	1000
Dual pivot	0.2909	49258	49258	yes	yes	1000
Single pivot	4212.3227	550056345	550056345	yes	yes	100000000
Dual pivot	3789.2978	550032888	550032888	yes	yes	100000000

Tabellen viser tidsmålinger for sortering av tabeller med tilfeldige tall. Hvor fort tabellene sorteres avgjøres av både lengden på tabellene og størrelsen på tallene. Summen av tallene i tabellen kan indikere hvor omfattende det er å summere tabellen.

I dette tilfellet, var single pivot raskere til å sortere en tabell av lengde 1000 enn det dual pivot var til å sortere enn annen tabell med samme lengde. Her var summen av begge tabellene nokså like, og single pivot slo dual pivot med omtrent 16 % i dette tilfellet. For en tabell av lengde 100 millioner, var dual pivot algoritmen raskere til å sortere med omtrent 11 %. Her var summen av tabellene mer eller mindre like, og det viser tydeligere hvilken algoritme som er raskest.

```
--- Test arrays of duplicates ---
```

Sort algorithm:	Time in ms:	Sum before:	Sum after:	Sum is equal:	Is sorted:	Array length:
Single pivot	0.0021	36050	36050	yes	yes	100
Dual pivot	0.0031	54650	54650	yes	yes	100
Single pivot	1.9389	25750000	25750000	yes	yes	100000
Dual pivot	1.8585	31950000	31950000	yes	yes	100000

For tabeller med annen hvert duplikat, var single pivot algoritmen betydelig raskere til å sortere en tabell av lengde 100. I dette tilfellet, var summen av de to tabellene varierende og dette kan ha spilt en vesentlig rolle på tiden. For en tabell av større lengde, var dual pivot raskere med omtrent 5 % til tross for at summen av tabellen var større enn summen av tabellen sortert av single pivot algoritmen. Dual pivot regnes av mange som den beste sorteringsalgoritmen, og på

tabeller med slike data skal den være raskere enn single pivot. Dette kommer frem når lengden på tabellen øker.

```
--- Test arrays that is sorted before ---
```

Sort algorithm:	Time in ms:	Sum before:	Sum after:	Sum is equal:	Is sorted:	Array length:
Single pivot	0.0257	48439	48439	yes	yes	1000
Dual pivot	0.0223	49258	49258	yes	yes	1000
Single pivot	3050.7402	550056345	550056345	yes	yes	100000000
Dual pivot	2806.0292	550032888	550032888	yes	yes	100000000
Single pivot	0.0021	36050	36050	yes	yes	100
Dual pivot	0.0029	54650	54650	yes	yes	100
Single pivot	1.8937	25750000	25750000	yes	yes	100000
Dual pivot	1.7652	31950000	31950000	yes	yes	100000

Dual pivot algoritmen var raskere å sortere tabeller som allerede var sortert fra før i tre av fire tilfeller. I det ene tilfellet hvor single pivot var raskere, var summen av tabellen betydelig mindre og dette kan ha vært årsaken til dette. De sorterte tabellene som i utgangspunktet var generert tilfeldig, tok vesentlig mindre tid å sortere på nytt. Tabellene med duplikater derimot forbedret tiden sin minimalt. I alle tilfellene ble tiden forbedret.

Alt i alt, er dual pivot algoritmen raskere i de fleste tilfellene. Single pivot var i raskere i enkelte av tilfellene hvor lengden av tabellene var mindre eller lik 1000. Først og fremst, kan dette komme av at mindre tabeller gir store forskjeller i tall når de genereres tilfeldig. Samtidig kan rekkefølgen på tallene være mer gunstig for sortering, altså at man har mer flaks med rekkefølgen slik at den er enklere å sortere. For tabeller av betydelig større lengde, vil både summen av tallene og hvor gunstig rekkefølgen er jevne seg ut drastisk. Dual pivot skal i praktisk være mellom 10 og 20 % raskere enn single pivot, og dette kommer tydeligere frem når lengden på tabellene blir betydelig større.