Intro algorithms

SD-c

Engels,Teun T.

2023

# Intro

In dit document leg in de meetresultaten van mijn 2 sorting algoritmes vast en verglijk ik de resultaten van de 2 verschillende algoritmes. Hierbij kijk ik naar het verband met de resultaten en het big-O model.

Ik heb gekozen voor het bubblesort algoritme en het merge sort algoritme.

Big-O Bubblesort:

Text

Description automatically generated with medium confidence

Figuur 1 <https://en.wikipedia.org/wiki/Bubble_sort>

Big-O Mergesort:

Text

Description automatically generated with medium confidence

Figuur 2 <https://en.wikipedia.org/wiki/Merge_sort>

## ! Belangrijk

Ik heb de db versie van de opdracht gemaakt dus moet deze iets anders worden gerund.

Voor het runnen van de unit test: /make test

# Metingen

We krijgen voor het testen/meten diverse files dus deze heb ik ook gebruikt om het verschil duidelijk te laten zien. Alle metingen zijn in seconden.

## Bubblesort

Voor het testen van mijn bubblesort implementatie:

/main 2 < testdata/testfile bijv in1\_4



Figuur 3 Getallen 1 t/m 100000 k = 1 (in1\_4)



Figuur 4 Getallen 1 t/m 1000 willekeurig k = 1 (int2\_4)

## Mergesort

Voor het testen van mijn bubblesort implementatie:

/main 3 < testdata/testfile bijv in1\_4



Figuur 5 Getallen 1 t/m 100000 k = 1 (in1\_4)



Figuur 6 Getallen 1 t/m 1000 willekeurig k = 1 (int2\_4)

# Conclusie

Als je van te voren al kijkt naar de BigO van deze 2 algoritmes was het al te verwachten dat mergesort een stuk sneller zou zijn.

Echter had ik niet verwacht dat mergesort ook op een al gesorteerde array zoveel sneller zou zijn maar aangezien bubblesort langzamer over alle 100000 elementen gaat is dit toch wel logisch. Het verschil wordt vooral duidelijk als je een nog niet gesorteerde array pakt.

Mergesort is dan al op een “kleine” array van 1000 elementen al bijna 10x sneller.