Laborationsrapport
Algoritmer och datastrukturer I , 7,5 HP B
HT2015

Laboration nr: 4
Exekveringstid for sorteringsalgoritmer

av

Viktor Hanstorp (19940413)

Institutionen för matematik, natur- och datavetenskap
Högskolan i Gävle

S-801 76 Gävle, Sweden

Email:
ndi14vhp@student.hig.se

Innehåll

1 Inledning

Laborationen går ut på att lära sig mer om sorteringsalgoritmer, användandet av empiriska studier för att analysera dem, samt dess ordonalitet.

2 Förutsättningar och krav

2.1 Uppgifter

- Lista ut vilken strategi som används av den givna implementationen av Quicksort för att välja pivotelement
- Motivera varför vissa tider avviker i det givna programmet
- Lista ut i vilken ordning som sorteringsalgoritmerna i det givna programmet körs.
- Visa antalet jämförelser utförda i sorteringsalgoritmerna med hjälp av artimetiska serier

2.2 Förutsättningar

De sorteringsalgoritmer som används är:

- Bubblesort
- Insertionsort
- Mergesort
- Quicksort

Algoritm	**Bäst**	**Medel**	**Värst**
Bubblesort	$O(n)$	$O(n^2)$	$O(n^2)$
Insertionsort	$O(n)$	$O(n^2)$	$O(n^2)$
Mergesort	$O(n \times log(n))$	$O(n \times log(n))$	$O(n \times log(n))$
Quicksort	$O(n \times log(n))$	$O(n \times log(n))$	$O(n^2)$

Tabell 1 - Tid för olika sorteringsalgoritmer

Enligt wikipedia [1] så är tidsåtgången för sorteringsalgoritmerna enligt Tabell 1

Artimetiska summan:

$$\sum_{i=1}^{a} f(i) = a \times \frac{f(1) + f(a)}{2}$$

3 Resultat

För framtagning av resultat för körningar av det givna programmet skapades ett program för att räkna ut medelvärdet för ett antal körningar (”AOD_Lab4_Runner”).

Programmet startar ”AOD_Lav4.exe” och extraherar tiderna från dess standard-output. Medelvärdet beräknas sedan.

Resultat kan ses i bilaga 1.

4 Diskussion

4.1 Uppgift 1

4.1.1 Fråga 1

Det första värdet väljs som pivotelement för denna implementation av Quicksort.

4.1.2 Fråga 2

Metod 1
”b.txt” och ”d.txt” ger lägre värden än förväntat.
Detta betyder att sorterade (ej omvänt) datamängder är optimalt.

Metod 2
”a.txt”, ”b.txt” och ”d.txt” avviker med att vara rejält mycket större än de borde.
Detta betyder att sorterade (även omvänt) datamängder ger dåligt resultat.

Metod 3
”b.txt” och ”d.txt” ger typ lägre värden än förväntat.
Detta betyder att sorterade (ej omvänt) datamänger är optimalt.

Metod 4
Alla värden ser jämna ut

4.1.3 Fråga 3

Mergesort är den enda sorteringsalgoritm där alla fall är lika. Detta kan endast observeras i metod 4.

”Divide and conquer” metoder ger de bästa tiderna för jämförelsebaserade sorteringar, $O(n \times log(n))$, detta kan ses i metod 2 och metod 4. Då metod 4 är mergesort så är metod 2 *Quicksort*.

Bubblesort är optimal då datamängden är sorterad, då det inte behöver göras några byten. Detta kan observeras i metod 3 (~ 0 för sorterat).

Uteslutningsmetoden säger att metod 1 är *Insertionsort*. Detta kan även bekräftas då sortering av sorterad datamängd tar längre tid än metod 3.

Metod 1 – Insertionsort
Metod 2 – Quicksort
Metod 3 – Bubblesort
Metod 4 – Mergesort

4.2 Uppgift 2

Uträkning av tidsåtgång för jämnföring i värsta fallet.
Gällande givna algoritmer.

4.2.1 Insertionsort

$$T(N) = \sum_{i=1}^{N-1} i = (N-1) \times \frac{1-(N-1)}{2} = \frac{N \times (N-1)}{2} = \frac{N^2 - N}{2}$$

$$O(T(N)) = N^2$$

4.2.2 Quicksort

$$T(N) = \sum_{i=1}^{N} N - i = N \times \frac{(N-1)+(N-N)}{2} = N \times \frac{N-1}{2} = \frac{N^2 - N}{2}$$

$$O(T(N)) = N^2$$

4.2.3 Bubblesort

$$T(N) = \sum_{i=1}^{N} N = N \times \frac{N+N}{2} = N \times \frac{2 \times N}{2} = N \times N = N^2$$

$$O(T(N)) = N^2$$

4.2.4 Mergesort

$$T(N) = \sum_{i=1}^{log_2(N)} \frac{N}{2^{(i-1)}} \times 2^{(i-1)} = \sum_{i=1}^{log_2(N)} N = log_2(N) \times \frac{N+N}{2} = log_2(N) \times \frac{2 \times N}{2}$$
$$= log_2(N) \times \mathrm{N}$$

$$O(T(N)) = N \times log(N)$$

5 Sammanfattning

Resultatet av uträkningarna i Uppgift 2 stämmer med teorin [1], vilket betyder att det kan vara korrekt.

6 Referenser

[1] Wikipedia, "Sorting algorithm --- Wikipedia{,} The Free Encyclopedia," 2015. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Sorting_algorithm&oldid=694927729. [Accessed 20 12 2015].

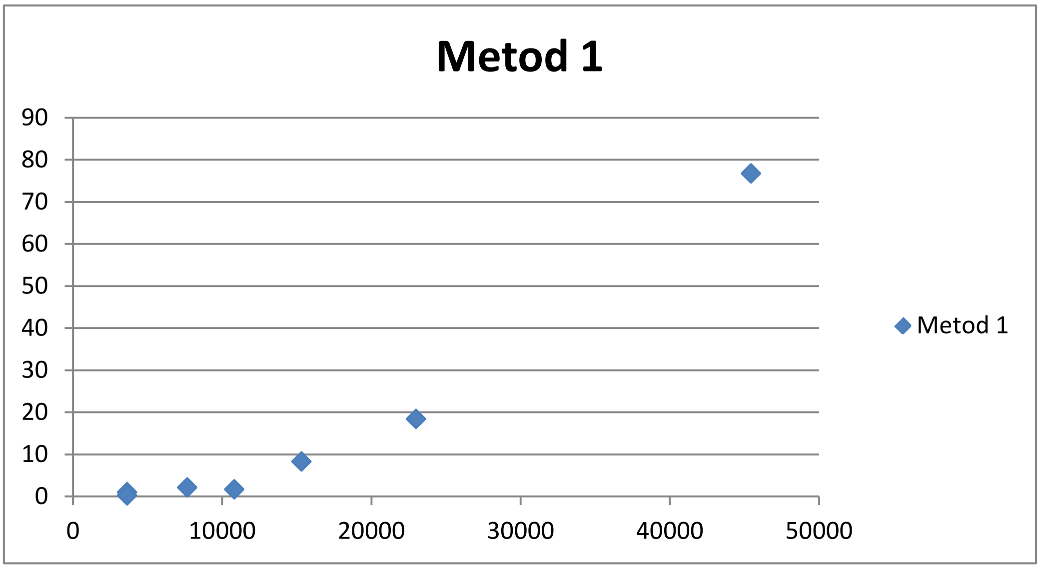
Bilaga 1

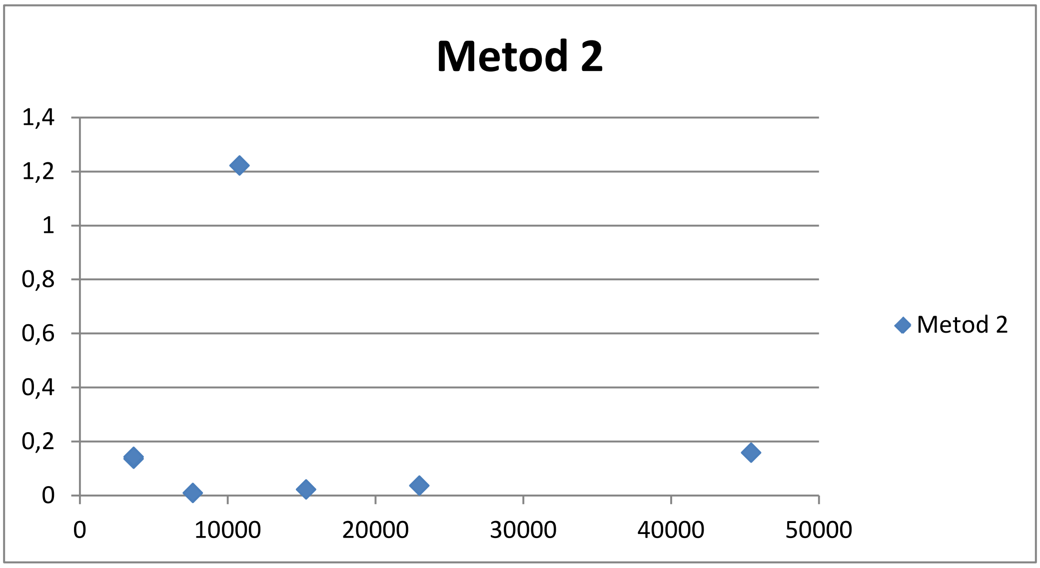
	a	b	c	d	e	f	g
Metod 1	0,9548	0,1754	2,114	1,6748	8,3014	18,3756	76,698
Metod 2	0,136	0,1432	0,0094	1,2214	0,0218	0,0362	0,1582
Metod 3	1,1444	0	3,1122	4,00E-04	11,9468	25,9896	110,8026
Metod 4	0,0054	0,0054	0,012	0,018	0,026	0,0402	0,086

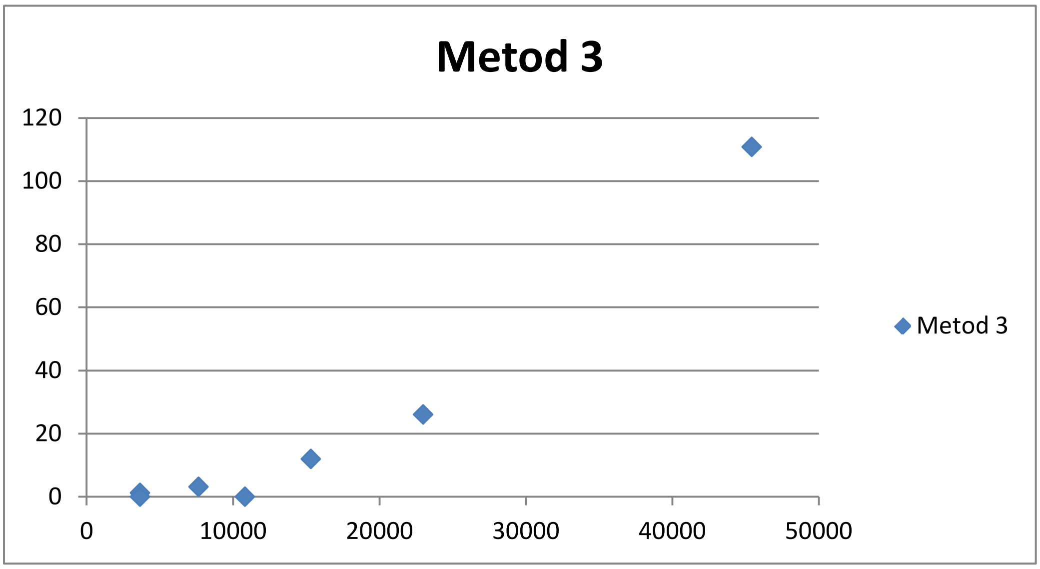
	Data typ	Storlek
a	omvänt	3645
b	sorterat	3645
c	slump	7664
d	sorterat	10821
e	slump	15327
f	slump	22990
g	slump	45439

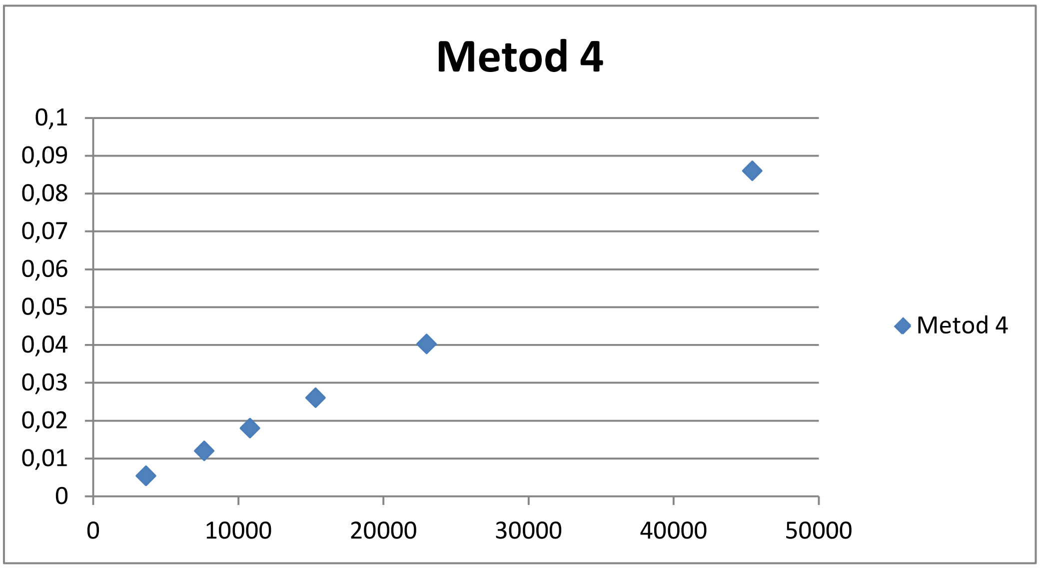
5 gånger

	Storlek	Metod 1	Metod 2	Metod 3	Metod 4		
a	3645	0,9548	0,136	1,1444	0,0054		omvänt
b	3645	0,1754	0,1432	0	0,0054		sorterat
c	7664	2,114	0,0094	3,1122	0,012		slump
d	10821	1,6748	1,2214	4,00E-04	0,018		sorterat
e	15327	8,3014	0,0218	11,9468	0,026		slump
f	22990	18,3756	0,0362	25,9896	0,0402		slump
g	45439	76,698	0,1582	110,8026	0,086		slump




Metod 2
1,4
1,2
1
0,8
0,6
0,4
0,2
0
0
10000
20000
30000
40000
50000
Metod 2


Metod 3
120
100
80
60
40
20
0
0
10000
20000
30000
40000
50000
Metod 3


Metod 4
0,1
0,09
0,08
0,07
0,06
0,05
0,04
0,03
0,02
0,01
0
0
10000
20000
30000
40000
50000
Metod 4