

Opgaveløsninger (sæt 5)

Opgave 1a (9.4)

Hvis alle elementer er ens, vil betingelserne $a[++i] < v$ og $a[++j] > v$ altid have værdien false i den indre løkke (se koden på side 118 i lærebogen). Der sker derfor $N/2$ ombytninger og N sammenligninger i første inkarnation af metoden, og i og j peger på midterelementet før de rekursive kald. Antallet af ombytninger X_N opfylder derfor rekursionsligningen

$$X_N = 2X_{N/2} + N/2$$

mens antallet af sammenligninger C_N opfylder rekursionsligningen

$$C_N = 2C_{N/2} + N$$

som henholdsvis har løsninger

$$X_N = N/2 \lg N$$

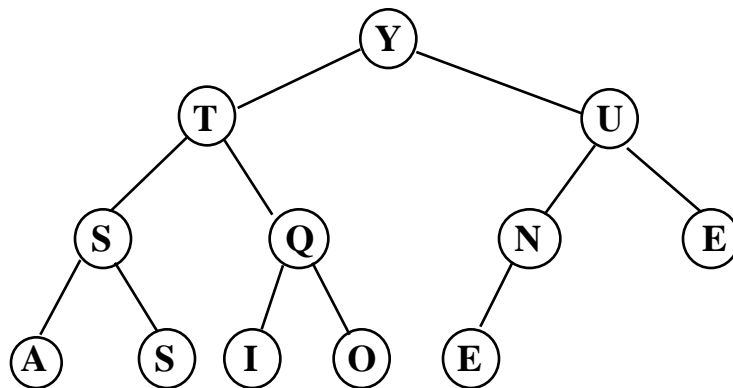
og

$$C_N = N \lg N$$

Køretiden bliver derfor $O(N \lg N)$.

Opgave 1b (9.9)

Quicksort opdeler datamængden i to dele, der sorteres hver for sig, helt uden interferens imellem delene. Derfor er det ligegyldigt, i hvilken rækkefølge delopgaverne løses. Man kan altså godt anvende en kø i stedet for en stak.

Opgave 2a (11.3)**Opgave 2b (11.6)**

Hvis posterne gives prioriteter, der svarer til deres “ankomsttidspunkt” bliver prioritetskøen en stak. For eksempel kunne prioriteten være antallet af indsættelser i prioritetskøen. Ved inspektion af koden for indsættelse på side 150 i lærebogen ses, at ens prioriteter for alle poster også vil bevirke, at prioritetskøen fungerer som en stak.

Hvis posterne derimod gives aftagende prioriteter, efterhånden som de indsættes, bliver prioritetskøen en normal kø. For eksempel kunne prioriteten være antallet af indsættelser med modsat fortegn.

Opgave 3a (12.4)

Køretiden for mergesort er uafhængig posternes nøgleverdier. Såvel enhver opsplitting i delproblemer som antallet af operationer i enhver fletning er uafhængig af nøgleverdier.

Opgave 3b (12.7)

E	A	S	Y	Q	U	E	S	T	I	O	N
A	E										
		S									
A	E	S									
			Q	Y							
					U						
A	E	Q	Q	U	Y						
			S	U	Y						
						E	S				
								T			
						E	S	T			
									I	O	
											N
									I	N	O
A	E	E	I	N	O	E	I	N	O	S	T
						Q	S	S	T	U	Y