Fakultät für Informatik
Lehrstuhl 11
Prof. Dr. Kevin Buchin
Dr. Carolin Rehs
Antonia Kalb
Erik Thordsen

Übungen zur Vorlesung

Datenstrukturen, Algorithmen und Programmierung 2

Sommersemester 2024

Übungsblatt 1

Besprechungszeit:
22. – 26.04.2024

Aufgabe 1.1 – Insertionsort (10 Punkte)

Führen Sie den Insertionsort auf das Array $A = [5, 3, 7, 1, 6, 4, 8, 2]$ entsprechend des Pseudocodes der Vorlesung aus. Füllen Sie dazu die Tabelle aus. Geben Sie jeweils zum **Ende der äußeren Schleife** key, i und A an.

j	key	i	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8
			5	3	7	1	6	4	8	2
2										
3										
4										
5										
6										
7										
8										

Aufgabe 1.2 – Schleifeninvariante und Korrektheitsbeweis (10 Punkte)

Betrachten Sie das folgende Programm. Das Programm erhält als Eingabe ein Array mit n Zahlen aus $\mathbb{N}$.

Algorithmus 1 Einfaches Programm

Eingabe: Array $A[1 \ldots n]$
Ausgabe: Integer v

```
n ← length[A]
v ← 0
for i ← 1 to n do
    if A[i] mod 2 = 1 then
        A[i] ← A[i]+1
    end if
    v ← v+A[i]
end for
return v
```

Beweisen Sie, dass das Programm korrekterweise eine gerade Zahl ausgibt.
Gehen Sie folgendermaßen vor:

a) Formulieren Sie eine Schleifeninvariante, die nach jedem Durchlauf der obigen Schleife gilt, und beweisen Sie diese.

   Wichtig: Die Schleifeninvariante soll unmittelbar für den Beweis in b) nützlich sein.

b) Verwenden Sie dann die Schleifeninvariante, um zu zeigen, dass das Programm eine gerade Zahl ausgibt.

Aufgabe 1.3 – Mergesort (10 Punkte)

Führen Sie den Mergesort auf das Array $A = [5, 3, 7, 1, 6, 4, 8, 2]$ aus. Füllen Sie dazu die vorgefertigten Felder aus.

$$\text{Sort}(A, l=\boxed{1}, r=\boxed{8})$$

A= $\boxed{5}\ \boxed{3}\ \boxed{7}\ \boxed{1}\ \boxed{6}\ \boxed{4}\ \boxed{8}\ \boxed{2}$

$\text{Sort}(A, l=\boxed{}, m=\boxed{})$ $\text{Sort}(A, m+1=\boxed{}, r=\boxed{})$

A= $\boxed{}\ \boxed{}\ \boxed{}\ \boxed{}$ $\boxed{}\ \boxed{}\ \boxed{}\ \boxed{}$

$\text{Sort}(A,\boxed{},\boxed{})$ $\text{Sort}(A,\boxed{},\boxed{})$ $\text{Sort}(A,\boxed{},\boxed{})$ $\text{Sort}(A,\boxed{},\boxed{})$

A= $\boxed{}\ \boxed{}\ \boxed{}\ \boxed{}\ \boxed{}\ \boxed{}\ \boxed{}\ \boxed{}$

l	m	m+1	r	l	m	m+1	r	l	m	m+1	r	l	m	m+1	r
$\boxed{}$	$\boxed{}$	$\boxed{}$	$\boxed{}$	$\boxed{}$	$\boxed{}$	$\boxed{}$	$\boxed{}$	$\boxed{}$	$\boxed{}$	$\boxed{}$	$\boxed{}$	$\boxed{}$	$\boxed{}$	$\boxed{}$	$\boxed{}$

A= $\boxed{}\ \boxed{}\ \boxed{}\ \boxed{}\ \boxed{}\ \boxed{}\ \boxed{}\ \boxed{}$

$\text{Merge}(A,\boxed{},\boxed{},\boxed{})$ $\text{Merge}(A,\boxed{},\boxed{},\boxed{})$ $\text{Merge}(A,\boxed{},\boxed{},\boxed{})$ $\text{Merge}(A,\boxed{},\boxed{},\boxed{})$

A= $\boxed{}\ \boxed{}\ \boxed{}\ \boxed{}\ \boxed{}\ \boxed{}\ \boxed{}\ \boxed{}$

$\text{Merge}(A, l=\boxed{}, m=\boxed{}, r=\boxed{})$ $\text{Merge}(A, l=\boxed{}, m=\boxed{}, r=\boxed{})$

A= $\boxed{}\ \boxed{}\ \boxed{}\ \boxed{}$ $\boxed{}\ \boxed{}\ \boxed{}\ \boxed{}$

$$\text{Merge}(A, l=\boxed{1}, m=\boxed{4}, r=\boxed{8})$$

A= $\boxed{1}\ \boxed{2}\ \boxed{3}\ \boxed{4}\ \boxed{5}\ \boxed{6}\ \boxed{7}\ \boxed{8}$

Aufgabe 1.4 – Teile und Herrsche und Korrektheitsbeweis (10 Punkte)

Gegeben sei der folgende rekursive Sortieralgorithmus:

Algorithmus 2 Sort

Eingabe: Array A, Integer l, Integer r
Ausgabe: Array A

```
if l < r − 1 then
    k ← (r − l + 1) div 3
    Sort(A, l, r − k)
    Sort(A, l + k, r)
    Sort(A, l, r − k)
else
    if A[l] > A[r] then
        temp ← A[l]
        A[l] ← A[r]
        A[r] ← temp
    end if
end if
```

Die Sortierung des Arrays A erfolgt durch den Aufruf Sort$(A, 1, \mathrm{length}\ [A])$.

Beweisen Sie die Korrektheit des Algorithmus mittels vollständiger Induktion. Nehmen Sie für die Eingabelänge n des Arrays eine Dreierpotenz $n = 3^k$ an.