



CHRISTIAN-ALBRECHTS-UNIVERSITÄT ZU KIEL

Institut für Informatik, Arbeitsgruppe Theorie der Parallelität

Prof. Dr. K. Jansen, K.-M. Klein, M.Maack

23. April 2015

Präsenzaufgaben zur Vorlesung »Algorithmen und Datenstrukturen«

Blatt 3

Aufgabe 3.1 (Selection Sort) Führen Sie für folgendes Array den Algorithmus zu Selection Sort durch:

$A = [4, 3, 2, 4, 1, 42, 23, 1]$

Geben Sie dabei jede Änderung des Arrays und die Werte der Variablen i und minindex nach durchlauf der inneren for -Schleife an.

Aufgabe 3.2 (Binäre Suche) Führen Sie für folgendes Array und $x=67$ sowie $x=9$ den Algorithmus zu binärer Suche durch:

$A = [7, 8, 13, 16, 19, 23, 32, 37, 41, 64, 67, 71, 128]$

Geben Sie dabei jede Änderung der Variablenwerte an.

Aufgabe 3.3 (\mathcal{O} -Notation) Seien $a, b \in \mathbb{R}_{>1}$. Zeigen Sie, dass $\log_a(n+1) \in \Theta(\log_b(n+1))$ gilt.

Hinweis: $\log_c(n)$ ist definiert als der Wert x , für den $c^x = n$ gilt.

Aufgabe 3.4 (Rekurrenzgleichung) Seien $\alpha, \beta \in \mathbb{N}_{>0}$ gegeben und sei $f : \mathbb{N}_{>0} \rightarrow \mathbb{N}_{>0}$ eine Abbildung mit

$$f(n) \leq \begin{cases} \alpha & \text{falls } n = 1 \\ \beta + f(\lfloor n/2 \rfloor) & \text{sonst} \end{cases}$$

für alle $n \in \mathbb{N}$. Zeigen Sie, dass

$$f(n) \leq \alpha + \beta \lfloor \log_2(n) \rfloor$$

für alle $n \in \mathbb{N}$ gilt.

Algorithmus BINARY-SEARCH (A,x)

```
1 integer n = length(A);
2 integer i = 0, j=n-1, mid;
3 boolean found = false;
4 while ((i<=j) ^ not(found)) do
5     mid = (i+j) div 2;
6     if (x < A[mid]) then
7         j = mid - 1;
8     else
9         if (x > A[mid]) then
10            i = mid + 1;
11        else
12            found = true;
13        fi
14    fi
15 od
16 return found
```

Algorithmus SELECTION-SORT(A)

```
1 integer minindex,min,temp;
2 integer n = length(A);
3 for i=0 to n-2 do
4     minindex = i;
5     min = A[i];
6     for j=i+1 to n-1 do
7         if (A[j] < min) then
8             min = A[j];
9             minindex = j;
10        fi
11    od
12    temp = A[i];
13    A[i] = A[minindex];
14    A[minindex] = temp;
15 od
```