

Dr. Herbert Süße

Lehrstuhl "Digitale Bildverarbeitung"

Jena, den 18.6.2007

Übungsaufgaben im Fach "Einführung in die Informatik für Physikstudenten" zur Vorbereitung auf die Klausur

18. Mit Distribution-Sort sind Namen zu sortieren, die aus genau 8 Zeichen bestehen. Die zugelassenen Zeichen sind die Buchstaben A,b,c,D,f und Z, die Ziffern 6,8,0,4 und das Leerzeichen sowie das Semikolon. Wie errechnet man aus einer konkreten gegebenen Zeichenkette von 8 zulässigen Zeichen die Adresse, bzw. die Nummer des Speicherplatzes, auf dem eine "Eins" zu inkrementieren ist?

19. Schreiben Sie eine rekursive C/C++ -Funktion zum Finden des Maximums eines eindimensionalen Arrays von char.

20. Schreiben Sie eine rekursive C/C++ -Funktion zum Finden des größten gemeinsamen Teilers zweier positiver ganzer Zahlen mit dem bekannten Euklidischen Algorithmus.

21. Die rekursive Funktion zur Berechnung der Fakultät soll folgendermaßen modifiziert worden sein:

Als Abbruch dient immer die "triviale Lösung". Wenn der Abbruch nun mit $n=5$ programmiert ist, da $5!=120$ ist und dies auch eine einfache Lösung darstellt, was für Konsequenzen würden sich beim Aufruf dieser rekursiven Funktion ergeben?

22. Was berechnet folgender Algorithmus, der durch den folgenden Quelltext dargestellt wird:

```
int f(int p, int q)
{if (p==0) return 2; else
return (q * f(p-1,q); }
```

23. Ein Algorithmus hat die Zeitkomplexität $O(n)$, ein anderer $O(n^2)$, Welcher ist bei $n = 12$ schneller?

24. Ein Feld von Zeichenketten ist mit Bubble-Sort zu sortieren. Wie sieht der "best case" und der "worst case" aus?

25. Wir bauen einen sortierten, binären Baum aus einer gegebenen Sequenz von Zeichenketten auf. Geben Sie Beispiele an für den "worst case", d.h. der Baum entartet in eine lineare Liste, und für den "best case", d.h. der Baum ist "voll",

d.h. jeder Knoten (bis auf die Blätter und die Knoten der vorletzten Ebene) hat genau 2 Nachfolger.

26. Die Komplexität von rekursiven Algorithmen läßt sich durch rekursive Relationen ausdrücken, die man lösen muß. Welche Komplexitäten besitzen folgende Algorithmen:

a) $T(n) = T(n - 1) + n, T(1) = 1$

b) $T(n) = T(n/2) + 1, T(1) = 0$

c) $T(n) = T(n/2) + n, T(1) = 0$

d) $T(n) = 2T(n/2) + n, T(1) = 0$