

Übungsblätter zur Vorlesung Algorithmen und Datenstrukturen  
(Magister)  
Sommersemester 2002

(Prof. Der)

**Abgabetermin: 10.05. 2002 vor der Vorlesung**

---

Serie 2

1. Aufgabe (3 Pkte.): Die Fakultät einer Zahl  $n$  ist durch folgende rekursive Funktion gegeben:

```
int fac(int n){  
if (n < 1) {return 1;} else {return n*fac(n-1);}  
}
```

Fragen: Wie sieht die Rekursionsgleichung für die Zeitkomplexität aus? (1 P) Leiten Sie die bestmögliche obere Schranke aus der Rekursionsgleichung ab. (2 P)

2. Aufgabe (6 Pkte.): Vergleichen Sie die Zeitkomplexität  $T(n)$  für die Berechnung eines Polynoms

$$y(x) = \sum_{i=0}^n a_i x^i$$

mit  $n \geq 0$ , wenn die Berechnung

- (a) gemäß der obigen Formel, wobei keine Potenzen von  $x$  zwischengespeichert werden und  
(b) nach dem Hornerischen Schema erfolgt:

$$y(x) = \sum_{i=0}^n a_i x^i = ((\dots (a_n x + a_{n-1}) x + \dots + a_2) x + a_1) + a_0$$

Es sollen nur die vier Grundrechenarten gezählt und je Operation eine Zeiteinheit angerechnet werden.