

---

**Familienname:**

**Vorname:**

**Matrikelnummer:**

Aufgabe 1 (3 Punkte):  
Aufgabe 2 (4 Punkte):  
Aufgabe 3 (3 Punkte):  
Aufgabe 4 (1 Punkte):  
Aufgabe 5 (2 Punkte):  
Aufgabe 6 (5 Punkte):  
Aufgabe 7 (5 Punkte):  
Aufgabe 8 (7 Punkte):

---

Gesamtpunktzahl:

---

**Schriftlicher Nachtest zu C (90 Minuten)**  
**VU Einführung ins Programmieren für TM**

**01. März 2011**

---

**Aufgabe 1 (3 Punkte).** Schreiben Sie einen Strukturdatentyp `pol` zur Speicherung von Polynomen  $p(x) = \sum_{j=0}^n a_j x^j$ . In der Struktur sollen neben dem Grad  $n \in \mathbb{N}$  auch die  $(n + 1)$  Koeffizienten  $a_j \in \mathbb{R}$  gespeichert werden. Diese Struktur soll auch in den übrigen Aufgaben des Tests verwendet werden.

**Aufgabe 2 (4 Punkte).** Schreiben Sie eine Funktion `newPol`, die für gegebenen Grad  $n$  ein neues Polynom allokiert und initialisiert.

**Aufgabe 3 (3 Punkte).** Schreiben Sie eine Funktion `delPol`, die den Speicher für ein dynamisch allokiertes Polynom freigibt und den `NULL`-Pointer zurückgibt.

**Aufgabe 4 (1 Punkt).** Schreiben Sie eine Funktion `getPolDeg`, die den Grad eines Polynoms zurückgibt.

**Aufgabe 5 (2 Punkte).** Schreiben Sie eine Funktion `getPolCoeff`, die für gegebenes  $j$  den Koeffizienten  $a_j$  eines Polynoms zurückgibt. Im Fall, dass  $j$  größer als der Grad  $n$  des Polynoms ist, soll 0 zurückgegeben werden.

**Aufgabe 6 (5 Punkte).** Schreiben Sie eine Funktion `evalPol`, die für gegebenes Polynom  $p$  und Auswertungspunkt  $x$  den Funktionswert  $p(x)$  zurückgibt. Dabei soll die Funktion `pow` *nicht* verwendet werden.

**Aufgabe 7 (5 Punkte).** Schreiben Sie eine Funktion `evalPolPrime`, die für gegebenes Polynom  $p$  und Auswertungspunkt  $x$  den Funktionswert  $p'(x)$  der Ableitung zurückgibt. Dabei soll die Funktion `pow` *nicht* verwendet werden.

**Aufgabe 8 (7 Punkte).** Eine Möglichkeit eine Nullstelle eines Polynoms  $p$  zu berechnen ist das Newton-Verfahren. Ausgehend von einem Startwert  $x_0$  definiert man induktiv

$$x_{k+1} = x_k - p(x_k)/p'(x_k).$$

Man schreibe eine Funktion `findRootPol`, die zu gegebenen Polynom  $p$ , Startwert  $x_0$  und Toleranz  $\tau$  das Newtonverfahren durchführt, wobei die Iteration abgebrochen wird, falls

$$|p(x_n)| \leq \tau \quad \text{und} \quad |x_n - x_{n-1}| \leq \tau$$

gilt.