

# Übungsblatt 4

Abgabe: 27.11.2006

---

## Aufgabe 1: Links oder Rechts?

(40%)

Schreibt eine Prozedur *binaereSuche*, die ein vom kleinsten zum größten Element sortiertes Array von Integer-Zahlen sowie eine zu suchende Integer-Zahl übergeben bekommt. Die Prozedur soll das Array nach der Zahl durchsuchen und auf dem Bildschirm ausgeben, ob die Zahl im Array enthalten ist oder nicht, sowie gegebenenfalls den Index, unter dem die Zahl im Array zu finden ist. Beim Durchsuchen soll das Prinzip der binären Suche angewendet werden. Dabei wird die zu suchende Zahl zunächst mit einem in etwa in der Mitte liegenden Element des Arrays verglichen. Je nachdem, ob die zu suchende Zahl kleiner oder größer als dieses Element ist, wird dann auf die gleiche Weise für den Teil des Arrays, der links bzw. rechts des Vergleichselements liegt, verfahren. Somit halbiert sich bei jedem Schritt die Zahl der betrachteten Elemente des Arrays.

Eure Implementierung soll das Problem iterativ lösen - nicht mittels Rekursion.

## Aufgabe 2: Wo ist die Null?

(60%)

**Beschreibung des Verfahrens** Fast bei jeder Kurvendiskussion stösst man auf das Problem, Nullstellen zu bestimmen, also Lösungen der Gleichung  $f(x) = 0$  für eine gegebene Funktion  $f$  zu suchen. Das Newton-Verfahren ist eine Methode für die numerische Lösung von Gleichungen dieses Typs.

Vorgegeben ist eine stetige, differenzierbare Funktion  $f(x)$ , die mindestens eine Nullstelle hat und ein Startwert  $x_0$ , der nah genug an der Nullstelle liegt.

Der Berührungspunkt der Kurventangente an der Stelle  $x = x_0$  hat die Koordinaten  $(x, f(x))$  und die Steigung von  $f$  in diesem Punkt beträgt  $f'(x)$ . Der Schnittpunkt  $x_1$  dieser Tangente mit der x-Achse stellt eine erste Näherung für die Nullstelle dar und berechnet sich durch

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}.$$

Durch Wiederholung dieser Konstruktion erhält man eine Folge (hoffentlich) besser werdender Näherungen mit der allgemeinen Formel:

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}.$$

## Aufgabenstellung

Schreibt ein Programm, dass mit Hilfe des Newton-Verfahrens die näherungsweise Nullstellen der folgenden Gleichungen berechnen kann:

1.  $f(x) = \sin(x) + x - 1$  (Startwert  $x_0 = 3$ )

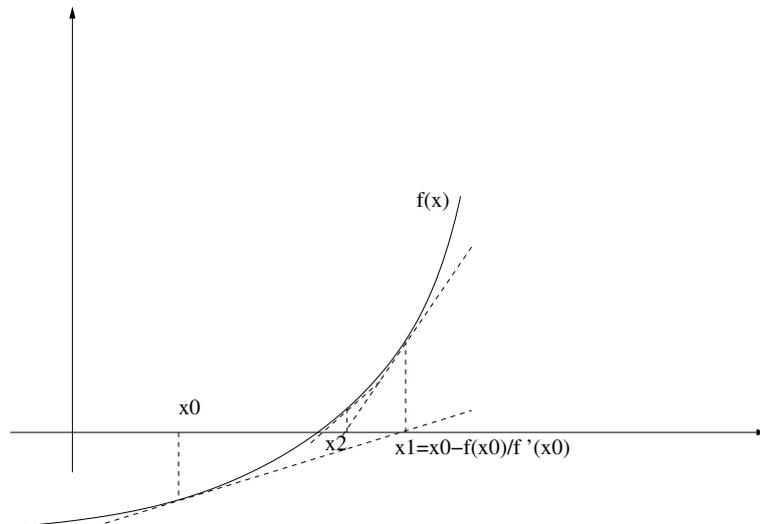


Abbildung 1: Graphische Darstellung des Newton-Verfahrens

2.  $f(x) = x^3 + x - 1$  (Startwert  $x_0 = 1$ )

Beschränkt Eure Berechnung dabei durch Festlegung einer maximalen Anzahl von Iterationen. Ausserdem soll die gewünschte Genauigkeit  $\varepsilon$  der Näherungslösung festgelegt werden, d.h. wenn  $|x_i - x_{i+1}| \leq \varepsilon$  gilt, ist  $x_{i+1}$  die gesuchte Lösung. Die Variablen zur Berechnung sollen vom Typ *double* sein, um eine höhere Genauigkeit zu erreichen.

Gibt es Situationen, in denen das Verfahren scheitern kann?

(Die Funktionen *sin* und *cos* (für die Ableitung) sind in `import java.lang.Math` definiert und werden als `Math.sin(...)` bzw. `Math.cos(...)` aufgerufen.)