



Einführung in die Programmierung für Studierende der Naturwissenschaften

Blatt 3 – 16.05.2022

Abgabe: bis Montag, 23. Mai, 16 Uhr per e-Mail an Ihre:n Tutor:in.

Aufgabe 1 (5 Punkte). Betrachten Sie das folgende C++-Programm

```
1 int main() {
2     int x = 17, y = 8;
3
4     int* a = &x;
5     int& b = y;
6
7     *a = b*8;
8
9     int* z1, z2;
10 }
```

Stellen Sie, wie in der Vorlesung, eine Tabelle mit den Spalten *Variablenname*, *Typ*, *Speicheradresse* und *Wert* auf. Nehmen Sie (vereinfachend) an, dass jeder Datentyp/jede Variable die Größe 1 hat und vergeben Sie fortlaufende Speicheradressen in der Reihenfolge, in der die Variablen angelegt werden. Die Werte sollen denen *unmittelbar vor dem Ende* des Programms.

Aufgabe 2 (2 Punkte). Schreiben Sie ein Programm, das eine Zahl n vom Terminal einliest. Anschließend soll ein Array mit n Gleitkommazahlen angelegt und n Zahlen von der Konsole eingelesen und im Array gespeichert werden. Danach soll das Programm den Durchschnitt, das Minimum und das Maximum der Zahlen ausrechnen, im Terminal ausgeben und das Array wieder löschen.

Aufgabe 3 (5 Punkte). Schreiben Sie eine Funktion

```
1     int pq(double p, double q, double* x1, double* x2)
```

die mit der pq -Formel die Nullstellen der quadratischen Funktion

$$f(x) = x^2 + px + q$$

berechnet. Dabei soll der Rückgabewert die Anzahl der Nullstellen sein (0, 1 oder 2). Wenn die Adressen $x1$, $x2$ nicht NULL sind, sollen dort die Nullstellen gespeichert werden, sofern es welche gibt.

Aufgabe 4 ((3 + 5) Punkte).

(a) Implementieren Sie eine Funktion `forward_diff_quot` die als Argumente Gleitkommazahlen x , h sowie eine Funktion f (als Funktionspointer) übergeben bekommt und die nach der Formel

$$f'(x) \approx \frac{f(x+h) - f(x)}{h}$$

angenäherte Ableitung von f an der Stelle x als Gleitkommazahl zurück gibt.

(b) Das Newton-Verfahren um finden einer Nullstelle x^* einer Funktion f geht so:

Eingabe: Reelle Zahl x_0 und Toleranz $\delta > 0$.

- (1) Setze $x_{alt} = x_0$,
- (2) Wenn $|f(x_{alt})| < \delta$: Stoppe
- (3) Setze $x_{neu} = x_{alt} - f(x_{alt})/f'(x_{alt})$.
- (4) Setze $x_{alt} = x_{neu}$
- (5) Gehe zu Schritt (1).

Ausgabe: Approximation x_k von x^* .

Implementieren Sie das Verfahren und testen Sie es beispielsweise an einer quadratischen Funktion wie in Aufgabe 3!

Hinweis: Sie dürfen das Verfahren mit einem x_0 in der Nähe der tatsächlichen Nullstellen initialisieren. Sie müssen nicht überprüfen, ob überhaupt Nullstellen existieren. Zum Berechnen der Ableitung können Sie die Approximation aus Teil (a) verwenden. Den Betrag $|f(\cdot)|$ können Sie ausrechnen oder `cmath` einbinden und die Funktion `std::abs` verwenden.

Aufgabe 5 (optional). Schildern Sie in einer Datei `erfahrung.txt` kurz Ihre Erfahrung mit dem aktuellen Übungsblatt. Berichten Sie darin wieder in Stichpunkten bzw. ein bis zwei kurzen Sätzen über Ihre Erfahrungen mit Kursinhalt und Übungsaufgaben. Was fiel Ihnen leicht? Was ist noch unklar? Wie viel Zeit haben Sie für die Bearbeitung der Hausaufgaben benötigt und welche Probleme traten dabei auf?