

Algorithmische Mathematik I

6. Übung

- Es sei $F = F(b, m, E_{\min}, E_{\max})$ ein Maschinenzahlbereich mit $m > 1$. Zeigen Sie:
 - $\text{eps}(F) \notin F$.
 - Sei rd eine Rundung zu F und $x \in \text{range}(F)$. Zeigen Sie, dass dann ein $\varepsilon \in \mathbb{R}$ existiert mit $|\varepsilon| \leq \text{eps}(F)$ und $\text{rd}(x) = x \cdot (1 + \varepsilon)$. (3+2 Punkte)
- Was ist die kleinste natürliche Zahl, die nicht in F_{double} ist? (1 Punkt)
- Wenn eps die Maschinengenauigkeit eines Maschinenzahlbereichs ist, wie viele signifikante Dezimalstellen hat eine von 0 verschiedene Zahl in diesem Maschinenzahlbereich dann mindestens? (3 Punkte)
- Betrachten Sie folgendes Problem: Es sei eine Funktion $f : [0, 1] \rightarrow \mathbb{R}$ gegeben, so dass für alle $x, y, \alpha \in [0, 1]$ gilt: $f(\alpha x + (1 - \alpha)y) \leq \alpha f(x) + (1 - \alpha)f(y)$. Die Funktion sei über ein Orakel gegeben, das zu einem beliebigen Wert $x \in [0, 1]$ den Wert $f(x)$ ausgibt. Außerdem sei ein $\epsilon > 0$ gegeben. Gesucht ist ein $x^* \in [0, 1]$, für das es ein \tilde{x} mit $|x^* - \tilde{x}| < \epsilon$ gibt, so dass für alle $x \in [0, 1]$ gilt: $f(\tilde{x}) \leq f(x)$. Zeigen Sie, dass $O(\lceil \log(\frac{1}{\epsilon} + 1) \rceil)$ Abfragen von Funktionswerten reichen, um ein solches x^* zu berechnen. (6 Punkte)
Hinweis: Modifizieren Sie die binäre Suche geeignet.
- Implementieren Sie den Algorithmus aus Aufgabe 3 des 5. Zettels für zwei Zahlen, die als `LargeInt` gegeben sind. Erweitern Sie dazu die Klasse `LargeInt` insbesondere um
 - einen `--`-Operator zur Subtraktion,
 - eine Methode `is_even()`, die überprüft, ob die Zahl gerade ist,
 - eine Methode `div2()`, die eine Division durch 2 durchführt,
 - eine Methode `is_zero()`, die überprüft, ob die Zahl gleich 0 ist.Benutzen Sie zum Testen und für die Abgabe das umseitig stehende Programm (das auch von der Übungsseite heruntergeladen werden kann). (5 Punkte)

Abgabe: Montag, den 19.11.2018, bis 10:12 Uhr.

Basisprogramm für die Programmieraufgabe:

```
1 // euclid.cpp (Euclidean Algorithm)
2
3 #include <iostream>
4 #include "largeint.h"
5
6 using myint = long long;
7
8 LargeInt gcd2(LargeInt a, LargeInt b)
9 {
10     // Ihre Implementierung
11 }
12
13 int main()
14 {
15     myint a, b;
16     std::cout << "This program computes the greatest common divisor.\n"
17         << "Enter two natural numbers, separated by blank: ";
18     std::cin >> a >> b;
19     std::cout << "gcd: " << gcd2(LargeInt(a), LargeInt(b)).decimal() << "\n";
20 }
```

Abgabe der Programmierübungen:

Per E-Mail an `alma_prog_gr_XX@dm.uni-bonn.de`, wobei `XX` durch die Nummer Ihrer Übungsgruppe zu ersetzen ist, also z.B. `alma_prog_gr_07@dm.uni-bonn.de`, wenn Sie in Gruppe 7 sind, oder `alma_prog_gr_12@dm.uni-bonn.de`, wenn Sie in Gruppe 12 sind. Wenn Sie Ihre Übungsgruppe nicht kennen, schreiben Sie an `alma_prog_gr_unbekannt@dm.uni-bonn.de`.

Öffnungszeiten des Help Desks:

Montags, 16 – 19 Uhr und freitags, 12 – 15 Uhr, jeweils in Raum N1.002, Endenicher Allee 60, Nebengebäude.

www.mathematics.uni-bonn.de/files/bachelor/help-desk.pdf

Zusätzlich gibt es ab sofort einen **Help Desk für Programmierfragen**, und zwar immer freitags, 8 – 10 Uhr und 12 – 16 Uhr, im PC-Pool in der Wegelerstraße 6, Raum E02 (Hochschulrechenzentrum).

Eine Mitteilung der Fachschaft Mathematik: Am 15.11.2018 findet um 19 Uhr (s.t.) im Lipschitzsaal ein Studentinnentreffen (erstes und drittes Fachsemester) statt. Wir möchten mit euch über eure bisherigen Erfahrungen im Mathematikstudium und eure Gründe nach Bonn zu kommen reden. Für ein zahlreiches Erscheinen wären wir sehr dankbar! Für mehr Informationen sendet einfach eine E-Mail an `gleichstellung@fsmath.uni-bonn.de`