

Name: Vorname: Matr.-Nr.:

Technische Universität München
Institut für Informatik I-16
Dr. Tobias Lasser

WiSe 2012/13
10. Dezember 2012

Musterklausur zu Algorithmen und Datenstrukturen

Allgemeine Hinweise

- Die Klausur besteht aus 4 einseitig bedruckten Seiten. Stellen Sie sicher, dass Ihr Exemplar vollständig ist, und **prüfen Sie alle Seiten auf Aufgaben!**
- Sie haben 45 Minuten Zeit für die Bearbeitung der Aufgaben, von 09:50 bis 10:35 Uhr.

Aufgabe 1 Rechnen mit Landau-Symbolen

(4 Punkte)

Untersuchen Sie, ob folgende Behauptung wahr ist und begründen Sie Ihre Aussage.

$$4n^2 + 3n - 5 = O(n^3)$$

Aufgabe 2 Boolesche Logik

(4 Punkte)

Zeigen Sie, dass der logische Ausdruck $a \Rightarrow b$ identisch ist mit $\neg a \vee b$.

Aufgabe 3 Zahldarstellung

(4 Punkte)

Konvertieren Sie die Zahl 7_{10} in eine 4-bit Binärzahl. Berechnen Sie dann das 2-Komplement dieser 4-bit Binärzahl. Als welche zwei Dezimalzahlen lässt sich das entstandene Bitmuster interpretieren?

Aufgabe 4 Sequentielle Liste

(1 Punkt)

Gegeben sei ein Feld A implementiert als sequentielle Liste (Array). Was ist die Komplexität der Einfüge-Operation `insert`?

Aufgabe 5 Stack

(5 Punkte)

Beschreiben Sie kurz eine Methode, um den abstrakten Datentyp Stack mittels elementarer Datenstrukturen zu implementieren. Erläutern Sie dann ausführlich die Implementation der Operation `push`. Was ist die Komplexität dieser Operation?

Aufgabe 6 Fibonacci-Folge, Verifikation

(6 Punkte)

Die Fibonacci-Folge $(f_n)_{n \in \mathbb{N}}$ ist rekursiv definiert als:

$$\begin{aligned} f_1 &= f_2 = 1 \\ f_n &= f_{n-1} + f_{n-2} \end{aligned}$$

Nachfolgend ist die aus der Vorlesung bekannte iterative Implementierung nach dem Prinzip der dynamischen Programmierung angegeben als Funktion **fib**(n). Die Verifikation dieser Implementierung ist Gegenstand dieser Aufgabe.

fib(n) :

```
fib = leeres Feld;
fib[1] = 1;
fib[2] = 1;
k = 3;
while ( $k \leq n$ ) {
    fib[k] = fib[k - 1] + fib[k - 2];
    k = k + 1;
}
return fib[n];
```

a) Geben Sie jeweils eine geeignete Vor- und Nachbedingung der Funktion **fib**(n) an!

b) Geben Sie eine geeignete Schleifeninvariante P für die **while** ($k \leq n$) { ... } Schleife an!

c) Welche drei Bedingungen müssen überprüft werden, damit die Korrektheit der **while** Schleife mittels der Invariante P gezeigt werden kann? Zeigen Sie, dass diese drei Bedingungen in

