

Mathematisch-technischer Softwareentwickler
Mathematisch-technische Softwareentwicklerin
0000

2 Softwareentwurf und Programmierung

Allgemeine Korrekturhinweise

Die Lösungs- und Bewertungshinweise zu den einzelnen Aufgaben sind nicht in jedem Fall Musterlösungen, sondern als Korrekturhilfen zu verstehen. Sie sollen nur den Rahmen der zu erwartenden Prüfungsleistungen abstecken. Der Bewertungsspielraum des Korrektors (z. B. hinsichtlich der Berücksichtigung regionaler, branchen- oder betriebsspezifischer Gegebenheiten) bleibt unberührt.

Zu beachten ist die unterschiedliche Dimension der Aufgabenstellung (nennen – erklären – beschreiben – usw.).

Für die Bewertung gilt folgender Punkte-Noten-Schlüssel:

Note 1 =	100 – 92 Punkte	Note 2 = unter	92 – 81 Punkte
Note 3 = unter	81 – 67 Punkte	Note 4 = unter	67 – 50 Punkte
Note 5 = unter	50 – 30 Punkte	Note 6 = unter	30 – 0 Punkte

1. Aufgabe (60 Punkte) [Korrekte Ergänzung von lückenhaften ISBN-Nummern]

a) [50 P.]

UML-Darstellung

10 P.

Programmierung (40 P.)

13 P.

- Datenstrukturen
- Analyse der Länge der Teile
- Finden und korrigieren der defekten Stelle (17 P.)
 - Normalfälle
 - „X“
 - Prüfziffer ist defekt

10 P.

12 P.

3 P.

2 P.

b) [10 P.]

UML-Darstellung

3 P.

Umsetzung in Programmcode

7 P.

2. Aufgabe (20 Punkte) [Sortieralgorithmen]

a) (4 Punkte)

Heapsort ist zwar nicht rekursiv aber dafür sehr stabil was die Anzahl der Vergleiche und Vertauschungen angeht. Das Verfahren hat immer eine asymptotische Laufzeit. Bubblesort dagegen hat in der Regel quadratischen Aufwand (ohne Optimierung).

b) (10 Punkte)

Die Simulation des Algorithmus der Zahlenfolge gliedert sich in zwei Teile.

1. Bildung des Heap aus der Zahlenfolge
2. Auslesen der Daten aus dem Heap sowie der Neuorganisation des Heaps

Somit ergibt sich die folgende Simulation:

Bemerkung/Index	1	2	3	4	5	6	7
Ausgangsdatensatz	3	7	4	9	11	5	7
Teil							
Nach der Heap Bildung	11	9	7	3	7	5	4
Teil							
1. Durchgang mit reheap	9	7	7	3	4	5	11
2. Durchgang mit reheap	7	5	7	3	4	9	11
3. Durchgang mit reheap	7	5	4	3	7	9	11
4. Durchgang mit reheap	5	3	4	7	7	9	11
5. Durchgang mit reheap	4	3	5	7	7	9	11
6. Durchgang	3	4	5	7	7	9	11

Ein 7. Durchgang ist nicht erforderlich, da die einelementige Restliste immer sortiert ist.

c) (6 Punkte)

Die Laufzeit für verschiedene Eingabefolgen variiert nur geringfügig. Daraus ergibt sich:

$$C_{\min}(N) = C_{\text{average}}(N) = C_{\max}(N) = O(N \cdot \log N)$$

für die Anzahl der Vergleiche und

$$M_{\min}(N) = M_{\text{average}}(N) = M_{\max}(N) = O(N \cdot \log N)$$

für die Anzahl der Vertauschungen.

3. Aufgabe (20 Punkte) [Datenbanken]

T_Kunden	<u>KundenNr</u> Kundenname Ort
T_Niederlassungen	<u>NiederlassungsNr</u> Niederlassung Ort Ansprechpartner
T_Maschinenarten	<u>Typ</u> Maschinenart Führungsnnachweis Herstellername Preis_je_km Mietpreis_pro_Tag
T_Maschinen	<u>MaschinenNr</u> F_Typ KFZKennzeichen
T_Mieten	<u>F_Kundennummer</u> <u>F_Maschinennummer</u> <u>F_Niederlassungsnummer</u> Miete_von Miete_bis Gefahrene_km

- Je Relation in 3. NF 5 Punkte
- Jeweils 1 Punkt Abzug für
 - fehlende Fremdschlüssel bzw. fehlende Markierung als Fremdschlüssel
 - fehlende Primärschlüssel
 - nicht in 3. NF
 - falsche Beziehungen
 - nicht Beachtung des Geschäftsmodells

