
Aufgaben zur Klausur in *Verteilte Systeme (WS 2005 / 2006)*

Zeit: 90 Minuten,
erlaubte Hilfsmittel: keine

Bitte tragen Sie Ihre Antworten und fertigen Lösungen auf gesonderten Blättern ein.

Vergessen Sie nicht, das Deckblatt zu unterschreiben.

Für die Prüfung werden insgesamt 40 Bewertungseinheiten (BE) vergeben. Zum Bestehen benötigen Sie mindestens 20 BE.

Viel Erfolg !

1. Aufgabe (5 BE), Thema: Allgemeine Anforderungen

Erklären Sie vier verschiedene Transparenzforderungen! Was wird allgemein bei verteilten Systemen unter Transparenz für eine bestimmte Eigenschaft verstanden?

2. Aufgabe (4 BE), Thema: Client-Server-Kommunikation

Ein Programmierer beschreibt sein Verfahren zur automatischen Beteiligung als Käufer an einer Internet-Versteigerung so:

- i) Falls die Auktionszeit noch nicht abgelaufen ist und das über dem aktuell höchsten Gebot nächsthöhere zulässige Gebot noch unter der eigenen Höchstgrenze liegt, überbiete das aktuell höchste Gebot.

Anderenfalls steige aus der Versteigerung aus.

- ii) Warte eine Stunde und gehe dann wieder zu i)

Bearbeiten Sie folgende Aufgaben:

- a) Nennen Sie das Prinzip der Client-Server-Kommunikation, das der Programmierer angewendet hat!
- b) Geben Sie ein alternatives Prinzip an und demonstrieren Sie dieses Prinzip durch eine (informelle) Lösung des vorliegenden Problems!

3. Aufgabe (3 BE), Thema: Nebenläufigkeitstechniken

Nennen Sie zwei Risiken bei der Thread-Programmierung und mögliche Vorsorgemaßnahmen dagegen!

4. Aufgabe (10 BE), Thema: Java-RMI

Gegeben seien die folgenden Java-Klassen:

Die Klasse `Storage` speichert beliebige Integerzahlen und beantwortet deren Vorhandensein mit Hilfe des Binärsuchverfahrens. Die Klasse `Tester` dient zum Nachprüfen, ob eine bestimmte Zahl in einer bestimmten Instanz von `Storage` vorhanden ist.

```
public class Tester {  
    public int test (Storage storage, int obj) {return storage.search(obj)}
```

```

public static void main (String args[]) {
    Storage storage = new Storage();
    Tester tester = new Tester();
    System.out.println (tester.test (storage,14));}}

public class Storage {
    public int[] data;

    public int binarySearch (int left, int right, int obj) {
        if (left > right) {return -1;}
        int mid = (left + right) / 2;
        if (k == data[mid]) {return mid;}
        if (k < data[mid]) {return binarySearch (left, mid-1, obj);}
        return binarySearch (mid+1, right, k); }

    public int search (int obj) {return binarySearch (1,data.length,obj)}}

```

Deklarieren Sie `Tester` und `Storage` so, dass eine Instanz von `Storage` auf einem entfernten Server namens "IntServer" läuft. Realisieren Sie dann die hier angegebene Funktionalität mittels Java-RMI durch folgende Teilaufgaben (benötigte Importe müssen nicht angegeben werden):

- Deklarieren Sie das zusätzliche Interface, das Sie benötigen! (3 BE)
- Deklarieren Sie die notwendigen Änderungen in der Server-Klasse (gleich bleibende Methoden und Attribute brauchen Sie nicht zu wiederholen) (3 BE)
- Deklarieren Sie die notwendigen Änderungen in der Client-Klasse (gleich bleibende Methoden und Attribute brauchen Sie nicht zu wiederholen) (4 BE)

5. Aufgabe (3 BE), Thema: Agententechnologie

- Erklären Sie den Unterschied zwischen starker und schwacher Migration von Agenten!
- Wenn Sie als Programmierer ein mobiles Agentensystem einsetzen wollen, welchen Migrationstyp würden Sie bevorzugen? Begründen Sie kurz Ihre Antwort!

6. Aufgabe (3 BE), Thema: Synchronisation von Daten

- Was versteht man unter strenger Konsistenz?
- Warum ist strenge Konsistenz in verteilten Systemen nicht möglich? Geben Sie eine genaue Begründung an!

7. Aufgabe (5 BE), Thema: Synchronisation von Daten

- Erklären Sie den Unterschied zwischen datenzentrierter und clientzentrierter Konsistenz! Welche Forderung ist schwächer?
- Geben Sie ein Beispiel an, das die schwächere Form erfüllt, die stärkere aber nicht!

8. Aufgabe (4 BE), Thema: Fehlertoleranz

Beschreiben Sie den Bully-Algorithmus:

- Geben Sie die Zielsetzung und die Voraussetzung an!
- Beschreiben Sie in Stichworten das Vorgehen!

9. Aufgabe (3 BE), Thema: Web Services

Erklären Sie die jeweilige Hauptfunktionalität von SOAP, WSDL und UDDI!