

1 Verschiedene Fragen

- Skizzieren Sie, wie UI-Elemente beim BorderLayout angeordnet werden können!
- Nennen Sie 3 Kriterien, wie man die Qualität eines Tests bezogen auf den darin abgedeckten Anteil des Codes beurteilen kann!
- Nennen sie den Unterschied zwischen *checked* und *unchecked* Exceptions und geben Sie für jedes Szenario ein kurzes Beispiel an!

2 Model-View-Control/GUI

Gegeben ist ein kleines Programm, welches zu einem Gedicht einzelne Strophen anzeigt. Mit zwei Buttons ist es möglich zwischen den Strophen zu wechseln.

Erläutern Sie kurz die Aufgaben von *Model*, *View* und *Control*! Welche Bestandteile des Programms sind welchem der drei zuzuordnen?

Schreiben Sie ein Programm mit folgender Funktionalität:

- Es gibt ein Fenster mit Titel "Shop Sign", welches einen Button mit Inschrift "Flip Sign" sowie ein Label enthält.
- Das Label hat zu Beginn die Inschrift "OPEN". Bei Betätigung des Buttons soll es von "OPEN" zu "CLOSE" wechseln oder umgekehrt.

3 Eingabe/Ausgabe

Gegeben ist eine Datei, die Daten über Obst enthält in der Form

```
banane reif 5
Apfel verfault 1
```

Gegeben ist zudem die Klasse `Fruit`, welche eine Sorte Obst spezifiziert.

```
public class Fruit {
    String sorte, zustand;
    int anzahl;
    public Fruit(String s, String z, int a) { sorte = s; zustand = z; anzahl = a; }
}
```

Vervollständigen Sie die folgende Methode, die einen Dateipfad übergeben erhält und aus der angegebenen Datei (von der man ausgehen kann, dass sie eine Datei nach obigem Format ist) die einzelnen Obst-Objekte ausliest.

```
public class FruitBasket {
    public List<Fruit> makeFruitBasket(Path file) {
    }
}
```

Nutzen Sie einen `FileVisitor`, um alle Dateien in allen Unterverzeichnissen eines angegebenen Pfades zu durchlaufen und sie nach Dateien zu durchsuchen, welche Obst nach obigem Format beschreiben. Ausgegeben werden soll eine Liste mit allen gefundenen Obstobjekten. Eine Methode `boolean isFruitFile(Path, file)` kann als gegeben verwendet werden.

4 XML

- Was sind die Unterschiede im Verhalten zwischen DOM und SAX?
- Was sind die jeweiligen Vorteile?
- Was bedeutet es, dass eine XML-Datei wohlgeformt ist? Wann ist sie valide?

Gegeben ist folgende DTD

```
<!ELEMENT obstkorb (obst)*>
<!ELEMENT obst (sorte,preis,zustand)>
<!ELEMENT sorte #CDATA>
<!ELEMENT kaufdatum #CDATA>
<!ELEMENT preis #CDATA>
<!ELEMENT zustand EMPTY>
<!ATTLIST zustand qualitaet (‘reif’|‘unreif’|‘ueberlagert’) ‘reif’>
<!ATTLIST preis imangebot (‘ja’|‘nein’) ‘ja’>
```

Geben Sie ein XML-Dokument an, welches den Inhalt der folgenden Tabelle widerspiegelt

| Element | Wert |
|------------|------------|
| Sorte | Banane |
| Kaufdatum | 01.01.1970 |
| Preis | 1.0 |
| Zustand | reif |
| im Angebot | ja |

5 Fork-Join

Implementieren Sie eine parallelisierte Version von MergeSort. Die Funktion `merge` kann hierbei als gegeben betrachtet werden. Gegeben ist das folgende Code-Gerüst.

```
public class MergeSortTask extends
{

}

public class MergeSort {

    public MergeSort(int parallelism) {
    }
    public void sort(int[] array) {
    }
}
```

6 Sockets

Implementieren Sie einen Server, mit dem sich ein Client verbinden kann und der zu einer Eingabe des Client die Anzahl der Wörter zurück gibt. Wörter sind durch Leerzeichen getrennt. Hinweis: Sie müssen nur den Server, nicht den Client implementieren.