

## Prüfung User Interaction (UI)

Prof. Dr.-Ing. Christian Heller <christian.heller@ba-leipzig.de>

Student	
Vor- und Nachname	Ihre Daten werden von der Klausuraufsichtsperson schriftlich auf der Anwesenheitsliste festgehalten.
Matrikelnummer	
Studienrichtung und Jahr	CS 2018-1
Anmeldename	Das Login "klaus..." muss unbedingt schriftlich auf der Anwesenheitsliste festgehalten werden, da sonst keine Zuordnung des Logins zu Ihrem Namen und damit keine Korrektur der Klausur möglich ist!

Prüfung	
Datum	Dezember 2019
Dauer [min]	100
Hilfsmittel	Dokumentation im lokalen Netzwerk (Intranet) sowie Recherche im Internet. NICHT gestattet: * Kommunikation in jeglicher Form * Anmeldung via SSH auf dem Rechner "fileserv" * Anmeldung mit Klausur-Login nach Ende der Prüfung Dies kann leicht geprüft werden (last cs16*, Server-Log-Dateien). Bitte unterlassen Sie also Täuschungsversuche in Ihrem eigenen Interesse.
Bemerkungen	Hinterlegen Sie alle Programme und Antworten in elektronischer Form! Es wird kein Papier angenommen. Möchten Sie Lösungen erläutern, so nutzen Sie Quelltext-Kommentare oder legen eine Text-Datei an. Speichern Sie sämtliche Daten im HOME-Verzeichnis des Nutzers, d. h. unter Windows auf Laufwerk H:\ (NICHT auf C:\ oder "Eigene Dateien")! Idealerweise legen Sie dort ein Unterverzeichnis namens "klausur" an. Lesen Sie die Aufgaben komplett durch, bevor Sie sie lösen! Die Reihenfolge der Lösung ist Ihnen überlassen. Probieren Sie immer, eine Aufgabe zu lösen, da auch auf richtige Teile nicht vollständiger Lösungen Punkte vergeben werden! Falls vom Prinzip her richtig, so werden auch alternative Lösungen akzeptiert. Sie dürfen beliebig viele Bildschirmausgaben von Werten in den Quelltext einbauen, um ein Programm besser nachvollziehen zu können. Bitte duplizieren Sie Ihre Quelltextdateien (workspace) NICHT, da beim Korrigieren dann beide durchsucht werden müssen, was sinnlosen Aufwand verursacht. Diese Aufgabenstellung in Papierform können Sie nach dem Ende der Klausur behalten.

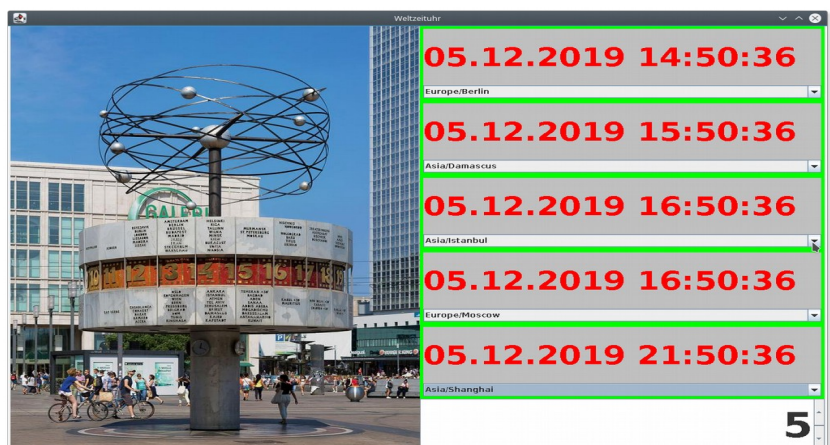
Bewertung						
Aufgabe	1	2	3	4	5	Summe
Punkte	10	20	20	20	30	100

Im Verlauf der Prüfung soll eine Anwendung namens „Weltzeituhr“ gemäß Bildschirmschnappschuss erstellt werden.

Linkerhand ist ein gegebenes Bild eingebunden.

Auf der rechten Seite können null bis maximal fünf Anzeigen mit Datum und Uhrzeit eingeblendet werden. Jede davon hat eine Klappliste mit zehn möglichen Zeitzonen zur Auswahl.

Am unteren Rand kann über einen „Spinner“ die Anzahl der Uhren eingestellt werden.



### Aufgabe 1: Main Frame [10]

Zweck: Bauen eines Fensters.

- a) Erstellen Sie eine Klasse „View“, die als Swing-Rahmenfenster dienen soll! Fügen Sie ihr eine Methode namens „initialise“ hinzu! [2]
- b) Geben Sie dem Fenster den Titel "Weltzeituhr"! Konfigurieren Sie das Fenster so, dass sich die Anwendung beim Schließen beendet! Justieren Sie die Fenstergröße so, dass alle enthaltenen Komponenten optimal dargestellt werden! Schalten Sie das Fenster sichtbar! [2]
- c) Nutzen Sie ein „JLabel“-Objekt, um das gegebene Bild einzubinden (siehe Datei „urania-weltzeituhr\_alexanderplatz\_berlin.jpeg“)! [2]
- d) Fügen Sie es auf der linken Seite des Rahmenfensters ein! [2]
- e) Erstellen Sie eine „Launcher“-Klasse mit „main“-Methode, um das Rahmenfenster zu instanzieren und zu initialisieren! [2]

Ergebnis: Das Hauptfenster der Anwendung wird angezeigt.

### Aufgabe 2: View Content [20]

Zweck: Befüllen des Fensters.

- a) Erzeugen Sie eine Kachel (JPanel)! Referenzieren Sie sie mittels einer lokalen Variablen namens „p“! Weisen Sie ihr ein Layout für Kompassrichtungen zu! [2]
- b) Erzeugen Sie eine weitere Kachel (JPanel)! Speichern Sie sie als Attribut mit dem Namen „clocks“, da sie später die Uhren enthalten soll! Weisen Sie ihr ein „BoxLayout“ für vertikale Darstellung zu! [2]
- c) Erzeugen Sie schließlich ein Zählelement (JSpinner)! Speichern Sie es als Attribut mit dem Namen „counter“! [2]
- d) Übergeben Sie ihm ein passendes „Model“, so dass in Einerschritten von null bis fünf gezählt werden kann! Setzen Sie seinen Anfangswert auf null! [2]
- e) Legen Sie die bevorzugte (preferred) Größe für das Zählelement auf 400x100 fest! [2]
- f) Weisen Sie ihm eine serifenlose Schriftart mit Schriftstil „Fettdruck“ und der Größe 64 zu! [2]
- g) Geben Sie ihm außerdem den Hinweistext „Erzeugt oder löscht eine Uhr“ (der beim Drüberstreichen mit der Maus erscheinen soll)! [2]
- h) Fügen Sie das Zählelement „counter“ zur Kachel „p“ hinzu! Platzieren Sie es unten! [2]
- i) Fügen Sie die Uhrenkachel „clocks“ zur Kachel „p“ hinzu! Lassen Sie sie den verbleibenden Raum einnehmen! [2]
- j) Fügen Sie die Kachel „p“ selbst dem Rahmenfenster hinzu! Platzieren Sie sie rechts des Bildes und lassen Sie sie den verbleibenden Raum einnehmen! [2]

Ergebnis: Im Hauptfenster wird nun auch das Zählelement angezeigt.

### Aufgabe 3: Java Bean [20 + 2]

Zweck: Erstellen einer Uhrenanzeige als grafische Komponente.

- a) Erstellen Sie eine Klasse namens „Display“, welche eine Spezialisierung von „JPanel“ ist! [2]
- b) Lassen Sie sie von einer Schnittstelle erben, die für JavaBeans gefordert ist! [2]
- c) Geben Sie ihr die folgenden vier Attribute: „formatter“ vom Typ „SimpleDateFormat“; „date“ vom Typ „String“; „label“ vom Typ „JLabel“; „timer“ vom Typ „javax.swing.Timer“! [2]

- d) Verhindern Sie ein Serialisieren von „formatter“ und „timer“! [2]
- e) Erzeugen Sie im Standardkonstruktor ein Objekt für das „formatter“-Attribut! Passen Sie sein Format so an, dass es dem Bildschirmschnappschuss entspricht! [2]
- f) Erzeugen Sie ein „JLabel“-Objekt! Setzen Sie seine Schriftfarbe auf rot! [2]
- g) Weisen Sie ihm eine serifenlose Schriftart mit Schriftstil „Fettdruck“ und der Größe 48 zu! Weisen Sie ihm als Aufschrift das Attribut „date“ zu! [2]
- h) Erzeugen Sie eine Klappliste (JComboBox)! Befüllen Sie sie mit den folgenden Werten [2]:  
"Asia/Bangkok", "Asia/Beirut", "Asia/Damascus", "Asia/Istanbul", "Asia/Jakarta",  
"Asia/Shanghai", "Asia/Tokyo", "Europe/Berlin", "Europe/Lisbon", "Europe/Moscow"  
Legen Sie den Standardwert auf "Europe/Berlin" fest!
- i) Legen Sie die bevorzugte (preferred) Größe der „Display“-Komponente selbst auf 600x120 fest! Geben Sie ihr die Hintergrundfarbe hellgrau! Geben Sie ihr einen grünen Rand der Breite 5 Pixel! Weisen Sie ihr ein „BorderLayout“ zu! [2]
- j) Ordnen Sie die Klappliste innerhalb der „Display“-Komponente unten an! Lassen Sie das „label“-Attribut-Objekt den verbleibenden Raum einnehmen! [2]
- k) Zusatzaufgabe: Fügen Sie weitere Zeitzonen ein! [2]
- Hinweis: Sie können dafür die gegebene Datei „java-timezones.txt“ verwenden.
- Ergebnis: Die Uhrenanzeige kann verwendet werden.

#### **Aufgabe 4: Spinner Controller [20]**

Zweck: Verarbeiten der Ereignisse des Zählelements (JSpinner).

- a) Erstellen Sie eine Klasse namens „Controller“, die als „ChangeListener“ dienen soll! Geben Sie ihr ein Attribut vom Typ „View“, dessen Wert via „initialise“-Methode zugewiesen wird! [2]
- b) Bestimmen Sie in der Ereignisverarbeitungsmethode zunächst den aktuellen Wert des „counter“-Zählelements der View und speichern Sie ihn in einer lokalen Variablen! [2]
- c) Speichern Sie dann die aktuelle Anzahl an Komponenten in der „clocks“-Uhrenkachel der View in einer weiteren lokalen Variablen! [2]
- d) Berechnen Sie die Differenz aus beiden und hinterlegen Sie sie in einer dritten Variablen! [2]
- e) Führen Sie eine Fallunterscheidung durch für Differenz größer und kleiner Null! (Hinweis: Im Falle der Gleichheit soll nichts unternommen werden.) [2]
- f) Nutzen Sie im einen Fall eine Zählschleife, um entsprechend viele „Display“-Objekte neu zu erstellen und der „clocks“-Uhrenkachel der View hinzuzufügen! [2]
- g) Nutzen Sie im anderen Fall ebenfalls eine Zählschleife, um die geforderte Anzahl vorhandener „Display“-Objekte aus der „clocks“-Uhrenkachel der View zu löschen (remove)! [2]
- h) Justieren Sie das Schleifenlaufkriterium wie auch den Index für die „remove“-Methode sorgfältig! [2]
- Hinweis: Für ersteres könnte eventuell die Bestimmung des Betrages der Differenz hilfreich sein. Aber verschiedene Lösungen sind möglich.
- i) Justieren Sie die Größe der „View“ neu! Veranlassen Sie ein Neuzeichnen der „View“! [2]
- j) Erzeugen und initialisieren Sie in der „main“-Methode ein „Controller“-Objekt! Übergeben Sie es an die „initialise“-Methode der „View“! Registrieren Sie es dort am „counter“-Zählelement als Ereignisverarbeiter! [2]

Ergebnis: Uhren können hinzugefügt oder entfernt werden.

## Aufgabe 5: Bean Action Listener [30]

Zweck: Wechseln der Zeitzone in der Uhrenanzeige.

a) Erzeugen Sie im Standardkonstruktor der „Display“-Klasse das Objekt für das „timer“-Attribut! Stellen Sie eine Verzögerung von 1 s ein! Registrieren Sie die Klasse selbst als Ereignisverarbeiter! [2]

b) Registrieren Sie die Klasse selbst auch als Ereignisverarbeiter für die Klappliste! [2]

c) Starten Sie den „Timer“! [2]

d) Implementieren Sie Kapselungs- beziehungsweise Zugriffsmethoden für zwei Eigenschaften (properties) namens „timeZone“ vom Typ „String“ und „date“ vom Typ „Date“! [2]

Hinweis: Bedenken Sie, dass Eigenschaften einer JavaBean nicht zwingend als Attribut existieren müssen, sondern ihre Werte auch auf andere Weise gespeichert werden können!

e) Bestimmen Sie in der „get“-Methode zur Zeitzone via „formatter“-Attribut die aktuelle Zeitzone und darüber ihren Langnamen als Zeichenkette! [2]

f) Bestimmen Sie in der „set“-Methode zur Zeitzone mittels statischer Methode der Klasse „TimeZone“ die Zeitzone, wofür die übergebene Zeichenkette zu verwenden ist! Weisen Sie sie dem „formatter“-Attribut zu! [2]

g) Erzeugen Sie in der „get“-Methode zum Datum via statischer Methode der Klasse „DateFormat“ zunächst eine Instanz! Nutzen Sie dann deren „parse“-Methode! Übergeben Sie ihr den aktuellen Wert des „date“-Attributes der „Display“-Klasse! Verwenden Sie das somit bestimmte Datum als Rückgabewert! [2]

h) Rufen Sie in der „set“-Methode zum Datum am „formatter“-Attribut eine passende Instanzmethode auf, die das „Date“-Argument entgegennimmt und eine Zeichenkette zurückliefert! Speichern Sie die Zeichenkette im „date“-Attribut! Weisen Sie sie außerdem dem „label“-Attribut als Aufschrift zu! [2]

i) Speichern Sie in der Ereignisverarbeitungsmethode die Ereignisquelle in einer lokalen Variablen! [2]

j) Finden Sie per Fallunterscheidung heraus, ob es sich um ein Ereignis von „Timer“ oder „JComboBox“ handelt! [2]

Hinweis: Operator „instanceof“

k) Geben Sie mittels „JOptionPane“ eine Fehlernachricht aus, wenn keiner der beiden Fälle zutrifft! [2]

l) Handelt es sich um ein Zeitgeberereignis (Timer), so erstellen Sie ein neues Datumsobjekt (zur Bestimmung der aktuellen Zeit) und rufen die weiter oben implementierte „set“-Zugriffsmethode zum Datum auf! [2]

m) Handelt es sich um ein Klapplistenereignis, so führen Sie zunächst eine passende Typumwandlung durch! [2]

n) Bestimmen Sie anschließend den aktuell ausgewählten Eintrag als Zeichenkette! [2]

o) Rufen Sie die weiter oben implementierte „set“-Zugriffsmethode zur Zeitzone auf! Übergeben Sie die Zeichenkette des Klapplisteneintrages als Argument! [2]

Ergebnis: Das Datum der Uhrenanzeige passt sich an die neu gewählte Zeitzone an.

**Viel Erfolg!**