

**Übungsblatt 5 WT:IV**

Bis zum 30.06.2021, 23:59 sind Lösungen zu folgenden Aufgaben abzugeben: 1 2, 3a,b,c,f, 4, 5b,c, 6, 7a,b.

Für die Programmieraufgaben dieses Übungsblattes benötigen Sie einen Web-Server zur Ausführung von PHP-Skripten sowie einen Servlet-Container zur Ausführung von Servlets und JSP-Skripten. Wir empfehlen die Nutzung des [Apache HTTP Server](#) sowie von [Apache Tomcat](#). Unter Windows bietet sich die Nutzung der Distribution [XAMPP](#) an.

**Aufgabe 1 : CGI, Servlets, PHP (2+1+1+1 Punkte)**

- (a) 100 Benutzer greifen gleichzeitig auf ein CGI-Programm, das als Shell-Skript realisiert ist, und auf ein Servlet zu. Beschreiben Sie, wie der Web-Server jeweils mit den gleichzeitigen Anfragen umgeht und stellen Sie die Unterschiede der Verarbeitung heraus.
- (b) Wie kann Session-Tracking in einem CGI-Programm implementiert werden? Nennen Sie zuerst die dafür nötigen Funktionen, Objekte, Variablen, etc. und beschreiben Sie dann zweitens knapp, was dabei im Web-Server geschieht. Beziehen Sie sich bei Ihrer Antwort auf eine einzelne Sprache für CGI-Programme.
- (c) Wie kann Session-Tracking in einem Servlet implementiert werden? Nennen Sie zuerst die dafür nötigen Funktionen, Objekte, Variablen, etc. und beschreiben Sie dann zweitens knapp, was dabei im Web-Server geschieht.
- (d) Entwerfen Sie ein PHP Programm, das Session-Tracking realisiert. Das Programm soll dazu manuell einen Cookie setzen und (ebenfalls manuell) wieder auslesen und ausgeben. "Manuell" heißt, dass auf die Verwendung von `set_cookie()`, `$_COOKIE`, und `session_start()` verzichtet werden soll. Stattdessen sollen `header()` und `getallheaders()` verwendet werden.

**Aufgabe 2 : JSP (3 Punkte)**

- (a) Entwerfen Sie ein JSP-Skript, das den Inhalt eines Ordners als HTML-Dokument anzeigt wenn zuvor das richtige Passwort eingegeben wurde.
- (b) Beschreiben Sie aus Sicht der Servlet-Code-Generierung die Unterschiede zwischen den JSP-Scriptlet-Auszeichnungen "`<%`", "`<%=`" und "`<%!`".

**Aufgabe 3 : Multiple Choice (1+1+1+0+0+1 Punkte)**

*Hinweis:* Für die Beantwortung von Teilaufgabe (a) gibt es einen Bonuspunkt.  
Kreuzen Sie Zutreffendes an:

- (a) Folgende Programme und Dokumenttypen können unmittelbar als CGI-Programm verwendet werden:
  - ☐ Jede Java-Servlet-Klasse
  - ☐ Jedes in einer Shell ausführbare Programm
  - ☐ Jedes Shell-Skript
  - ☐ Jedes JSP-Dokument

- (b) Bei jeder HTTP-GET-Anfrage an ein Servlet wird dieses zunächst in Maschinencode der Zielpattform übersetzt.
- ☐ Richtig
  - ☐ Falsch
  - ☐ Es wird jedes Mal Bytecode erzeugt
- (c) Folgende Aussagen treffen auf einen Apache-2-Server zu:
- ☐ Er läuft standardmäßig auf einem dynamischen Port
  - ☐ Er läuft standardmäßig auf Port 80
  - ☐ Er läuft standardmäßig auf Port 443
  - ☐ Er kann XHTML-Seiten ausliefern
  - ☐ Er ist ein Servlet-Container
  - ☐ Er kommuniziert auf Bitübertragungsschicht
- (d) Jakarta HTTPServlets kommunizieren
- ☐ auf ISO/OSI-Schicht 2
  - ☐ auf ISO/OSI-Schicht 3
  - ☐ auf ISO/OSI-Schicht 4
  - ☐ auf ISO/OSI-Schicht 7
- (e) Jakarta Servlets kommunizieren im Allgemeinen
- ☐ auf ISO/OSI-Schicht 2
  - ☐ auf ISO/OSI-Schicht 3
  - ☐ auf ISO/OSI-Schicht 4
  - ☐ auf ISO/OSI-Schicht 7
- (f) Welche der folgenden Aussagen aus dem Gebiet der formalen Sprachen sind wahr?
- ☐ Es gibt kontextfreie Sprachen die sich nicht mit einem regulären Ausdruck definieren lassen.
  - ☐ Die Beschreibungsmächtigkeit endlicher Automaten ist äquivalent zu der von regulären Ausdrücken.
  - ☐ Die Beschreibungsmächtigkeit von Kellerautomaten ist äquivalent zu der von regulären Ausdrücken.

Beachten Sie, dass zu einer Frage mehrere Antworten zutreffen können. Eine Frage gilt als richtig beantwortet, falls alle zutreffenden und keine unzutreffende Antwort angekreuzt ist.

#### Aufgabe 4 : Grammatiken und reguläre Ausdrücke (1+1+1+1 Punkte)

- (a) Was versteht man unter dem Wortproblem für eine reguläre Sprache?
- (b) Gegeben sei eine Grammatik  $G = (\{S, A, B, C\}, \{0, 1\}, P, S)$  mit

$$P = \{S \rightarrow 01A, A \rightarrow 00B, B \rightarrow 1A, B \rightarrow 0C, C \rightarrow 00\}$$

Geben Sie einen regulären Ausdruck an, der die zu  $G$  gehörende Sprache spezifiziert.

- (c) Gegeben sei die Sprache  $L = \{w | w \in \{0, 1\}^* \text{ mit } w = 1^m 0^n 1, m \geq 0, n = 3m\}$ . Geben Sie eine kontextfreie Grammatik  $G$  an, die  $L$  erzeugt.

- (d) Wenn die Produktionen einer Grammatik der Form  $A \rightarrow wB$  oder  $A \rightarrow w$  sind, wobei  $A, B$  Nichtterminale und  $w$  eine (möglicherweise leere) Zeichenkette von Terminalen ist, so nennen wir die Grammatik rechtslinear.
- Geben Sie für die von  $R = (011)^*00(011)^*$  spezifizierte Sprache eine rechtslineare Grammatik an.

#### Aufgabe 5 : Web-Server (0+1+1 Punkte)

- (a) In der Vorlesung wurde der Apache Webserver vorgestellt. Nennen sie mindestens zwei weitere Webserver und erläutern sie kurz, wie sich diese vom Apache unterscheiden. In welchen Anwendungsfällen würden sie die Alternativen dem Apache vorziehen?
- (b) Informieren sie sich über FastCGI und nennen sie den wichtigsten Unterschied gegenüber CGI. Erklären sie die Vorteile von FastCGI, die sich daraus ergeben.
- (c) Erklären sie kurz die folgenden Konzepte, um Weiterleitungen zu realisieren: HTTP-Redirects, Proxy Server und das HTML Tag `<meta http-equiv="refresh">`. Wann würden sie welche Methode bevorzugen?

#### Aufgabe 6 : Reguläre Ausdrücke mit PHP (3+2 Punkte)

- (a) Erstellen Sie Funktionen in PHP zur Überprüfung, ob eine Eingabe die folgende Form hat (Verwenden Sie dazu reguläre Ausdrücke):
- (a1) eine IPv4-Adresse (Stellen Sie sicher, dass Sie nur Blöcke im Bereich 0–255 zulassen)
  - (a2) eine E-Mail-Adresse.
  - (a3) eine URL nach [RFC 3986](#) (Beschränken Sie sich auf die URL-Bestandteile Scheme, Authority und Path)

Hinweis: In den obigen Beispielen können die korrekten Ausdrücke mitunter sehr lang und schwer verständlich werden. Wenn dies der Fall ist, beschränken Sie sich auf eine Lösung, die möglichst präzise ist, deren regulärer Ausdruck aber auch in maximal 3 Zeilen Code passt.

- (b) Informieren Sie sich über sogenannte “Capturing Groups”, “Backreferences” (Rückwärtsreferenzen), und “Lookahead/Lookbehind” in regulären Ausdrücken.
- (b1) Erklären Sie in diesem Zusammenhang den Unterschied zwischen den Ausdrücken `/(aa)?bb/` und `/(?:aa)?bb/`.
  - (b2) Erklären Sie weiterhin den Unterschied zwischen den drei Ausdrücken:  
`/hello(=? world)/`  
`/hello( world)/`  
`/hello world/`  
 Geben Sie an, welcher Gesamtausdruck zurückgegeben wird, sowie (falls vorhanden) den Inhalt aller Capture-Groups.

#### Aufgabe 7 : Programmiermodul: HTTP-API P (3+2+0 Punkte)

Schreiben Sie ein Servlet, dass HTTP Lese- und Schreib-Anfragen für den FeedReader implementiert.

Arbeiten Sie für diese Aufgabe in der Ordnerstruktur des `feed-reader-http-api-scaffold`-Ordners im Repository zur Übung. Die mitgelieferte `readme.txt` enthält Hinweise zum Deployment und zur Einbringung Ihrer Dateien aus den vorangegangenen Übungen.

- (a) Schreiben Sie ein Servlet, dass den Feed unter [localhost:8080/feed-reader/Feed](http://localhost:8080/feed-reader/Feed) zur Verfügung stellt. Schreiben Sie eine entsprechende `WEB-INF/web.xml`.
- (b) Implementieren Sie das Hinzufügen eines neuen Eintrags im Servlet. Die Datei `add-entry.html` (zu erreichen unter [localhost:8080/feed-reader/add-entry.html](http://localhost:8080/feed-reader/add-entry.html)) enthält das Formular, dass die Daten mittels POST an das Servlet sendet. Fügen Sie den Eintrag mit der entsprechenden Methode aus der letzten Übung dem Feed hinzu. Nutzen Sie eine Response mit See Other (303) Statuscode und Location-Header um den Nutzer nach dem Hinzufügen auf die Ausgabe des Feeds aus (b) weiterzuleiten.
- (c) Erweitern Sie die bereits implementierten Methoden, so dass neben eines herkömmlichen GET-Requests auch ein Conditional GET-Request auf Basis des letzten Änderungsdatums des Feeds benutzt werden kann.

## Hinweise zur Abgabe

- Erstellen Sie *eine* PDF-Datei, in der Texte und Grafiken zu den Aufgaben enthalten sind.
- Quellcode (Java, JavaScript, PHP, HTML, CSS, XML, XSL, XSD, etc.) innerhalb des PDF-Dokuments wird nicht korrigiert. Quellcode jeder Aufgabe soll, wie im Tutorium vorgestellt, über GitLab verwaltet werden.
- Referenzieren Sie Ihre Quellcode-Dateien in dem PDF-Dokument, so dass sie einer Aufgabe eindeutig zuzuordnen sind. Verlinken sie sie entsprechendem Repositories.
- Abzugeben ist *eine* PDF-Datei die wie folgt benannt ist:  
`<Nachname>-<MatrikelNr>-webtec-blatt<Übungsblattnummer>.pdf`.
- Gruppenabgaben mit bis zu drei Personen sind erlaubt; pro Gruppe genügt dann eine Abgabe, in der alle Gruppenmitglieder mit Namen, Matrikel, und E-Mail-Adresse verzeichnet sind
- Legen sie die PDF-Datei in das GitLab Repository, in dem auch der Code ihrer Gruppe verwaltet wird.