

1. Schulaufgabe						
Prüfungsfach:	<b>Prog.</b>	Teil:		Name:		
Prüfungstag:	<b>06.12.17</b>	Zeit:	<b>60</b>	Minuten	Klasse:	<b>I3A</b>
Hilfsmittel:	<b>keine</b>				Gruppe:	
Umfang der Aufgabe	<b>1</b>	Blätter	Aufgabenblatt =Lösungsblatt	<b>nein</b>	Studien- Halbjahr	<b>WH 17/18</b>
Aufgabensteller	<b>Dipl.-Ing. Andreas Maier</b>			erreichte Punktzahl:		
<hr/> Unterschrift des Lehrers				Note:		

Viel Erfolg!

**Gesamtpunkte: 60**

In dieser Schulaufgabe geht es um die *Blockchain*-Technologie, wie sie beispielsweise bei der Kryptowährung *Bitcoin* eingesetzt wird. Der Einfachheit halber wird in dieser Klausur ein stark vereinfachtes und nicht vollständig korrektes Modell verwendet.

Gegeben ist eine *SQLite*-Datenbank *Blockchain.db*. In dieser Datenbank sind Blöcke persistent und nicht in der korrekten Reihenfolge gespeichert. Ein Block besteht dabei aus einer *id*, einem Zeitstempel (Erstellungszeit in Sekunden), den eigentlichen Daten (eine Zeichenkette), dem *SHA-256*-Hashwert des vorherigen Blocks sowie dem *SHA-256*-Hashwert aller eigenen Daten:

```
sqlite> select id, timestamp, data, prev_hash, self_hash from block;
13|1512457369.0|Start:|0|79AE223599FC7872E91C466A058EE6972965FD422A8142742075A77B7ABFF9DD
18|1512457669.0|Frohe!|79AE223599FC7872E91C466A058EE6972965FD422A8142742075A77B7ABFF9DD|300B15CD48AA40AE16489F83628C25A72FFC0B651B08A638ADB42A16923B470
114|1512457433.0|Jahr!|9F1F9DCE319C4700EF28EC8C5BD3CC8E6ABE64C68385479AB89215806A5BDD6|BF0798663DBDF6FE98ACB8E1148651A1E65118A428EBD10EF6AA0D48CB97C718
113|1512457372.0|neues!|6C658EE83FB7E812482494F3E416A876F63F418A088A1F5E76D47EE4177035CB|34AFFB814A2ACF53B1D92D5E7343544A233A508DE99346D67086869F7F7898A88
2|1512457499.0|||53CC0C57E6825607CD582635C44714E25313C381BC5D023ABC9582AEA0E1400D|DCFC3DD5583B64FC3AAC81D3EF6EB7F49189BDA712C07EC7B658D5C77DE2B2
85|1512457551.0|den!|00E07AE08BCA679186562C361C88566F7AC85680402780C45C25225847AB8549|46984817908F82E7B87188307A5A3A538D6A30F8F0521A2289D10C688E36F0AC
11|1512457662.0|ein!|4FC82B26AECB47D2868C4EFBE3581732A3E7CBCC6C2EFB32062C08170A05EEB8|114BC73437E948B3CE8E9FBF6047019D051AE6C48993A02B2AFC01B5791A20BF
85|1512457525.0|bei!|08315846DFB2F12A4500829A71DA834D6281D5A6A3EB2188B1A98D1A8AC58B|00E07AE08BCA679186562C361C88566F7AC85680402780C45C25225847AB8549
12|1512457814.0|Frohes!|6B51D431DF507F141CBCECCF79EDF3DD861C3B4069F0811661A3EEFACB8A918|B8914D226A8C311A5E0DCF818E02DD070D228DF929AFD3DA42F05D0670CAE3
4|1512457613.0|Weihnachten!|300B15CD48AA40AE16489F83628C25A72FFC0B651B08A638ADB42A16923B470|53CC0C57E6825607CD582635C44714E25313C381BC5D023ABC9582AEA0E1400D
23|1512457921.0|||B117073F0DE4911AF06A1D8B105CA16509105C08CC0DBA1F0CF72E56098C540|96108E262C405F6CF7FE2E649E84894DDC2B2358505E03DF1ADA0DEE35E0BC5
76|1512457272.0|Erfolg!|E40029F96A8550CC4F5198A2E250A49CE902AB2090771E59EA303DDE033F207F|08315846DFB2F12A4500829A71DA834D6281D5A6A3EB2188B1A98D1A8AC58B
85|1512457322.0|Prüfungen!|46984817908F82E7B87188307A5A3A538D6A30F8F0521A2289D10C688E36F0AC|B117073F0DE4911AF06A1D8B105CA16509105C08CC0DBA1F0CF72E56098C540
56|1512457236.0|Und!|7688B6FE52555962D008FF894223582C484517CEA7DA49EE67800ADC7FC8866|CD3DABAD6E047881E12E6CD28BD8CD6F0E4735E82F5905A3B9E6E08E73A0EBFB
42|1512457109.0|Viel!|DCFC3DD5583B64FC3AAC81D3EF6EB7F49189BDA712C07EC7B658D5C77DE2B2|E40029F96A8550CC4F5198A2E250A49CE902AB2090771E59EA303DDE033F207F
```

Das Ziel dieser Schulaufgabe ist es, alle Daten aus der Datenbank zu lesen, Java-Objekte zu erzeugen und diese in der korrekten Reihenfolge in einer Liste abzulegen. Die korrekte Reihenfolge der Blöcke ergibt sich aus der Tatsache, dass jeder Block den *SHA-256*-Hashwert seines direkten Vorgängers gespeichert hat. Nicht alle in der Datenbank vorhandenen Datensätze haben jedoch einen gültigen Vorgänger. Werden die Daten aller Blöcke in der korrekten Reihenfolge ausgegeben, ergibt sich ein Lösungstext. Der erste Block dieses Lösungstextes hat als Vorgänger-Hash die Zeichenkette "0". Dieser Lösungstext soll auf der Konsole ausgegeben werden. Zur Berechnung bzw. Überprüfung der Hashwerte steht Ihnen die Klasse *Util* mit einer Klassenmethode `public static String hashFunction(String toHash)` zur Verfügung.

**1. Aufgabe (15 Punkte)**

Erstellen Sie die Klasse *Block*. Beim Erstellen von Objekten sollen alle Werte als Konstruktordparameter übergeben werden, nur der Hash-Wert für das eigene Objekt soll im Konstruktor errechnet werden. Erstellen Sie weiterhin auch Getter- und Setter-Methoden.

**2. Aufgabe (15 Punkte)**

Erstellen Sie die Klasse *BlockService*. Ein Objekt dieser Klasse soll eine Methode zum Lesen aller Daten aus der Datenbank zur Verfügung stellen.

**3. Aufgabe (30 Punkte)**

Erstellen Sie die Klasse *GBSBlockchain*. Diese Klasse soll eine Liste mit Blöcken halten. Die Blöcke in dieser Liste sollen in korrekter Reihenfolge liegen (aktueller Block besitzt Hashwert des vorherigen Blocks). Beim Erzeugen eines Objektes dieser Klasse sollen alle Daten aus der Datenbank gelesen und in die Liste überführt werden. Schreiben Sie weiterhin eine Methode, welche die Daten aller Blöcke der Liste (Lösungstext) nacheinander auf der Konsole ausgibt.