

Minitest 5

Minh Khue Pham (2579036)

29.06.2021 13:50:15 - 14:08:41

4/6 Multiple Choice (6 Punkte)

Beantworten Sie die folgenden Fragen mit **wahr** oder **falsch**. Für jede richtige Antwort bekommen Sie 1 Punkt. Für falsche Antworten bekommen Sie 0 Punkte. Wenn Sie keine Antwort geben bekommen Sie 0.5 Punkte.

English instruction: Answer the following questions with **true** or **false**. For every correct answer you obtain 1 point. For incorrect answers you get 0 points. If you do not give an answer you receive 0.5 points.

1. Jede Implementierung des Interfaces `Set` muss die Methode `get(int i)` zur Verfügung stellen, um das i-te Element der Menge zurück zu geben.

1. Every implementation of the interface `Set` must provide the method `get(int i)`, which retrieves the i-th element of the set.

Response:

+1

- ☐ wahr / true
- ☒ falsch / false

2. In einer Instanz von `List` kann jedes Element nur einmal enthalten sein.

2. In an instance of `List`, every element can occur at most once.

Response:

+1

- ☐ wahr / true
- ☒ falsch / false

3. Kann ein Iterator keine weiteren Elemente liefern, so liefert `hasNext()` `true`.

3. If an iterator can not supply further elements, the method `hasNext()` returns `true`.

Response:

+1

- ☐ wahr / true
- ☒ falsch / false

4. Der Lastfaktor einer Hashtabelle berechnet sich dadurch, dass man die Anzahl der eingefügten Objekte durch die Größe der Hashtabelle teilt.

4. The load factor ("Lastfaktor") of a hash table is defined as the ratio of inserted objects to size of the hash table.

Response:

+1

- ☒ wahr / true
- ☐ falsch / false

5. Um beim quadratischen Sondieren alle freien Plätze zu durchlaufen muss darauf geachtet werden, dass die Hashfunktion surjektiv ist.

5. To traverse all free spaces of a hash table when using quadratic probing ("quadratisches Sondieren"), it is important for the hash function to be surjective.

Response:

f

- ☐ wahr / true
- ☒ falsch / false

6. Die Methode `int hashCode() return 42;` erfüllt den Vertrag zwischen `equals` und `hashCode`.

6. The method `int hashCode() return 42;` fulfills the contract between `equals` and `hashCode`.

Response:

f

- ☐ wahr / true
- ☒ falsch / false

[[unknown type: ``endAttemptInteraction""]]

3/3 Listen und Mengen (3 Punkte)

Welche Ausgabe liefern die folgenden Programme? (1.5 Punkte pro Ausgabe)

English instruction: What is the output of these programs? (1.5 points per output)

Programm 1:

```
Set<Integer> s = new HashSet<Integer>();  
s.add(42);  
s.add(0);  
s.add(0);  
s.add(1);  
s.remove(0);  
System.out.println(s.size());
```

Programm 2:

```
List<Integer> l = new LinkedList<Integer>();
l.add(5);
l.add(6);
l.add(7);
l.add(6);
l.remove(6);
System.out.println(l.contains(6));
```

Antwort Programm 1 / Answer program 1 :

Antwort Programm 2 / Answer program 2:

[[unknown type: ``endAttemptInteraction'']]

11/11 Hashing (11 Punkte)

Im Folgenden sollen Sie mehrere Elemente in eine Hashtabelle der Größe $m = 7$ einfügen, wobei verschiedene Hashing-Varianten zum Einsatz kommen. Es seien folgende Funktionen gegeben:

English instruction: In the following exercise, you must insert several elements into a hash table of size $m = 7$, using various hashing strategies. Let the following functions be given:

$$h(x) := (x - 3) \% m \quad h'(x, i) := h(x) + i$$

1. Fügen Sie die Elemente 8, 2, 0, 7, 16, 21 in dieser Reihenfolge in die unten dargestellte Hashtabelle ein. Benutzen Sie h als Hashfunktion und lösen Sie Konflikte durch *Kollisionlisten*. Skizzieren Sie nur den Endzustand nachdem alle Elemente eingefügt wurden. (4.5 Punkte)

English instruction: Insert the elements 8, 2, 0, 7, 16, 21 in this order into the hash table depicted below. Use h as the hash function and resolve conflicts using *collision lists*. Sketch only the final state after inserting all elements. (4.5 points)

4.5 Hashtabelle/Hash table:

				0,7,21	8	2, 16
--	--	--	--	--------	---	-------

2. Fügen Sie die Elemente 1, 9, 41, 2, 20 in dieser Reihenfolge in die unten stehende Hashtabelle ein. Lösen Sie Konflikte durch *lineares Sondieren*, das heißt mit der Hashfunktion h' . Skizzieren Sie nur den Endzustand nachdem alle Elemente eingefügt wurden. (5 Punkte)

English instruction: Insert the elements 1, 9, 41, 2, 20 in this order into the hash table depicted below. Resolve conflicts by using *linear probing* this time, i.e. using h' as the hash function. Sketch only the final state after inserting all elements. (5 points)

5 Hashtabelle/Hash table:

2			41	20	1	9
---	--	--	----	----	---	---

3. Beim linearen Sondieren kann es zu Clusterbildung kommen, wenn oft Elemente mit dem selben Hashwert eingefügt werden. Nennen Sie eine Sondierungsmethode aus der Vorlesung, welche die Clusterbildung vermindert. (1.5 Punkte)

English instruction: When using linear probing, clusters can form when inserting elements with the same hash value often. State a probing technique from the lecture which mitigates the issue of clustering. (1.5 points)

1.5

[[unknown type: ``endAttemptInteraction'']]

Unknown item: ``endAttemptInteraction"
Unknown item: ``endAttemptInteraction"
Unknown item: ``endAttemptInteraction"