

I. Erläuterungen

Aufgabenart

materialgebundene Aufgabenstellung

Voraussetzungen gemäß Lehrplan und Erlass „Hinweise zur Vorbereitung auf die schriftlichen Abiturprüfungen im Landesabitur 2016“ vom 20. Juni 2014

Q1 objektorientierte Modellierung

Objektmodell, Klassen, iterative Verfahren, Standardalgorithmen

II. Lösungshinweise und Bewertungsraster

In den nachfolgenden Lösungshinweisen sind alle wesentlichen Gesichtspunkte, die bei der Bearbeitung der einzelnen Aufgaben zu berücksichtigen sind, konkret genannt und diejenigen Lösungswege aufgezeigt, welche die Prüflinge erfahrungsgemäß einschlagen werden. Selbstverständlich sind jedoch Lösungswege, die von den vorgegebenen abweichen, aber als gleichwertig betrachtet werden können, ebenso zu akzeptieren.

Aufg.	erwartete Leistungen	BE			
		I	II	III	Σ
1	Autogrid ▢ Anzahl: int ▢ Elektroautos: Elektroauto[] ▢ Gesamtladezustand: int ⊕ Autogrid() ⊕ berechneGesamtladezustand(): int ⊕ holeEnergie(Betrag: int) ⊕ ausgeben() ⊕ bestimmeMaxIndex(): int ⊕ bestimmeRelativenMaxIndex(): int Elektroauto ▢ Kennzeichen: String ▢ Halter: String ▢ Ladezustand: int ▢ Ausgegeben: boolean ⊕ Elektroauto(Kennzeichen: String, Halter: String, Ladezustand: int) ⊕ getKennzeichen(): String ⊕ getHalter(): String ⊕ getLadezustand(): int ⊕ aendereLadezustand(Betrag: int) ⊕ getAusgegeben(): boolean ⊕ setAusgegeben(Ausgegeben: boolean) Da ein Autogrid aus mehreren Elektroautos besteht, ist die Beziehung zwischen den Klassen <i>Autogrid</i> und <i>Elektroauto</i> eine Aggregation.	6	10		16
2	<i>bestimmeMaxIndex()</i>: Zunächst wird die int-Variable <i>MaxIndex</i> mit dem Wert 0 belegt. In der folgenden von 1 bis <i>Anzahl-1</i> laufenden Zählschleife wird in einer bedingten Anweisung geprüft, ob das Elektroauto mit dem Index <i>i</i> einen höheren Ladezustand hat als das Elektroauto, dessen Index gerade in <i>MaxIndex</i> gespeichert ist. Ist dies der Fall, so wird der Index <i>i</i> als neuer <i>MaxIndex</i> gespeichert. Schließlich wird der Wert von <i>MaxIndex</i> zurückgegeben. Die Methode <i>bestimmeMaxIndex()</i> liefert den Index eines Elektroautos mit höchstem Ladezustand. <i>bestimmeRelativenMaxIndex()</i>: Zunächst werden die Variablen <i>Index</i> und <i>Maximum</i> mit -1 initialisiert. Die folgende Zählschleife durchläuft alle Elektroautos des Grids. In jedem Durchlauf wird in einer bedingten Anweisung geprüft, ob der Ladezustand des Elektroautos mit dem Index <i>i</i> größer ist als der Wert, der momentan in <i>Maximum</i> gespeichert ist und ob im Attribut <i>Ausgegeben</i> dieses Elektroautos der Wert <i>false</i> gespeichert ist. Ist das der Fall, so werden der momentane Wert von <i>i</i> in <i>Index</i> und der Ladezustand des Elektroautos in <i>Maximum</i> gespeichert. Nach dem Beenden				

Aufg.	erwartete Leistungen	BE								
		I	II	III	Σ					
	der Schleife wird das Attribut <i>Ausgegeben</i> des Elektroautos an der Stelle <i>Index</i> auf <i>true</i> gesetzt und der Wert von <i>Index</i> zurückgeliefert. Die Methode <i>bestimmeRelativenMaxIndex()</i> liefert den Index eines Elektroautos mit höchstem Ladezustand aller noch nicht ausgegebener Elektroautos.	6	6	3	15					
3	<pre>public void holeEnergie(int Betrag) { int MaxIndex; while (Betrag > 0) { MaxIndex = bestimmeMaxIndex(); if (Elektroautos[MaxIndex].getLadezustand() <= 50) { Betrag = 0; } else { Elektroautos[MaxIndex].aendereLadezustand(-1); Betrag--; } } }</pre>	6	6	5	17					
4	Algorithmus Ausgeben <table border="1"><tr><td>wiederhole für i=0 bis Anzahl-1</td></tr><tr><td> Ausgegeben von Elektroautos[i] auf false setzen</td></tr><tr><td>wiederhole für i=0 bis Anzahl-1</td></tr><tr><td> MaxIndex = bestimmeRelativenMaxIndex()</td></tr><tr><td> Ausgabe: Halter, Ladezustand und Kennzeichen des Fahrzeugs an der Stelle MaxIndex</td></tr></table>	wiederhole für i=0 bis Anzahl-1	Ausgegeben von Elektroautos[i] auf false setzen	wiederhole für i=0 bis Anzahl-1	MaxIndex = bestimmeRelativenMaxIndex()	Ausgabe: Halter, Ladezustand und Kennzeichen des Fahrzeugs an der Stelle MaxIndex		8	4	12
wiederhole für i=0 bis Anzahl-1										
Ausgegeben von Elektroautos[i] auf false setzen										
wiederhole für i=0 bis Anzahl-1										
MaxIndex = bestimmeRelativenMaxIndex()										
Ausgabe: Halter, Ladezustand und Kennzeichen des Fahrzeugs an der Stelle MaxIndex										
	Summe	18	30	12	60					

Die Aufgabe und das Material beziehen sich auf die folgenden Quellen:

http://de.wikipedia.org/wiki/Vehicle_to_Grid (abgerufen am 4.8.2011).

Reinhard Löser: Elektroautos - die rollenden Stromspeicher, in: Spektrum der Wissenschaft (Heft 04/09) 2009, S. 96 ff.

III. Bewertung und Beurteilung

Die Bewertung und Beurteilung erfolgt gemäß den Bestimmungen in der OAVO in der jeweils geltenden Fassung, insbesondere § 33 OAVO in Verbindung mit den Anlagen 9a und ggf. 9b bis 9f, sowie in den Einheitlichen Prüfungsanforderungen in der Abiturprüfung (EPA). Für die Umrechnung von Prozentanteilen der erbrachten Leistungen in Notenpunkte nach § 9 Abs. 12 der OAVO gelten die Werte in der Anlage 9a der OAVO. Darüber hinaus sind die Vorgaben des Erlasses „Hinweise zur Vorbereitung auf die schriftlichen Abiturprüfungen im Landesabitur 2016“ vom 20. Juni 2014 zu beachten.

Im Fach Informatik (Grundkurs) können Vorschläge zu den Themen der drei Kurshalbjahre Q1 (objektorientierte Modellierung), Q2 (Datenbanken) und Q3 (Konzepte und Anwendungen der theoretischen Informatik) vorgelegt werden.

Die Prüfungsleistung besteht aus der Bearbeitung von zwei Vorschlägen, einem zum Halbjahresthema „objektorientierte Modellierung“ und einem weiteren zu einem der beiden anderen Halbjahresthemen, wofür insgesamt maximal 100 BE vergeben werden können. Ein Prüfungsergebnis von **5 Punkten (ausreichend)** setzt voraus, dass insgesamt 46 BE, ein Prüfungsergebnis von **11 Punkten (gut)**, dass insgesamt 76 BE erreicht werden.