

Mündliche Abiturprüfung / Mündliche Abschlussprüfung Fach Informatik - Grundkurs

Prüfender Fachlehrer (Autor der Aufgabe): _____

Vorbereitungszeit: 20 min, Prüfungszeit 30 min

Automaten und Sprachen

1. Einordnung der Aufgabe in den Lehrplan, Taxonomie:

Die Aufgabe bezieht sich auf den Lernbereich 8A: *Theoretische Informatik – Theoretische Grundlagen* von Programmiersprachen:

Einblick gewinnen in den Aufbau von Sprachen Syntax und Semantik	natürliche, künstliche, formale Sprachen
Kennen des hierarchischen Regelaufbaus formaler Sprachen	Klassen von formalen Sprachen nach Chomsky reguläre Sprachen kontextsensitive Sprachen kontextfreie Sprachen
Einblick gewinnen in den Prozess der Synthese Kennen der Analyse von Sprachelementen mit Hilfe von Automatenmodellen	Erzeugungsprozess durch Regelanwendung Aufbau und Arbeitsweise anhand einfacher Beispiele endlicher Automat Kellerautomat Turingmaschine
Anwenden der Kenntnisse zur Sprachanalyse auf Funktionsprinzipien von Compiler und Interpreter	

Kognitive Lernziele:

- Der/die Schüler:in leitet Wörter aus einer Grammatik ab.
- Der/die Schüler:in kennt die Klassen der Chomsky-Hierarchie, mit ihren entsprechenden Merkmalen und Automaten, die die Sprachen jeweils erkennen.
- Der/Die Schüler:in analysiert die Grammatik hinsichtlich von Klassenmerkmalen.
- Der/die Schüler:in versteht die formale Darstellung einer Sprache.
- Der/die Schüler:in zeichnet zu einer Sprache einen korrekten DEA.

Affektive Lernziele:

- Der/die Schüler:in ist beachtet den Unterschied zwischen Syntax und Semantik.
- Der/die Schüler:in begründet die Notwendigkeit der Formalisierung von Sprache am Beispiel des Sprachverarbeitungsprozesses.

2. Aufgabenstellung (so wie sie dem Prüfling vorgelegt wird):

Automaten und Sprachen

Es gibt sehr viele verschiedene Sprachen: auch mathematische Notation kann als Sprache verstanden werden. Eine vereinfachte Notation von Termen kann mithilfe folgender Grammatik beschrieben werden:

$$G = (V, T, S, P)$$

$$\Sigma = \{x, +, y, (,)\}, S$$

$$T = \{x, +, y, (,)\}$$

$$P = \{S \rightarrow (S + S), \quad (R1)$$

$$S \rightarrow x, \quad (R2)$$

$$S \rightarrow y\} \quad (R3)$$

1. Stellen Sie die gegebene Grammatik vor, indem Sie
 - a. zunächst die Elemente der Grammatik am Beispiel benennen
 - b. an der Grammatik eine wesentliche Eigenschaft von Typ-2-Grammatiken im Unterschied zu Typ-1-Grammatiken zeigen
 - c. alle Klassen formaler Grammatiken und der zugehörigen formalen Sprachen in der Hierarchie nach Chomsky und das Ordnungsprinzip der Hierarchie nennen;
 - d. die Automaten nennen, die die Sprache der gegebenen Grammatik erkennen können;
 - e. drei aus dieser Grammatik abgeleitete Wörter unterschiedlicher Länge angeben.
1. Wir verändern nun die Sprache, um eine andere mathematische Notation darstellen zu können:
 $\Sigma = \{x, y, +\}$
 $L = x^+ + \{xy\}^*$
Zeichnen Sie den zugehörigen deterministischen endlichen Automaten als Zustandsübergangsdiagramm.
Benennen Sie im Vortrag die Elemente des Automaten.
2. Nennen Sie den Unterschied zwischen Syntax und Semantik. Zeigen Sie ihn an einem Wort der Sprache von Aufgabe 2 (unter Beachtung der üblichen mathematischen Notation) und einen Satz der deutschen Sprache.
Nennen Sie zwei Phasen der Sprachverarbeitung, in denen eine formalisierte Syntax der Programmiersprachen gebraucht wird.

3. Tabellarisches Erwartungsbild mit Angaben der jeweils erreichbaren BE und der Zuordnung zu den Anforderungsbereichen:

Aufgabe Nr.	Sachverhalt	AB1	AB2	AB3
1	Elemente der Grammatik	2		
	Abgrenzung Typ 1 – Typ 2		1	
	Klassen und Sprachen hierarchisch Ordnungsprinzip	2 1 1		
	Automaten der Sprache	1		1
	abgeleitete Terme		3	
2	Zeichnung 1 – richtige Abfolge vor dem + 1 – richtige Abfolge nach dem + 1 – Übergang bei + 1 - richtige Darstellungsweise (Kreise, Pfeile, Beschriftungen, Start- und Endzustand) 1 – minimal (richtige Teilprobleme verstanden)		1 2 1 1 1	
	Benennung der Elemente	1		
3	Unterschied am Beispiel zeigen - Definition Syntax und Semantik - 2 Beispiele selber finden - Beispiele erklären	1		2
	Nennen der 2 Phasen		2	
	Summe BE	9	14	3
	Gesamt		25	

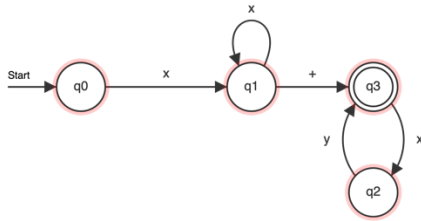
4. Musterlösung mit Angabe der Zuordnung der einzelnen BE:

Automaten und Sprachen – Musterlösung

Es gibt sehr viele verschiedene Sprachen: auch mathematische Notation kann als Sprache verstanden werden. Eine vereinfachte Notation von Termen kann mithilfe folgender Grammatik beschrieben werden:

$$\begin{aligned}G &= (\Sigma, T, S, P) \\ \Sigma &= \{x, +, y, (,)\} \\ T &= \{x, +, y, (,)\} \\ P &= \{S \rightarrow (S + S), \quad (R1) \\ &\quad S \rightarrow x, \quad (R2) \\ &\quad S \rightarrow y\} \quad (R3)\end{aligned}$$

2. Stellen Sie die gegebene Grammatik vor, indem Sie
 - a. zunächst die Elemente der Grammatik am Beispiel benennen [2 BE]
 Σ = Alphabet aus
Terminalen und
Nichtterminalen (Variablen) – hier nur S
T = Terminale (aus denen die Wörter am Ende bestehen)
S = Startsymbol
P = Produktionsregeln
 - b. an der Grammatik eine wesentliche Eigenschaft von Typ-2-Grammatiken im Unterschied zu Typ-1-Grammatiken zeigen
- links stehen nur Nichtterminale [1 BE]
 - c. alle Klassen formaler Grammatiken und der zugehörigen formalen Sprachen in der Hierarchie nach Chomsky und das Ordnungsprinzip der Hierarchie nennen;
(1) Typ-3-Grammatik: reguläre Sprachen [0.5 BE]
(2) Typ-2-Grammatik: kontextfreie Sprachen [0.5 BE]
(3) Typ-1-Grammatik: kontextsensitive Sprachen [0.5 BE]
(4) Typ-0-Grammatik: alle Sprachen [0.5 BE]
Hierarchie nach zugeordnetem Typ-x [1 BE]
geordnet nach Weite der Definition (reguläre Sprachen am meisten eingegrenzt) [1 BE]
 - d. die Automaten nennen, die die Sprache der gegebenen Grammatik erkennen können;
- Kellerautomat + Turingmaschine [2 BE]
 - e. drei aus dieser Grammatik abgeleitete Wörter unterschiedlicher Länge angeben.
- z.B. x, (x + y), (x + (x + x)), ((x + x) + (x + x)) [3 BE]
3. Wir verändern nun die Sprache, um eine andere mathematische Notation darstellen zu können:
 $\Sigma = \{x, y, +\}$
 $L = x^+ + \{xy\}^*$
Zeichnen Sie den zugehörigen deterministischen endlichen Automat als Zustandsübergangsdiagramm. [6 BE]



Benennen Sie im Vortrag die Elemente des Automaten: Zustand, Startzustand, Endzustand, Übergänge [1 BE]

4. Nennen Sie den Unterschied zwischen Syntax und Semantik. Zeigen Sie ihn an einem Wort der Sprache von Aufgabe 2 (unter Beachtung der üblichen mathematischen Notation) und einen Satz der deutschen Sprache.

Nennen Sie zwei Phasen der Sprachverarbeitung, in denen eine formalisierte Syntax der Programmiersprachen gebraucht wird.

- Bsp: „x + „, und sowas wie „Der Baum isst Pizza“ [2 BE]
- Syntax: richtige Regelanwendung, Semantik: richtige Bedeutung [1 BE]
- syntaktisch richtig heißt nicht unbedingt semantisch richtig
- z.B. Term richtig gebildet aber mathematisch trotzdem nicht korrekt [1 BE]
- z.B. Satz syntaktisch richtig, aber ergibt keinen Sinn [1 BE]
- Lexikalische Analyse (Scanner) benötigt formale Grammatik, um Schlüsselwörter zu erkennen [1 BE]
- syntaktische Analyse (Parser) beim Kompilieren benötigt formale Grammatik, um aus Aufbau des Programms einen korrekten Syntaxbaum zu bilden [1 BE]
- Nur Quellcode ohne syntaktische Fehler kann kompiliert werden

5. Hinweise zur Umsetzung (benötigte Arbeitsmittel, ggf. Software auf dem Prüfungsrechner, ...):

Bei der Auswahl dieser Aufgabe ist zu beachten:

- Aufgabe 2 im Vorbereitungsraum muss auf einer Folie für OHP gelöst werden
- Für Aufgabe 1 stehen bei Bedarf weitere Folien zur Verfügung (z.B. für Syntax-/Ableitungsbäume)

6. Anhang: Abbildungen:

7. Quellenangabe, Abbildungsnachweise, ...:

Gesellschaft für Informatik (GI) e.V. (2016): Bildungsstandards Informatik für die Sekundarstufe II. Beilage zu LOG IN 2016, 36(183/184).

Sächsisches Staatsministerium für Kultus (2019): Lehrplan Gymnasium. Informatik. Dresden: Comenius-Institut.

8. Erklärung der Freigabe zur Nachnutzung der Aufgabe:

Hiermit erkläre ich Lioba, Kauk diese Aufgabe unter Wahrung des Urheberrechts erstellt zu haben.

Ich stelle diese Aufgabe zur Nachnutzung nach Lizenz CC BY-NC (Namensnennung, Bearbeitung, nicht kommerziell) zur Verfügung.



L. Kauk

(Unterschrift des Autors / elektron. Signatur)