

Leistungskurs Informatik

Was machen wir?

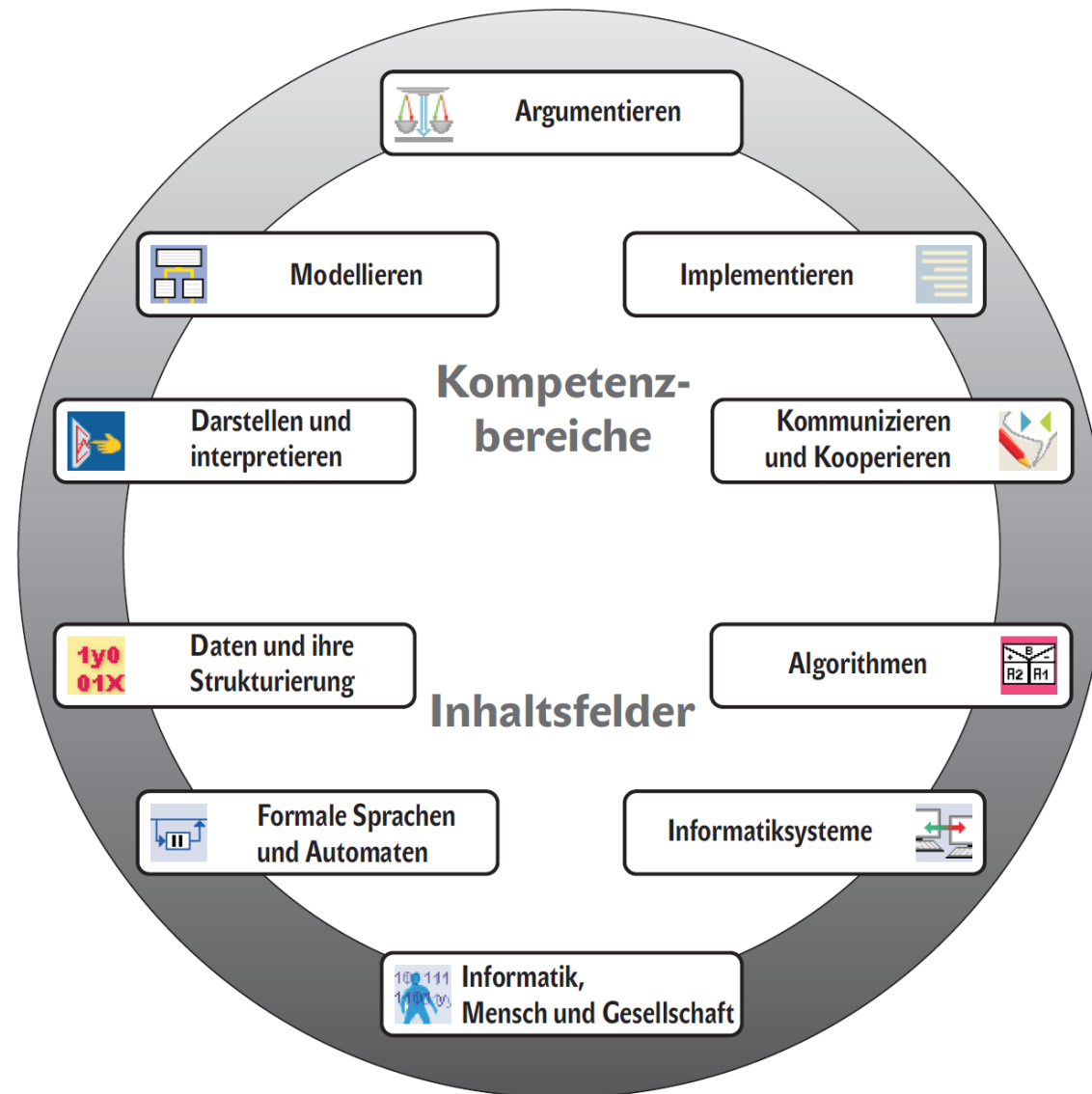
(Prozessbezogene Kompetenzbereiche)

- Argumentieren
- Modellieren
- Implementieren
- Darstellen und Interpretieren
- Kommunizieren und Kooperieren

Was machen wir? (Inhaltsfelder)

- Daten und ihre Strukturierung
- Algorithmen
- Formale Sprachen und Automaten
- Informatiksysteme
- Informatik, Mensch und Gesellschaft

Was machen wir?



Leistungskurs

Daten und ihre Strukturierung*	Algorithmen*	Formale Sprachen und Automaten*	Informatiksysteme	Informatik, Mensch und Gesellschaft
<p>Objekte und Klassen</p> <ul style="list-style-type: none"> – Entwurfsdiagramme und Implementationsdiagramme – Lineare Strukturen <ul style="list-style-type: none"> • array bis zweidimensional • Stapel (Klasse Stack) • Schlange (Klasse Queue) • lineare Liste (Klasse List) – Nicht-lineare Strukturen <ul style="list-style-type: none"> • Binärbaum (Klasse BinaryTree) • binärer Suchbaum (Klasse BinarySearchTree) • Graphen (Klassen Graph, Vertex, Edge) 	<p>Analyse, Entwurf und Implementierung von Algorithmen</p> <ul style="list-style-type: none"> – Struktogramme 	<p>Syntax und Semantik einer Programmiersprache</p> <ul style="list-style-type: none"> – Java – SQL 	<p>Einzelrechner und Rechnernetzwerke</p>	<p>Wirkungen der Automatisierung</p> <ul style="list-style-type: none"> – Grundprinzipien des Datenschutzes
<p>Datenbanken</p> <ul style="list-style-type: none"> – Klassen DatabaseConnector, QueryResult 	<p>Algorithmen in ausgewählten informatischen Kontexten</p> <ul style="list-style-type: none"> – Algorithmen zur Kommunikation in Netzwerken (Klassen Connection, Client, Server) 	<p>Endliche Automaten und Kellerautomaten</p> <ul style="list-style-type: none"> – Deterministische endliche Automaten – Nichtdeterministische endliche Automaten – Nichtdeterministische Kellerautomaten 	<p>Nutzung von Informatiksystemen</p>	<p>Grenzen der Automatisierung</p>
		<p>Grammatiken regulärer und kontextfreier Sprachen</p> <ul style="list-style-type: none"> – Produktionen mit ϵ 	<p>Sicherheit</p>	
		<p>Scanner, Parser und Interpreter für eine reguläre Sprache</p>		
		<p>Möglichkeiten und Grenzen von Automaten und formalen Sprachen</p>		

Themenübersicht

- OOM und OOP
 - Diagramme (UML), Kontrollstrukturen
- Dynamische und statische lineare Datenstrukturen
 - Feld, Schlange, Stapel, Liste
- Such- und Sortialgorithmen
- Relationale Datenbanken
 - SQL
- Nichtlineare Datenstrukturen
 - Baum, Graph

Leistungskurs

Daten und ihre Strukturierung*	Algorithmen*	Formale Sprachen und Automaten*	Informatiksysteme	Informatik, Mensch und Gesellschaft
<p>Objekte und Klassen</p> <ul style="list-style-type: none"> – Entwurfsdiagramme und Implementationsdiagramme – Lineare Strukturen <ul style="list-style-type: none"> • array bis zweidimensional • Stapel (Klasse Stack) • Schlange (Klasse Queue) • lineare Liste (Klasse List) – Nicht-lineare Strukturen <ul style="list-style-type: none"> • Binärbaum (Klasse BinaryTree) • binärer Suchbaum (Klasse BinarySearchTree) • Graphen (Klassen Graph, Vertex, Edge) 	<p>Analyse, Entwurf und Implementierung von Algorithmen</p> <ul style="list-style-type: none"> – Struktogramme 	<p>Syntax und Semantik einer Programmiersprache</p> <ul style="list-style-type: none"> – Java – SQL 	<p>Einzelrechner und Rechnernetzwerke</p>	<p>Wirkungen der Automatisierung</p> <ul style="list-style-type: none"> – Grundprinzipien des Datenschutzes
<p>Datenbanken</p> <ul style="list-style-type: none"> – Klassen DatabaseConnector, QueryResult 	<p>Algorithmen in ausgewählten informatischen Kontexten</p> <ul style="list-style-type: none"> – Algorithmen zur Kommunikation in Netzwerken (Klassen Connection, Client, Server) 	<p>Endliche Automaten und Kellerautomaten</p> <ul style="list-style-type: none"> – Deterministische endliche Automaten – Nichtdeterministische endliche Automaten – Nichtdeterministische Kellerautomaten 	<p>Nutzung von Informatiksystemen</p>	<p>Grenzen der Automatisierung</p>
		<p>Grammatiken regulärer und kontextfreier Sprachen</p> <ul style="list-style-type: none"> – Produktionen mit ϵ 	<p>Sicherheit</p>	
		<p>Scanner, Parser und Interpreter für eine reguläre Sprache</p>		
		<p>Möglichkeiten und Grenzen von Automaten und formalen Sprachen</p>		

Leistungskurs

Daten und ihre Strukturierung*	Algorithmen*	Formale Sprachen und Automaten*	Informatiksysteme	Informatik, Mensch und Gesellschaft
<p>Objekte und Klassen</p> <ul style="list-style-type: none"> – Entwurfsdiagramme und Implementationsdiagramme – Lineare Strukturen <ul style="list-style-type: none"> • array bis zweidimensional • Stapel (Klasse Stack) • Schlange (Klasse Queue) • lineare Liste (Klasse List) – Nicht-lineare Strukturen <ul style="list-style-type: none"> • Binärbaum (Klasse BinaryTree) • binärer Suchbaum (Klasse BinarySearchTree) • Graphen (Klassen Graph, Vertex, Edge) 	<p>Analyse, Entwurf und Implementierung von Algorithmen</p> <ul style="list-style-type: none"> – Struktogramme 	<p>Syntax und Semantik einer Programmiersprache</p> <ul style="list-style-type: none"> – Java – SQL 	<p>Einzelrechner und Rechnernetzwerke</p>	<p>Wirkungen der Automatisierung</p> <ul style="list-style-type: none"> – Grundprinzipien des Datenschutzes
<p>Datenbanken</p> <ul style="list-style-type: none"> – Klassen DatabaseConnector, QueryResult 	<p>Algorithmen in ausgewählten informatischen Kontexten</p> <ul style="list-style-type: none"> – Algorithmen zur Kommunikation in Netzwerken (Klassen Connection, Client, Server) 	<p>Endliche Automaten und Kellerautomaten</p> <ul style="list-style-type: none"> – Deterministische endliche Automaten – Nichtdeterministische endliche Automaten – Nichtdeterministische Kellerautomaten 	<p>Nutzung von Informatiksystemen</p>	<p>Grenzen der Automatisierung</p>
		<p>Grammatiken regulärer und kontextfreier Sprachen</p> <ul style="list-style-type: none"> – Produktionen mit ϵ 	<p>Sicherheit</p>	
		<p>Scanner, Parser und Interpreter für eine reguläre Sprache</p>		
		<p>Möglichkeiten und Grenzen von Automaten und formalen Sprachen</p>		

Themenübersicht

- Datenschutz, Grenzen der Automatisierung
 - Nutzung von Informatiksystemen
- Netzwerkkommunikation
 - Protokolle
 - Implementation
- Automaten und formale Sprachen
- Arbeitsweise des Computers
 - Maschinennahe Programmiersprachen
 - Scanner und Parser
- Vorbereitung auf das Abitur → Wiederholung / Vertiefung

Leistungskurs

Daten und ihre Strukturierung*	Algorithmen*	Formale Sprachen und Automaten*	Informatiksysteme	Informatik, Mensch und Gesellschaft
<p>Objekte und Klassen</p> <ul style="list-style-type: none"> – Entwurfsdiagramme und Implementationsdiagramme – Lineare Strukturen <ul style="list-style-type: none"> • array bis zweidimensional • Stapel (Klasse Stack) • Schlange (Klasse Queue) • lineare Liste (Klasse List) – Nicht-lineare Strukturen <ul style="list-style-type: none"> • Binärbaum (Klasse BinaryTree) • binärer Suchbaum (Klasse BinarySearchTree) • Graphen (Klassen Graph, Vertex, Edge) 	<p>Analyse, Entwurf und Implementierung von Algorithmen</p> <ul style="list-style-type: none"> – Struktogramme 	<p>Syntax und Semantik einer Programmiersprache</p> <ul style="list-style-type: none"> – Java – SQL 	<p>Einzelrechner und Rechnernetzwerke</p>	<p>Wirkungen der Automatisierung</p> <ul style="list-style-type: none"> – Grundprinzipien des Datenschutzes
<p>Datenbanken</p> <ul style="list-style-type: none"> – Klassen DatabaseConnector, QueryResult 	<p>Algorithmen in ausgewählten informatischen Kontexten</p> <ul style="list-style-type: none"> – Algorithmen zur Kommunikation in Netzwerken (Klassen Connection, Client, Server) 	<p>Endliche Automaten und Kellerautomaten</p> <ul style="list-style-type: none"> – Deterministische endliche Automaten – Nichtdeterministische endliche Automaten – Nichtdeterministische Kellerautomaten 	<p>Nutzung von Informatiksystemen</p>	<p>Grenzen der Automatisierung</p>
		<p>Grammatiken regulärer und kontextfreier Sprachen</p> <ul style="list-style-type: none"> – Produktionen mit ϵ 	<p>Sicherheit</p>	
		<p>Scanner, Parser und Interpreter für eine reguläre Sprache</p>		
		<p>Möglichkeiten und Grenzen von Automaten und formalen Sprachen</p>		

Leistungskurs

Daten und ihre Strukturierung*	Algorithmen*	Formale Sprachen und Automaten*	Informatiksysteme	Informatik, Mensch und Gesellschaft
<p>Objekte und Klassen</p> <ul style="list-style-type: none"> – Entwurfsdiagramme und Implementationsdiagramme – Lineare Strukturen <ul style="list-style-type: none"> • array bis zweidimensional • Stapel (Klasse Stack) • Schlange (Klasse Queue) • lineare Liste (Klasse List) – Nicht-lineare Strukturen <ul style="list-style-type: none"> • Binärbaum (Klasse BinaryTree) • binärer Suchbaum (Klasse BinarySearchTree) • Graphen (Klassen Graph, Vertex, Edge) 	<p>Analyse, Entwurf und Implementierung von Algorithmen</p> <ul style="list-style-type: none"> – Struktogramme 	<p>Syntax und Semantik einer Programmiersprache</p> <ul style="list-style-type: none"> – Java – SQL 	<p>Einzelrechner und Rechnernetzwerke</p>	<p>Wirkungen der Automatisierung</p> <ul style="list-style-type: none"> – Grundprinzipien des Datenschutzes
<p>Datenbanken</p> <ul style="list-style-type: none"> – Klassen DatabaseConnector, QueryResult 	<p>Algorithmen in ausgewählten informatischen Kontexten</p> <ul style="list-style-type: none"> – Algorithmen zur Kommunikation in Netzwerken (Klassen Connection, Client, Server) 	<p>Endliche Automaten und Kellerautomaten</p> <ul style="list-style-type: none"> – Deterministische endliche Automaten – Nichtdeterministische endliche Automaten – Nichtdeterministische Kellerautomaten 	<p>Nutzung von Informatiksystemen</p>	<p>Grenzen der Automatisierung</p>
		<p>Grammatiken regulärer und kontextfreier Sprachen</p> <ul style="list-style-type: none"> – Produktionen mit ϵ 	<p>Sicherheit</p>	
		<p>Scanner, Parser und Interpreter für eine reguläre Sprache</p>		
		<p>Möglichkeiten und Grenzen von Automaten und formalen Sprachen</p>		

Themen im Detail – Q1

OOM und OOP

- Wiederholung:
 - Problemanalyse
 - Objektorientierung (Klassen, Objekte)
 - Algorithmen (Methoden, Kontrollstrukturen)
 - Modellierungen (Klassendiagramme, Struktogramme, Pseudocode)
- Benutzeroberflächen (GUI)
- Kombination der Themen

Lineare Datenstrukturen

- Eigenschaften der Datenstrukturen
- Erarbeitung, Analyse und Verwendung der Abiturklassen
 - Feld (Array)
 - Schlange (Queue)
 - Stapel (Stack)
 - Liste (List)

Such- und Sortieralgorithmen

- Suche
 - Lineare Suche
 - Binäre Suche
- Rekursion
- Sortieren
 - Selection Sort
 - Insertion Sort
 - Quicksort
- Effizienz (Laufzeit und Speicher)

Datenbanken-Modellierung

- Aufbau von Datenbanken
- SQL-Abfragen
- ER-Diagramme
- Normalformen

Nicht-lineare Datenstrukturen

- Grahen
 - Aufbau und Funktionsweise
 - Abiturklasse
 - Wegsuche
- Binäre Bäume
 - Aufbau und Funktionalität
 - Abiturklasse
 - Traversierungen
 - Binäre Suchbäume

Datenbanken - Implementation

- Abiturklassen zum Datenbankzugriff
- Ablauf eines Softwareentwicklungszyklus
- Softwareprojekt mit Datenbankzugriff

Themen im Detail – Q2

Datenschutz

- Blick auf die Informatik
- Moral in der Informatik
- Datenschutz
- Urheberrecht
- Sicherheit und Verschlüsselung

Netzwerk

- Aufbau von Netzwerkkommunikation
- Protokolle
- Client-Server-Netzwerke
- Abiturklassen zum Netzwerkzugriff
- Simulation und Analyse von Netzwerkkommunikation

Automaten und formale Sprachen

- Automaten (Abstrakte Modelle)
- Transduktoren
- Akzeptoren
- DEA, NEA und Epsilon-Übergänge
- Reguläre und kontextfreie Sprachen
- Grenzen von Automaten
- Kellerautomaten
- Das Halteproblem

Arbeitsweise eines Computers

- Aufbau und Struktur von Computern
- Maschinennahe Programmierung
- Scanner
- Parser
- Interpreter

Programmierprojekt und Wiederholung

- Kombination von Themen
- Software mit
 - Netzwerk-Kommunikation
 - Datenbankzugriff
 - Benutzeroberfläche
- Ablauf einer Softwareentwicklung

Abitur (Informationen ohne Gewähr)

- Dauer: 270 Minuten = 3 x 90 Minuten
- Umfang: 3 Aufgaben (Themenbereiche)
 - Keine Auswahl durch Schülerinnen und Schüler
 - Auswahl 3 / 4 durch Lehrkraft
- Inhalt: Alles aus der Q1 und der Q2
- Wiederholungszeit im 2. HJ der Q2

Abitur – Aufgabenstellungen (Beispiele)

a) *Erläutern Sie, warum der endliche Automat zum Zustandsübergangsgraph in Abbildung 1 nichtdeterministisch ist.*

Geben Sie für die Trainingseinheit $rrirhrreerrr$ die Zustandsfolge des Automaten an.

Zeigen Sie, dass der Automat die Trainingseinheit $rrhrrihrerr$ nicht akzeptiert.

Beurteilen Sie, ob es im Sinne des Automaten eine genau 15 Minuten lange Trainingseinheit geben kann, in der die Belastungsphasen der Intervalle ausschließlich mit extensiver Trainingsintensität (e) gelaufen werden können.

Abitur – Aufgabenstellungen (Beispiele)

Stellen Sie die beispielhafte Kommunikation zwischen dem Client und dem Server nach den Vorgaben aus Tabelle 1 tabellarisch dar.

Erweitern Sie das Kommunikationsprotokoll um mindestens zwei geeignete Fehlermeldungen bei der Abwahl eines Faches.

Analysieren Sie die Methoden und erläutern Sie die Funktionalität im Sachkontext.

Beurteilen Sie die Aussage der Schülerin zum Problem bei gescheiterten Anmeldeversuchen.

Abitur – Aufgabenstellungen (Beispiele)

Erläutern Sie die Zeilen 5-17 im Sachzusammenhang!

Vervollständigen Sie die Implementierung ab Zeile 18 gemäß den Vorgaben aus dem ursprünglichen Kommunikationsprotokoll (vgl. Tabelle 1).

Implementieren Sie die Methode `ladewettkampfgruppe` mit der von Ihnen entwickelten SQL-Anweisung.

Informatik = Programmieren?

- Programmieren ist nur ein kleiner (aber meist interessanter) Teil der Informatik
 - Implementationen sind aber ein wichtiger Teil
 - Modellierungen und Fakten sind ebenfalls sehr wichtig
 - Ausführliche Planungen vor dem Programmieren sind sinnvoll
-
- Abituraufgaben sind keine Wirtschaftsaufträge, sondern fassen ein umfangreiches Wissen für sehr viele Themenbereiche zusammen