

LEHRVERANSTALTUNGSBESCHREIBUNG

1. Angaben zum Programm

1.1 Hochschuleinrichtung	Babes-Bolyai Universität, Cluj-Napoca
1.2 Fakultät	Mathematik und Informatik
1.3 Department	Informatik
1.4 Fachgebiet	Informatik
1.5 Studienform	Bachelor
1.6 Studiengang / Qualifikation	Informatik

2. Angaben zum Studienfach

2.1 LV-Bezeichnung	Datenstrukturen und Algorithmen						
2.2 Lehrverantwortlicher – Vorlesung	Lekt. dr. Diana Troancă						
2.3 Lehrverantwortlicher – Seminar	Lekt. dr. Diana Troancă						
2.4 Studienjahr	1	2.5 Semester	2	2.6 Prüfungsform	P	2.7 Art der LV	Verpflichtend

3. Geschätzter Workload in Stunden

3.1 SWS	4	von denen: 3.2 Vorlesung	2	3.3 Seminar/Übung	2
3.4 Gesamte Stundenanzahl im Lehrplan	56	von denen: 3.5 Vorlesung	28	3.6 Seminar/Übung	28
Verteilung der Studienzeit:					Std.
Studium nach Handbücher, Kursbuch, Bibliographie und Mitschriften					12
Zusätzliche Vorbereitung in der Bibliothek, auf elektronischen Fachplattformen und durch Feldforschung					8
Vorbereitung von Seminaren/Übungen, Präsentationen, Referate, Portfolios und Essays					10
Tutorien					6
Prüfungen					8
Andere Tätigkeiten:					-
3.7 Gesamtstundenanzahl Selbststudium	44				
3.8 Gesamtstundenanzahl / Semester	100				
3.9 Leistungspunkte	4				

4. Voraussetzungen (falls zutreffend)

4.1 curricular	Grundlagen der Programmierung
4.2 kompetenzbezogen	Programmierungsfähigkeiten

5. Bedingungen (falls zutreffend)

5.1 zur Durchführung der Vorlesung	<ul style="list-style-type: none"> • Vorlesungsraum, Beamer, Laptop
5.2 zur Durchführung des Seminars / der Übung	<ul style="list-style-type: none"> • Labor, Beamer, Laptop

6. Spezifische erworbene Kompetenzen

Berufliche Kompetenzen	<p>K 4.1 Definieren der Grundkonzepte und Prinzipien der Informatik, sowie der mathematischen Theorien und Modelle</p> <p>K 4.2 Interpretation der formalen Modelle der Mathematik und Informatik</p> <p>K 4.3 Identifizierung der geeigneten Modelle und Methoden für die Lösung realer Probleme</p>
Transversale Kompetenzen	<p>TK1 Anwendung der Regeln für gut organisierte und effiziente Arbeit, für verantwortungsvolle Einstellungen gegenüber der Didaktik und der Wissenschaft, für kreative Förderung des eigenen Potentials, mit Rücksicht auf die Prinzipien und Normen der professionellen Ethik</p> <p>TK3 Anwendung von effizienten Methoden und Techniken für Lernen, Informieren und Recherchieren, für das Entwickeln der Kapazitäten der praktischen Umsetzung der Kenntnisse, der Anpassung an die Bedürfnisse einer dynamischen Gesellschaft, der Kommunikation in rumänischer Sprache und in einer internationalen Verkehrssprache</p>

7. Ziele (entsprechend der erworbenen Kompetenzen)

7.1 Allgemeine Ziele der Lehrveranstaltung	<ul style="list-style-type: none"> Erlernen und Verstehen der grundlegenden Datenstrukturen, sowie die Implementierung verschiedener Algorithmen.
7.2 Spezifische Ziele der Lehrveranstaltung	<ul style="list-style-type: none"> Studium des Konzeptes abstrakter Datentyp, sowie der am meisten verwendeten abstrakte Datentypen Studium der passenden Datenstrukturen für die Implementierung abstrakter Datentypen (Arrays, verkettete Listen, Heaps, Hashtabellen, Binärbäume, usw.) Die Fähigkeit einen bestimmten Datentyp für die Lösung eines Problems auszuwählen und Anwendungen zu entwerfen die verschiedene abstrakte Datentypen verwenden. Die Fähigkeit Daten die in verschiedene Datenstrukturen gespeichert sind zu verwenden. Die Fähigkeit Komplexitäten unterschiedlichen Algorithmen zu vergleichen

8. Inhalt

8.1 Vorlesung	Lehr- und Lernmethode	Anmerkungen
<p>1. Einführung. Datenstrukturen. Abstrakte Datentypen:</p> <ul style="list-style-type: none"> Abstrahierung Komplexitäten 	<ul style="list-style-type: none"> Darstellung der Thematik Diskussion Vortrag 	

<ul style="list-style-type: none"> • Pseudocode Notationen 		
2. Arrays. Iteratoren. Amortisierte Analyse	<ul style="list-style-type: none"> • Vortrag • Beschreibung • Erklärungen • Unterrichtsgespräch • Diskussion • Beispiele 	
3. Verkettete Listen: <ul style="list-style-type: none"> • Einfach verkettete Listen: Repräsentierung und Operationen • Doppelt verkettete Listen: Repräsentierung und Operationen • Iterator für verkettete Listen 	<ul style="list-style-type: none"> • Vortrag • Beschreibung • Erklärungen • Unterrichtsgespräch • Diskussion • Beispiele 	
4. Verkettete Listen: <ul style="list-style-type: none"> • Sortierte verkettete Listen: Repräsentierung und Operationen • Zykläre Listen: Repräsentierung und Operationen • Verkettete Listen auf Arrays: Repräsentierung und Operationen 	<ul style="list-style-type: none"> • Vortrag • Beschreibung • Erklärungen • Unterrichtsgespräch • Diskussion • Beispiele 	
5. Abstrakte Datentypen: <ul style="list-style-type: none"> • Skip Listen • ADT Set/Menge: Beschreibung, Domäne, Schnittstelle/Interface, Repräsentierung • ADT Map: Beschreibung, Domäne, Schnittstelle/Interface, Repräsentierung • ADT Matrix: Beschreibung, Domäne, Schnittstelle/Interface, Repräsentierung 	<ul style="list-style-type: none"> • Vortrag • Beschreibung • Erklärungen • Unterrichtsgespräch • Diskussion • Beispiele 	
6. Binären Heap: <ul style="list-style-type: none"> • Definition, Repräsentierung, spezifische Operationen • HeapSort ADT List: <ul style="list-style-type: none"> • Beschreibung, Domäne, Schnittstelle und mögliche Repräsentierungen ADT Stack/Stapel: <ul style="list-style-type: none"> • Beschreibung, Domäne, Schnittstelle und mögliche Repräsentierungen auf Arrays und verkettete Listen 	<ul style="list-style-type: none"> • Vortrag • Beschreibung • Erklärungen • Unterrichtsgespräch • Diskussion • Beispiele 	

<p>7. ADT Stack:</p> <ul style="list-style-type: none"> • Spezielle Operationen <p>ADT Queue/Schlange:</p> <ul style="list-style-type: none"> • Beschreibung, Domäne, Schnittstelle und mögliche Repräsentierungen auf Arrays, zykläre Arrays und verkettete Listen <p>Aufgaben mit ADT Stack und ADT Queue</p>	<ul style="list-style-type: none"> • Vortrag • Beschreibung • Erklärungen • Unterrichtsgespräch • Diskussion • Beispiele 	
<p>8. ADT Dequeue:</p> <ul style="list-style-type: none"> • Beschreibung und mögliche Repräsentierungen <p>ADT Prioritätsschlange:</p> <ul style="list-style-type: none"> • Beschreibung, Domäne, Schnittstelle und mögliche Repräsentierungen auf Arrays, verkettete Listen und Heaps <p>ADT Binomial-Heap:</p> <ul style="list-style-type: none"> • Beschreibung, Repräsentierung und Operationen 	<ul style="list-style-type: none"> • Vortrag • Beschreibung • Erklärungen • Unterrichtsgespräch • Diskussion • Beispiele 	
<p>9. Hashtabellen:</p> <ul style="list-style-type: none"> • Beschreibung und Eigenschaften • Kollisionbehandlung mittels Verkettung 	<ul style="list-style-type: none"> • Vortrag • Beschreibung • Erklärungen • Unterrichtsgespräch • Diskussion • Beispiele 	
<p>10. Hashtabellen:</p> <ul style="list-style-type: none"> • Kollisionbehandlung mittels coalesced Hashing • Kollisionbehandlung mittels offener Adressierung • Containers repräsentiert auf Hashtabellen 	<ul style="list-style-type: none"> • Vortrag • Beschreibung • Erklärungen • Unterrichtsgespräch • Diskussion • Beispiele 	
<p>11. Bäume:</p> <ul style="list-style-type: none"> • Beschreibung, Eigenschaften • Anwendungen <p>Binärbäume:</p> <ul style="list-style-type: none"> • Beschreibung, Eigenschaften • Domäne und Schnittstelle für ADT Binärbaum • Operationen 	<ul style="list-style-type: none"> • Vortrag • Beschreibung • Erklärungen • Unterrichtsgespräch • Diskussion • Beispiele 	
<p>12. Binärbäume:</p> <ul style="list-style-type: none"> - Traversierungen: rekursive und nicht-rekursive Algorithmen 	<ul style="list-style-type: none"> • Vortrag • Beschreibung • Erklärungen • Unterrichtsgespräch • Diskussion 	

	<ul style="list-style-type: none"> • Beispiele 	
13 Binärsuchbäume: <ul style="list-style-type: none"> • Beschreibung, Eigenschaften • Repräsentierung • Operationen: rekursive und nicht-rekursive Algorithmen • Containers repräsentiert auf Binärsuchbäume 	<ul style="list-style-type: none"> • Vortrag • Beschreibung • Erklärungen • Unterrichtsgespräch • Diskussion • Beispiele 	
14. . Balancierte Binärsuchbäume: <ul style="list-style-type: none"> • AVL Bäume 	<ul style="list-style-type: none"> • Vortrag • Beschreibung • Erklärungen • Diskussion • Beispiele 	

Literatur

In deutscher Sprache:

1. Dietzfelbinger, M., Mehlhorn, K., Sanders, P., Algorithmen und Datenstrukturen. Die Grundwerkzeuge, Springer, 2014
2. Weicker, K., Weicker, N., Algorithmen und Datenstrukturen, Springer, 2013
3. Sedgewick, R., Algorithmen, Addison Wesley, New York, 1991.
4. Wirth, N., Datenstrukturen und Algorithmen, Teubner, Stuttgart, 1999.
5. Solymosi, A., Grude, U., Grundkurs Algorithmen und Datenstrukturen in Java, Eine Einführung in die praktische Informatik, Vieweg Teubner, Wiesbaden, 2008.
6. Ottmann, Th., Widmayer, P., Algorithmen und Datenstrukturen, Springer, 1997.
7. Sattler, K-U., Algorithmen und Datenstrukturen, Springer, 2008.

8. Niculescu V., Czibula G., Structuri fundamentale de date. O perspective orientate obiect. Editura Casa Cartii de Stiinta, Cluj-Napoca, 2011
9. T. Cormen, C. Leiserson, R. Rivest, C. Stein: Introduction to algorithms, Third Edition, The MIT
10. Horowitz, E.: Fundamentals of Data Structures in C++. Computer Science Press, 1995.
11. Frentiu M., Pop H.F., Serban G., Programming Fundamentals, Ed.Presa Universitara Clujeana, Cluj-Napoca, 2006, 234 pagini

8.2 Seminar / Übung / Labor	Lehr- und Lernmethode	Anmerkungen
S1. Komplexitäten	Beschreibung, Erklärungen, Diskussionen, Beispiele, Übungen	2 Stunden jede 2 Wochen
S2. ADT Bag/MultiMenge – Repräsentierung und Implementation auf Array. Iterator für ADT Bag	Beschreibung, Erklärungen, Diskussionen, Beispiele, Übungen	
S3. BucketSort, Lexikographischer Sort, Radix Sort. Merge von sortierten Listen	Beschreibung, Erklärungen, Diskussionen, Beispiele, Übungen	

S4. Sortierte MultiMap – Repräsentierung und Implementation auf verkettete Listen	Beschreibung, Erklärungen, Diskussionen, Beispiele, Übungen	
S5. Schriftliche Zwischen-Prüfung und Projektthemen Verteilung	Beschreibung, Erklärungen, Diskussionen, Beispiele, Übungen	
S6. Hashtabellen. Kollisionbehandlung mittels coalesced Hashing	Beschreibung, Erklärungen, Diskussionen, Beispiele, Übungen	
S7. Binärbäume	Beschreibung, Erklärungen, Diskussionen, Beispiele, Übungen	
	Ein Labor basiert auf eine Datenstruktur. Die Aufgaben bestehen aus unterschiedlichen Containers die mit Hilfe der gegebenen Datenstruktur implementiert werden müssen.	2 Stunden jede 2 Wochen
L1. Dynamisches Array	Unterrichtsgespräch, Erklärungen, Beispiele	Für Alle Aufgaben: <ul style="list-style-type: none"> • Implementierung • Dokumentation • Komplexitäten
L2. Einfach verkettete Liste mit dynamischer Allokation	Unterrichtsgespräch, Erklärungen, Beispiele	Für Alle Aufgaben: <ul style="list-style-type: none"> • Implementierung • Dokumentation • Komplexitäten
L3. Einfach verkettete Liste auf Arrays	Unterrichtsgespräch, Erklärungen, Beispiele	Für Alle Aufgaben: <ul style="list-style-type: none"> • Implementierung • Dokumentation • Komplexitäten
L4. Heap	Unterrichtsgespräch, Erklärungen, Beispiele	Für Alle Aufgaben: <ul style="list-style-type: none"> • Implementierung • Dokumentation • Komplexitäten
L5. Hashtabelle	Unterrichtsgespräch, Erklärungen, Beispiele	Für Alle Aufgaben: <ul style="list-style-type: none"> • Implementierung • Dokumentation • Komplexitäten
L6.-L7. Binärsuchbäume	Unterrichtsgespräch, Erklärungen, Beispiele	Für Alle Aufgaben: <ul style="list-style-type: none"> • Implementierung • Dokumentation • Komplexitäten
Literatur		
In deutscher Sprache:		

12. Dietzfelbinger, M., Mehlhorn, K., Sanders, P., Algorithmen und Datenstrukturen. Die Grundwerkzeuge, Springer, 2014

13. Weicker, K., Weicker, N., Algorithmen und Datenstrukturen, Springer, 2013

14. Sedgewick, R., Algorithmen, Addison Wesley, New York, 1991.

15. Wirth, N., Datenstrukturen und Algorithmen, Teubner, Stuttgart, 1999.

16. Solymosi, A., Grude, U., Grundkurs Algorithmen und Datenstrukturen in Java, Eine Einführung in die praktische Informatik, Vieweg Teubner, Wiesbaden, 2008.

17. Ottmann, Th., Widmayer, P., Algorithmen und Datenstrukturen, Springer, 1997.

18. Sattler, K-U., Algorithmen und Datenstrukturen, Springer, 2008.

19. Niculescu V., Czibula G., Structuri fundamentale de date. O perspective orientate obiect. Editura Casa Cartii de Stiinta, Cluj-Napoca, 2011

20. T. Cormen, C. Leiserson, R. Rivest, C. Stein: Introduction to algorithms, Third Edition, The MIT

21. Horowitz, E.: Fundamentals of Data Structures in C++. Computer Science Press, 1995.

1. Frentiu M., Pop H.F., Serban G., Programming Fundamentals, Ed.Presa Universitara Clujeana, Cluj-Napoca, 2006, 234 pagini

9. Verbindung der Inhalte mit den Erwartungen der Wissensgemeinschaft, der Berufsverbände und der für den Fachbereich repräsentativen Arbeitgeber

Diese Vorlesung wird an international bekannten Universitäten im Fachgebiet Informatik angeboten.

Der Inhalt der Vorlesung entspricht der IEEE und ACM Richtlinien.

10. Prüfungsform

Veranstaltungsart	10.1 Evaluationskriterien	10.2 Evaluationsmethoden	10.3 Anteil an der Gesamtnote
10.4 Vorlesung	<ul style="list-style-type: none"> • Korrekter Umgang mit den Grundbegriffen der Vorlesung. • Kenntnisse für das Lösen von Aufgaben 	Prüfung	60%
10.5 Seminar	<ul style="list-style-type: none"> • Korrekter Umgang mit den Grundbegriffen der Vorlesung. • Kenntnisse für das Lösen von Aufgaben 	Klausur (schriftliche Arbeit) Mitarbeit während der Übungsstunden	20%
10.5 Übung	Korrektheit und Effizienz in der gelösten Aufgaben	Praktische Aufgaben	20%
10.6 Minimale Leistungsstandards			

Um in die Prüfung eingelassen zu werden gelten folgende Kriterien:

- Wenigstens 6 Anwesenheiten bei den Übungen
- Wenigstens 5 Anwesenheiten bei dem Seminar
- Bei den praktischen Aufgaben muss die Mindestnote 5 erzielt werden

Für das Bestehen der Prüfung gelten folgende Kriterien:

- bei der Klausur muss die Mindestnote 5 erzielt werden
- bei den praktischen Aufgaben muss die Mindestnote 5 erzielt werden
- die Endnote muss mindestens 5 sein

Ausgefüllt am:

Vorlesungsverantwortlicher

Seminarverantwortlicher

Lekt. Dr. Diana Troancă

Lekt. Dr. Diana Troancă

Genehmigt im Department am:

Departmentdirektor

Prof. Dr. Andreica Anca