



Fakultät Informatik

Studiengang Informatik - Game Engineering (Bachelor)

Modulhandbuch

Stand: WS 2021/22

August 2021

Prof. Dr. E. Müller
Studiendekan der Fakultät Informatik

Prof. Dr. B. Dreier
Studiengangkoordinator

Prof. Dr. U. Göhner
Vorsitzender der Prüfungskommission

Inhaltsverzeichnis

1 Ziele und Aufbau des Studiengangs Informatik - Game Engineering	2
2 Begriffserläuterungen	6
3 Modulbeschreibungen	8
GEB1101 Lineare Algebra und Analytische Geometrie	8
GEB1102 IT-Systeme	10
GEB1103 Einführung in die Informatik	11
GEB1104 Modellierung und Animation 1	12
GEB1105 Programmieren 1	14
GEB1106 Analysis	16
GEB1107 Theoretische Informatik	18
GEB1108 Game Design	19
GEB1109 Gestaltung und Zeichnen	20
GEB1110 Programmieren 2	22
GEB1111 Algorithmen und Datenstrukturen	24
GEB1112 Diskrete Mathematik für Games	26
GEB1113 Datenbanken	28
GEB1114 Betriebssysteme	30
GEB1115 Computergrafik	32
GEB1116 Software Engineering / Softwaretechnik 1	34
GEB1117 Human Computer Interaction / Softwaretechnik 2	36
GEB1118 Wahrscheinlichkeitsrechnung und Numerik	38
GEB1120 Game Engineering	40
GEB1121 Ansteuerung virtueller Charaktere	41
GEB1122 Verteilte Softwaresysteme	43
GEB1123 Projektmanagement	44
GEB5200 GE-Lab	46
GEB2101 Game Physik	47
GEB2104 Computergestützte Bildbearbeitung mit Photoshop	49
GEB2106 Design interaktiver Anwendungen	50
GEB2107 Künstliche Intelligenz für Games	52
GEB2109 Digitale Medien	54
GEB2114 Operations Research	56
GEB2115 Modellierung und Animation 2	58
GEB2116 Sound- und Musikproduktion für Games	60
GEB2117 Farb- und Formpsychologie	61
GEB2118 Logik	64
GEB3100.1 Praktisches Studiensemester	66
GEB3100.2 Praxisbegleitende Lehrveranstaltung	67
GEB4100 Seminar	68
GEB5100 Projektarbeit	69
GEB6100.1/6100.2 Bachelorarbeit / -seminar	70

1 Ziele und Aufbau des Studiengangs Informatik - Game Engineering

Ziel des Studiengangs Informatik - Game Engineering ist es, den Studierenden aufbauend auf einem grundständigen Informatikstudium ausgeprägte Kenntnisse und Fähigkeiten in der gestalterisch-technischen Entwicklung, im Softwareengineering, in der Realisierung von Echtzeitsimulationen, in der Schaffung von immersiven, dreidimensionalen Welten, sowie in 3D-Visualisierungen für Computerspiele oder anderen Anwendungen, zu vermitteln.

Die Studierenden verstehen die Vorgehensweisen in Modellierung, Animation sowie allgemeiner und spezieller Algorithmen für Games und Computergrafik und können diese anwenden. Dies wird in diesem Bachelor-Studiengang nicht ausschließlich auf Computerspiele ausgerichtet, so dass den Studierenden nach Abschluss eine breite Auswahl an Berufen oder Zusatzausbildungen offen steht.

Der Studiengang fördert zudem die für die berufliche Praxis notwendige Fähigkeit zur Kommunikation und Teamarbeit sowie das Verantwortungsbewusstsein für den Umgang mit moderner Informationstechnik.

Der Bachelorstudiengang Informatik - Game Engineering ist auch eine Basis und Zugangsmöglichkeit für eine anwendungsorientierte Weiterqualifizierung in einem sich anschließenden konsekutiven Masterstudiengang.

Absolventinnen und Absolventen des Studiengangs Informatik - Game Engineering erwerben während Ihres Studiums vielfältige Kenntnisse und Fähigkeiten. Sie besitzen grundlegendes Verständnis für zentrale Konzepte der Informatik, für Hard- und Softwaresysteme und Programmierung. Zu ihren Kompetenzen zählen die Befähigung, aus konkreten Fragestellungen der Praxis entstandene informationstechnische Probleme systemgerecht zu analysieren und Lösungen unter Beachtung technischer, ökonomischer und ergonomischer Randbedingungen zu erstellen. Die Absolventen beherrschen rechnerorientierte Arbeits- und Verfahrensweisen, deren Kernpunkt die Softwareentwicklung darstellt. Sie verfügen über logisches und algorithmisches Denken und die Fähigkeit, in abstrakten Modellen zu denken. Sie besitzen Verständnis der Methodik der Modellbildung, die Fähigkeit zur Planung und Durchführung von Softwareprojekten, sind kontaktfähig und teamfähig.

Nachfolgende Tabelle zeigt die angestrebten Studienziele und Lernergebnisse des Studiengangs Informatik - Game Engineering der Hochschule Kempten:

Nr.	Studienziel	Lernergebnisse
1	Grundlagenkompetenz	Grundlegendes Verständnis für zentrale Konzepte der Informatik Kenntnisse über formale, algorithmische und mathematische Hilfsmittel der Informatik
2	Game Engineering-Kompetenz	Verstehen und Anwenden der im Gaming angewandten Theorien und Methoden Fähigkeit zur Analyse, Bewertung und zur eigenen Entwicklung von Computerspielen, ergänzt durch Einblicke in relevante Kreativitätstechniken

3	Analyse-, Design- und Realisierungskompetenz	Kenntnisse der für die Informatik typischen Vorgehensmodelle und Methoden zur Analyse, Modellierung, Realisierung und Qualitätssicherung Fähigkeit, in abstrakten Modellen zu denken und konstruktiv vorzugehen Übertragung dieser Vorgehensweisen auf den Bereich der Game- und Simulationsentwicklung
4	Anwendungskompetenz	Kenntnisse über Aufbau von gängigen, heutigen Games und Verständnis für die anwendungsbezogenen Zusammenhänge Fähigkeit, Lösungen für spezielle Anforderungen beurteilen und erarbeiten.
5	Soziale und überfachliche Kompetenzen	Kenntnisse in Arbeits-, Präsentations- und Kommunikationstechniken Erlernen von Fertigkeiten im Umgang mit Personen, Gruppen und Institutionen im Kontext des späteren Berufsfeldes Fähigkeit, im Team fachlich als auch leitend verantwortliche Funktionen zu übernehmen

Das Studium "Informatik - Game Engineering" teilt sich in ein Basisstudium sowie ein Vertiefungsstudium. Die Module des Basisstudiums orientieren sich an den Grundlagen der oben genannten Fachdisziplinen. Sie sollen den Studierenden zu Beginn des Studiums ermöglichen, sich in Fachdidaktik und "Fachsprachen" einzuarbeiten. Im Basisstudium finden sich folglich die Modulbereiche für die fachlichen Grundlagen in den Bereichen Mathematik, Theorie der Informatik, Modellierung, Animation und Computergrafik, aber auch in der klassischen Informatik. Die Lehrveranstaltungen finden in Präsenz mit kleinen Übungsgruppen statt. Im Vertiefungsstudium werden darüber hinaus unterschiedliche Lehr- und Lernformen eingesetzt. So finden sich neben Seminaren und Übungen auch Projektarbeiten.

Der Zusammenhang zwischen den übergeordneten Studienzielen (1) Grundlagenkompetenz, (2) Game Engineering Kompetenz, (3) Analyse-, Design und Realisierungskompetenz, (4) Anwendungskompetenz und (5) Soziale und überfachliche Kompetenzen sowie den Lernergebnissen des Bachelorstudiengangs Informatik nebst dem Beitrag der Wahlpflichtmodule zur Umsetzung dieser Ziele sind in der folgenden Zielmatrix dargestellt:

Modul	Studienziele				
	1	2	3	4	5
Einführung in die Informatik	++				
Analysis	++		+		
Programmieren 1	+		+	+	
IT-Systeme		++			
Modellierung und Animation 1		++	+	+	
Lineare Algebra und Analytische Geometrie	++		+	+	
Theoretische Informatik	++				
Programmieren 2	+		+	+	
Algorithmen und Datenstrukturen	++			+	
Game Design		++			+
Gestaltung und Zeichnen		++		+	+
Diskrete Mathematik für Games	++		+		
Software Engineering	+		++		+
Human Computer Interaction			++	+	+
Datenbanken	+		+		
Betriebssysteme	+	++			

Computergrafik				++	+
Wahrscheinlichkeitsrechnung und Numerik	++		+		
Verteilte Softwaresysteme	+	++		+	
Game Engineering		+	+		+
GE-Lab		+	+		++
Projektmanagement			+		++
Praktisches Studiensemester					++
Praxisbegleitende Lehrveranstaltung					++
Seminar		+		+	++
Projektarbeit			+	+	++
Ansteuerung virtueller Charaktere	++			++	
Fachwissenschaftliche Wahlpflichtfächer		++	+	+	+
Bachelorarbeit / -seminar				++	+

Mit einem Studienabschluss "Informatik - Game Engineering" bieten sich den Absolventen heute und zukünftig vielfältige Einsatzmöglichkeiten, unter anderem im Bereich der Trainingsapplikationen, der 3D Simulationen und anderen gameverwandten Sparten.

Durch die rasante Entwicklung im DV-Sektor entstehen aber auch immer wieder neue Berufs- und Aufgabenfelder. Durch Fokussierung auf die Konzepte des Game Engineerings und nicht auf konkrete Anwendungsprogramme bzw. Game Engines sollen Studierende dazu befähigt werden auch diese zukünftigen Anforderungen zu meistern.

Aufbau des Studiengangs Informatik - Game Engineering

Informatik - Game Engineering, Bachelor (B.Sc.)

Hochschule für angewandte Wissenschaften Kempten

Semester

gültig mit Studienbeginn WS 2019/20

7	BA-SEM ⁽⁸⁾	Bachelorarbeit ⁽⁸⁾			WP-Fach ⁽⁶⁾	WP-Fach ⁽⁶⁾	WP-Fach ⁽⁶⁾
6	Projektarbeit ⁽⁸⁾			Seminar ⁽⁸⁾	WP-Fach ⁽⁶⁾	WP-Fach ⁽⁶⁾	
5	Praktisches Studiensemester ⁽⁷⁾						Praxisbegleitende Lehrveranstaltung ⁽⁴⁾
4	Wahrscheinlichkeitsrechnung und Numerik ⁽¹⁾	Diskrete Mathematik für Games ⁽¹⁾	Human-Computer Interaction ⁽⁵⁾	Ansteuerung virtueller Charaktere ⁽⁵⁾	Verteilte Softwaresysteme ⁽³⁾	IT-Projektmanagement ⁽⁴⁾	
3	Datenbanken ⁽³⁾	GE-Lab ⁽⁵⁾	Game Engineering ⁽⁵⁾	Computergrafik ⁽⁵⁾	Software Engineering ⁽³⁾	Betriebssysteme ⁽³⁾	
2	Analysis ⁽¹⁾	Theoretische Informatik ⁽²⁾	Game Design ⁽⁵⁾	Zeichnen ⁽⁵⁾	Programmieren 2 ⁽³⁾	Algorithmen und Datenstrukturen ⁽³⁾	
1	Lineare Algebra und Analytische Geometrie ⁽¹⁾	IT Systeme ⁽²⁾	Einführung in die Informatik ⁽³⁾	Modellierung und Animation I ⁽⁵⁾	Programmieren 1 ⁽³⁾		

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
---	---	---	---	---	---	---	---	---	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

ECTS-Punkte

Legende:

1	Mathematik	5	Game Engineering und Design
2	Theoretische Informatik & Computertechnik	6	Wahlfach
3	Softwaretechnik	7	Praktikum
4	Angewandte Informatik und interdisziplinäre Fächer	8	Seminar- und Abschlussarbeit

Für den Studiengang Informatik - Game Engineering stehen folgende Ansprechpartner zur Verfügung:

Studiengangkoordinator:	Prof. Dr. Bernd Dreier
Studienfachberater:	Prof. Dr. Klaus Ulhaas
Beauftragter für das Praxissemester:	Prof. Dr. Bernd Dreier
Vorsitzender der Prüfungskommission:	Prof. Dr. Ulrich Göhner



2 Begriffserläuterungen

ECTS - European Credit Transfer System

Diese Vereinbarungen zur Anrechnung, Übertragung und Akkumulierung von Studienleistungen basieren auf dem Arbeitspensum, das Studierende durchzuführen haben, um die Ziele des Lernprogramms zu erreichen. Für jede studienbezogene Leistung wird der voraussichtliche durchschnittliche Arbeitsaufwand angesetzt und auf das Studienvolumen angerechnet. Der Arbeitsaufwand umfasst Präsenzzeit und Selbststudium ebenso wie die Zeit für die Prüfungsleistungen, die notwendig sind, um die Ziele des vorher definierten Lernprogramms zu erreichen. Mit dem ECTS können Studienleistungen international angerechnet und übertragen werden.

Arbeitsaufwand (Workload) und Leistungspunkte (ECTS-LP)

Der Arbeitsaufwand der Studierenden wird im ECTS in Credit Points angegeben. Deutsche Übersetzungen für Credit Point sind die Begriffe Leistungspunkt oder ECTS-Punkt. Ein Arbeitsaufwand von 30 Zeitstunden bedeutet einen Leistungspunkt. Der Arbeitsaufwand von Vollzeitstudierenden entspricht 60 Leistungspunkten pro Studienjahr, also 30 Leistungspunkten pro Semester. Das sind 1.800 Stunden pro Jahr oder 45 Wochen/Jahr mit 40 Stunden/Woche.

Der Arbeitsaufwand setzt sich zusammen aus:

- Präsenzzeit
- Zeit für die Vor- und Nachbereitung des Vorlesungsstoffs,
- Zeit für die Vorbereitung von Vorträgen und Präsentationen,
- Zeit für die Erstellung eines Projekts,
- Zeit für die Ausarbeitung einer Studienarbeit,
- Zeit für notwendiges Selbststudium,
- Zeit für die Vorbereitung auf mündliche oder schriftliche Prüfungen.

Die Bachelorstudiengänge mit sieben Semestern bescheinigen erfolgreichen Studierenden 210 ECTS-LP, die dreisemestrigen Masterstudiengänge weitere 90 ECTS-LP. Damit ist die Forderung nach 300 ECTS-LP für ein erfolgreich abgeschlossenes Masterstudium erfüllt.

Semesterwochenstunden und Präsenzzeit

Eine Semesterwochenstunde ist die periodisch wiederkehrende Lehreinheit in einem Modul, in der Regel im Rhythmus von einer oder zwei Wochen. Dabei wird eine Präsenz von 45 Minuten plus Wegzeiten gerechnet, sodass die Vorlesungsstunde als eine Zeitstunde gewertet wird.

Wir rechnen mit einer Vorlesungszeit von 15 Wochen pro Semester, wodurch sich aus der Zahl der Semesterwochenstunden die geforderte Präsenzzeit ("Kontaktzeit") direkt ableitet: 1 SWS entspricht 15 Stunden Präsenzzeit.

Module

Der Studiengang setzt sich aus Modulen zusammen. Ein Modul repräsentiert eine inhaltlich und zeitlich zusammengehörige Lehr- und Lerneinheit. Module werden in der Regel in einem



Semester abgeschlossen.

Modulgruppen sind Zusammenfassungen von Modulen mit einem weiteren inhaltlichen Zusammenhang. In allen Fällen stellt ein Modul oder ein Teilmodul eine Einheit dar, für die innerhalb und am Ende eines Semesters eine Prüfungsleistung erbracht werden kann, für die Leistungspunkte vergeben werden.

Wahlpflichtmodule werden bedarfsorientiert, meist in jährlichem Rhythmus angeboten. Das jeweilige Semester (Sommer- oder Wintersemester) kann der Modulbeschreibung entnommen werden. Grundsätzlich können Wahlpflichtmodule ab einer Untergrenze von 15 angemeldeten Teilnehmern durchgeführt werden. In besonderen Ausnahmefällen, z. B. bei wiederholter Unterschreitung der Mindestteilnehmerzahl oder erstmaligem Angebot einer Veranstaltung, kann die Fakultät von dieser Regelung abweichen. Die Entscheidung treffen der Fakultätsrat und der Dekan der Fakultät. Wahlpflichtmodule, die gleichzeitig in einem anderen Studiengang als Pflichtmodul gehalten werden, können auch stattfinden, wenn weniger als 15 Anmeldungen vorliegen.

Die Lehrveranstaltungen werden derzeit in deutscher Sprache gehalten.

Studienbegleitende Prüfungen und Studienfortschritt

Sämtliche Prüfungen erfolgen über das gesamte Studium verteilt studienbegleitend und stehen in direktem Bezug zur Lehrveranstaltung. Prüfungsbestandteile können je nach Lehrveranstaltung begleitend oder nach Abschluss des Moduls stattfinden, beispielsweise als Referat, Klausurarbeit, mündliche Prüfung, Hausarbeit mit Kolloquium, Entwurf mit Kolloquium, Laborbericht, Exkursionsbericht oder einer Kombination. In den Beschreibungen der einzelnen Module wird im Modulhandbuch die jeweilige Prüfungsform festgelegt. Eine Wiederholung der Prüfung eines Moduls erfolgt bei Nichtbestehen im folgenden Semester. Die Prüfung für ein Modul darf in der Regel nur einmal wiederholt werden, genau regelt dies die Rahmenprüfungsordnung. Es gelten allgemeine Studienfortschrittsberechtigungen, die die jeweils gültige Studien- und Prüfungsordnung regelt.

3 Modulbeschreibungen

GEB1101 Lineare Algebra und Analytische Geometrie

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Markus Preisenberger
Dozent(en):	Prof. Dr. Markus Preisenberger / Epple
Modultyp:	Pflichtmodul
Voraussetzungen:	keine
Verwendbarkeit:	Studiengang Informatik - Game Engineering (B.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	3 SWS Seminaristischer Unterricht 1 SWS Übung
Leistungspunkte:	5
Arbeitsaufwand:	60 Stunden Präsenzzeit Unterricht/Übung 30 Stunden Selbststudium - Betreute Studierzeit 60 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Schriftliche Prüfung 90 Minuten am Ende des Semesters; Mathematik-Onlinetest als Zulassungsvoraussetzung
Zur Prüfung zugelassene Hilfsmittel:	kein Taschenrechner, ansonsten ohne Einschränkung

Lernergebnisse und Inhalte

Lernergebnisse:

Nach erfolgreicher Teilnahme am Modul Lineare Algebra und Analytische Geometrie können die Studierenden

- wesentliche Begriffe der Linearen Algebra und Analytischen Geometrie wiedergeben
- wesentliche Berechnungsmethoden der Linearen Algebra und Analytischen Geometrie problemabhängig auswählen
- einfache Berechnungen mit Methoden der Linearen Algebra und der Analytischen Geometrie durchführen
- einfache algebraische Beweise analysieren und wiedergeben

Lehrinhalte:

- Grundbegriffe: Mengen, Abbildungen und Relationen
- Koordinatensysteme, Punkte, Vektoren und Vektorrechnung, elementare geometrische Objekte der Ebene und des Raumes, einfache Kurven und Flächen
- Skalarprodukt, Orthogonalität, Längen- und Winkelmessung, Flächen- und Volumemessung, zwei- und dreireihige Determinanten
- Schnittmengenberechnung und lineare Gleichungssysteme
- Lineare und affine Abbildungen der Ebene und des Raumes

- Vektor- und Matrizenrechnung in der Ebene und im Raum
- Homogene Koordinaten
- Vektorräume: Begriff des Vektorraumes, Linearkombinationen, Skalarprodukt, Basis, Dimension, Lineare Unterräume, Orthogonales Komplement, Dimensionsformel
- Lineare Abbildungen: Begriff der linearen Abbildung, Matrix einer linearen Abbildung, Rang, Defekt, Kern und Bild, Eigenwerte, Eigenvektoren und Determinanten

Literatur:

- Brill, M.: Mathematik für Informatiker, Hanser Verlag, 2. Auflage, 2005
- Hachenberger, D.: Mathematik für Informatiker, Pearson Studium, 2. Auflage, 2008
- Hartmann, P.: Mathematik für Informatiker, Springer Vieweg Verlag, 2019
- Stingl, P.: Einstieg in die Mathematik für Fachhochschulen, Hanser Verlag, 2013
- Stingl, P.: Mathematik für Fachhochschulen, Hanser Verlag, 2007

GEB1102 IT-Systeme

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Elmar Böhler
Dozent(en):	Prof. Dr. Elmar Böhler
Modultyp:	Pflichtmodul
Voraussetzungen:	keine
Verwendbarkeit:	Studiengang Informatik (B.Sc.), Studiengang Informatik - Game Engineering (B.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht und 2 SWS Übung
Leistungspunkte:	5
Arbeitsaufwand:	60 Stunden Präsenzzeit Unterricht 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Schriftliche Prüfung 90 Minuten am Ende des Semesters. Leistungsnachweise sind Zulassungsvoraussetzungen
Zur Prüfung zugelassene Hilfsmittel:	handschriftliche Notizen, 1 DIN A4 Blatt, einseitig beschrieben, keine Kopie

Lernergebnisse und Inhalte

Lernergebnisse:

Die Studierenden sind in der Lage,

- den grundlegenden Aufbau, die Komponenten und die Funktionsweise von programmierbaren Systemen zu erläutern
- einfache Assembler-Programme implementieren.

Lehrinhalte:

Begriffsdefinitionen, Boole'sche Algebra, Digitale Logik, Digitale Grundschaltungen, Rechenwerk, Steuerwerk, Speicher, Peripherie, Mikroprogrammierung, Assemblerprogrammierung

Literatur:

- H.-P. Gumm, M. Sommer: Einführung in die Informatik, Oldenbourg Verlag, 2012
- A. Tanenbaum: Rechnerarchitektur: Von der digitalen Logik zum Parallelrechner, Pearson Studium - IT, 2014



GEB1103 Einführung in die Informatik

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Stefan Rieck
Dozent(en):	Prof. Dr. Stefan Rieck
Modultyp:	Pflichtmodul
Voraussetzungen:	keine
Verwendbarkeit:	Studiengang Informatik (B.Sc.), Studiengang Informatik - Game Engineering (B.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	4 SWS Seminaristischer Unterricht
Leistungspunkte:	5
Arbeitsaufwand:	60 Stunden Präsenzzeit Unterricht 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Schriftliche Prüfung 90 Minuten
Zur Prüfung zugelassene Hilfsmittel:	handschriftliche Notizen, 1 DIN A4 Blatt, beidseitig beschrieben, keine Kopie

Lernergebnisse und Inhalte

Lernergebnisse:

Die Studierenden sind in der Lage,

- die Informationsdarstellung im Rechner zu beschreiben
- einfache Algorithmen und Datenstrukturen wiederzugeben
- die Vorgehensweise beim Übersetzen von Programmen zu erläutern
- einfache Automaten und Sprachen zu definieren
- die wichtigsten Schritte bei der Software-Entwicklung zu erläutern
- die wirtschaftlichen und gesellschaftlichen Auswirkungen der Informatik zu diskutieren

Lehrinhalte:

- Grundbegriffe der Informatik; Nachricht und Information, Codierung; Zahlensysteme
- Einführung in Algorithmen und Datenstrukturen
- Grundlagen der Automatentheorie und formaler Sprachen
- Grundlagen der Softwareentwicklung
- Wirtschaftliche und gesellschaftliche Verantwortung der Informatik

Literatur:

- H.P. Gumm, M. Sommer: Einführung in die Informatik, 10. Auflage, Oldenburg, 2012



GEB1104 Modellierung und Animation 1

Allgemeines

Modulverantwortliche(r):	Hr. André Kettner
Dozent(en):	Hr. André Kettner
Modultyp:	Pflichtmodul
Voraussetzungen:	keine
Verwendbarkeit:	Studiengang Informatik - Game Engineering (B.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS betreutes Praktikum in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Seminaristischer Unterricht 30 Stunden Präsenzzeit Praktikum 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Schriftliche Prüfung 90 Minuten am Ende des Semesters Teilnahmepflicht im Praktikum, Leistungsnachweise im Praktikum Leistungsnachweise sind Zulassungsvoraussetzungen
Zur Prüfung zugelassene Hilfsmittel:	handschriftliche Notizen, 1 DIN A4 Blatt, beidseitig beschrieben, keine Kopie

Lernergebnisse und Inhalte

Lernergebnisse:

- Die Studierenden sind in der Lage, 3D-Modelle mit einem gängigen Modellierungsprogramm (etwa Maya oder Blender) zu erstellen.
- Die Studierenden sind in der Lage, die Qualität von 3D-Modellen beurteilen zu können und diese hinsichtlich geometrischer und topologischer Anforderungen zu optimieren.
- Die Studierenden sind in der Lage, Beleuchtungs-Setups, Materialien und Texturen für 3D-Modelle zu erstellen sowie damit verbundene Effekte bezüglich ihrer Echtzeitfähigkeit zu beurteilen und dahingehend zu optimieren.
- Die Studierenden verstehen kollaborativ an virtuellen Szenen zu arbeiten und kennen gängige Austauschformate.
- Die Studierenden können 3D-Animationen erstellen, welche als Actions in externe Engines importierbar sind.

Lehrinhalte:

- Grundlegende Prinzipien der Modellierung, Materialerstellung, Beleuchtung, Animation und des Renderns virtueller Szenen
- Abbildung von Transformationen, lokaler und globaler Koordinatensysteme sowie Definition und Kontrolle der virtuellen Kamera
- Übersicht über wichtige Modellierungstools und Modifier (z.B. enthalten in Maya, 3D Studio Max und Blender)

- Modellierung von polygonalen Netzen sowie mittels parametrischer Kurven und Flächen (Splines, NURBS)
- Gängige Lichtquellentypen und Beleuchtungsmodelle sowie die Erzeugung von realistischen Schatten für Echtzeitanwendungen
- Materialdefinition und physikalische Effekte wie Spiegelung, Refraktion, Transluzenz und Dispersion
- Texturierung von Modellen, UV mapping und prozedurale Texturen
- Überblick über erweiterte Material- und Beleuchtungstechniken wie IES, HDRI, IBL und Skydomes
- Keyframe-Animation, Interpolation von Transformationen, Gimbal Lock und Pfadanimation
- Erstellung von Animationsabläufen mit animation curves und Actions
- Animation von Charakteren mittels kinematischer Modelle (Rigging, Skinning)

Literatur:

- Akenine-Möller et al.: Real-Time Rendering
- Rick Parent: Computer Animation
- Jeremy Birn: Digital Lighting and Rendering



GEB1105 Programmieren 1

Allgemeines

Im Modul Programmieren 1 erlangen die Studierenden die Grundkenntnisse im Programmieren mit C und C++ in einer aktuellen Entwicklungsumgebung. Es sind keine Vorkenntnisse nötig.

Modulverantwortliche(r):	Prof. Dr. Christoph Bichlmeier
Dozent(en):	Prof. Dr. Christoph Bichlmeier / Prof. Dr. Tobias Breiner
Modultyp:	Pflichtmodul
Voraussetzungen:	Keine
Verwendbarkeit:	Studiengang Informatik - Game Engineering (B.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	4 SWS Seminaristischer Unterricht 4 SWS Übungen (verpflichtend)
Leistungspunkte:	10
Arbeitsaufwand:	60 Stunden Präsenzzeit Unterricht 60 Stunden Präsenzzeit Praktikum 180 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Schriftliche Prüfung 120 Minuten am Ende des Semesters Teilnahmepflicht im Praktikum, Leistungsnachweise im Praktikum Leistungsnachweise sind Zulassungsvoraussetzungen
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

- Die Studierenden sind in der Lage eine IDE (Integrated Development Environment) zum Programmieren, übersetzen und zur Fehlersuche zu benutzen.
- Die Studierenden können kleine Probleme der realen Welt mit strukturierten, modularen Programmen lösen.

Lehrinhalte:

Nach erfolgreicher Teilnahme am Fach Programmieren 1 für Games können Studierende

- Grundlegende und strukturierte Datentypen von C kennen und anwenden
- Kontrollstrukturen und logische Bedingungen in C kennen und anwenden
- Felder, Zeiger und dynamisches Speichermanagement in C kennen und anwenden
- C-Funktionen und -Prozeduren kennen und anwenden
- Integrated Development Environment, Präprozessor, Compiler und Linker kennen und anwenden
- Grundlegende Konzepte der Objektorientierten Programmierung am Beispiel von C++ kennen und anwenden
- Modulare Programme schreiben und die "toolchain" der Entwicklung von C/C++-Pro-

grammen und -Bibliotheken verstehen und anwenden

Literatur:

- M. Dausmann, U. Bröckl, J. Goll: C als erste Programmiersprache, Teubner, 2005
- Herrmann: Grundkurs C++ in Beispielen, Vieweg, 2004
- Peter Prinz und Ulla Kirch-Prinz: C++ lernen und professionell anwenden, 2015
- Kernighan, Ritchie: Programmieren in C
- S. Rieck: OOP für Ingenieure, VDE-Verlag, 2002
- <http://www.cplusplus.com/>

GEB1106 Analysis

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Markus Preisenberger
Dozent(en):	Prof. Dr. Markus Preisenberger / Epple
Modultyp:	Pflichtmodul
Voraussetzungen:	Keine
Verwendbarkeit:	Studiengang Informatik - Game Engineering (B.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	3 SWS Seminaristischer Unterricht 1 SWS Übung
Leistungspunkte:	5
Arbeitsaufwand:	60 Stunden Präsenzzeit Unterricht/Übung 30 Stunden Selbststudium - Betreute Studierzeit 60 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Schriftliche Prüfung 90 Minuten am Ende des Semesters
Zur Prüfung zugelassene Hilfsmittel:	kein Taschenrechner, ansonsten ohne Einschränkung

Lernergebnisse und Inhalte

Lernergebnisse:

Nach erfolgreicher Teilnahme am Modul Analysis können die Studierenden

- Grundbegriffe der Analysis wiedergeben
- wesentliche Berechnungsmethoden der Analysis problemabhängig auswählen
- einfache Berechnungen mit Methoden der Analysis durchführen
- einfache Beweise analysieren und einfache Beweismethoden anwenden

Lehrinhalte:

- Aussagenlogik, Grundbegriffe der Prädikatenlogik
- Natürliche Zahlen und Vollständige Induktion
- Rationale Zahlen, Funktionen, Reelle Zahlen und Körperaxiome,
- Folgen und Reihen, Grenzwerte
- Stetigkeit und stetige Funktionen
- Differential- und Integralrechnung
- Taylor-Entwicklung, Potenzreihen
- Differentialgleichungen: Beispiele

Literatur:

- Brill, M.: Mathematik für Informatiker, Hanser Verlag, 2. Auflage, 2005
- Hachenberger, D.: Mathematik für Informatiker, Pearson Studium, 2. Auflage, 2008
- Hartmann, P.: Mathematik für Informatiker, Springer Vieweg Verlage, 2019
- Stingl, P.: Einstieg in die Mathematik für Fachhochschulen, Hanser Verlag, 2013

-
- Stingl, P.: Mathematik für Fachhochschulen, Hanser Verlag, 2007



GEB1107 Theoretische Informatik

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Elmar Böhler
Dozent(en):	Prof. Dr. Elmar Böhler
Modultyp:	Pflichtmodul
Voraussetzungen:	keine
Verwendbarkeit:	Studiengang Informatik - Game Engineering (B.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übung in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Vorlesung 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Teilnahmepflicht in den Übungen, Schriftliche Prüfung 90 Minuten am Ende des Semesters. Erfolgreiche Teilnahme an den Übungen ist Zulassungsvoraussetzungen
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel, handschriftliche Notizen, 1 DIN A4 Blatt, einseitig beschrieben, keine Kopie

Lernergebnisse und Inhalte

Lernergebnisse:

Die Studierenden kennen die wichtigsten Grundlagen der theoretischen Informatik. Sie beherrschen die Grundbegriffe der Automatentheorie, der formalen Sprachen, der Berechenbarkeit und der Entscheidbarkeit.

Lehrinhalte:

- Formale Sprachen und Grammatik
- Chomsky-Hierarchie
- Reguläre Sprachen und reguläre Ausdrücke
- Automatentheorie (NEA, DEA)
- Thompson-Algorithmus
- Turing-Berechenbarkeit
- Entscheidbarkeit
- Komplexitätstheorie, NP-Vollständigkeit

Literatur:

- Hoffmann, Dirk W.: Theoretische Informatik. Hanser-Verlag, 3. Auflage. 2015
- Schöning, Uwe: Theoretische Informatik - kurz gefasst, 5. Auflage, 2008
- Wagner, Klaus: Theoretische Informatik, 2. Auflage, 2003

GEB1108 Game Design

Allgemeines

Modulverantwortliche(r):	Prof. Dr.-Ing. René Bühling
Dozent(en):	Prof. Dr.-Ing. René Bühling
Modultyp:	Pflichtmodul
Voraussetzungen:	keine
Verwendbarkeit:	Studiengang Informatik - Game Engineering (B.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übung
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Unterricht 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Schriftliche Prüfung 90 Minuten am Ende des Semesters
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

- Game Designs analysieren, verstehen und bewerten
- Essentielle Inhalte und Strukturen von Spieleproduktionen kennen
- Eigene Game Design Documents erstellen

Lehrinhalte:

Die Veranstaltung klärt zunächst Begriff, Ziel und Zweck eines Game Design Documents. Die Studierenden durchlaufen anschließend alle wichtigen Aspekte der Spielekonzeption und setzen die Lerninhalte in einem eigenen Konzeptdokument um:

- Platzierung und Kontext
- Narrativ, Geschichte und Szenerie
- Charaktere und Persönlichkeiten
- Level-Design und -Planung
- Produktmodellierung und Spielmechaniken
- Innovation und Alleinstellungsmerkmale
- Projektorganisation und Ressourcenplanung
- Publikation und Marketing

Literatur:

- Rehfeld, Gunther (2020): Game Design und Produktion. Grundlagen, Anwendungen und Beispiele. München (Hanser). ISBN 978-3-446-46315-8
- Schell, Jesse: Die Kunst des Game Designs : Bessere Games konzipieren und entwickeln. Norderstedt: BoD – Books on Demand, 2020. ISBN 978-3-958-45282-4

GEB1109 Gestaltung und Zeichnen

Allgemeines

Modulverantwortliche(r):	Prof. M. A. Sabine Buchwieser
Dozent(en):	Prof. M. A. Sabine Buchwieser
Modultyp:	Pflichtmodul
Voraussetzungen:	Keine
Verwendbarkeit:	Studiengang Informatik - Game Engineering (B.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	4 SWS Seminaristischer Unterricht mit integrierten praktischen Übungen
Leistungspunkte:	5
Arbeitsaufwand:	60 Stunden Präsenzzeit Unterricht mit Übungen 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Leistungsnachweise: Mappe mit erstellten Zeichnungen; Schriftliche Prüfung, 90 Minuten am Ende des Semesters
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

Die zeichnerischen Fähigkeiten werden soweit geschult, dass Ideen und Vorstellungen treffsicher zeichnerisch dargestellt werden können.
Die Studierenden sind in der Lage, Modellierungsvorlagen, Storyboards oder Character Sheets wirkungsvoll und überzeugend zu visualisieren.

Lehrinhalte:

Techniken zum Erfassen von Motiven
Zeichentechniken, Entwicklung einer individuellen Formensprache
Licht, Schatten, Farbe
Gegenständliches Zeichnen
Räumlich-/ Perspektivisches Zeichnen
Figürliches Zeichnen
Storyboards
Character Sheets

- Darstellung von Körpern
- Perspektive
- Schatten und Licht
- verschiedenste Zeichentechniken
- Mimik
- Proportionen
- Storyboard

Literatur:

GEB1110 Programmieren 2

Allgemeines

Programmieren 2 setzt grundlegende Programmierkenntnisse aus der Erstsemestervorlesung Programmieren 1 voraus. In dieser Veranstaltung werden erste Programmiermuster, einfache Algorithmen, GUI Programmierung, erweiterte Konzepte der OOP sowie zusätzliche Programmierkonzepte wie generische Programmierung und Operatorüberladung in C++ vermittelt und praktisch umgesetzt.

Modulverantwortliche(r):	Prof. Dr. Elmar Böhler
Dozent(en):	Prof. Dr. Elmar Böhler / Prof. Dr. Christoph Bichlmeier
Modultyp:	Pflichtmodul
Voraussetzungen:	Programmieren 1
Verwendbarkeit:	Studiengang Informatik (B.Sc.), Studiengang Informatik - Game Engineering (B.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS betreutes Praktikum in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Unterricht 30 Stunden Präsenzzeit Praktikum 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Schriftliche Prüfung 90 Minuten am Ende des Semesters
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

- Die Studierenden sind in der Lage, bei der Lösung von Programmieraufgaben strukturiert vorzugehen.
- Die Studierenden können kleine Probleme der realen Welt (gegebenenfalls mit Hilfe von Standard-Bibliotheken und Frameworks) lösen.

Lehrinhalte:

- Grundlegende Schritte des Software-Entwicklungsprozesses (Analyse, Design, Implementierung, Test)
- Basis-Einführung in Entwurfsmuster
- Einsatz von Standard-Bibliotheken am Beispiel der C++ Standard Library
- Grundlagen der Programmierung von Graphischen User Interfaces (GUIs)
- Einsatz von Frameworks
- Festigung des Wissens zur OOP
- Templates in C++
- Operator Überladung in C++

Literatur:



-
- Peter Prinz und Ulla Kirch-Prinz: C++ lernen und professionell anwenden, 2015
 - .NET Basisbibliothek (BCL). Das Referenzhandbuch, Microsoft Press, 2004
 - Gamma, Helm, Johnson, Vlissides: Entwurfsmuster. Elemente wiederverwendbarer Software, Addison-Wesley, München, 2014
 - <http://www.cplusplus.com/>



GEB1111 Algorithmen und Datenstrukturen

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Ulrich Göhner
Dozent(en):	Prof. Dr. Ulrich Göhner
Modultyp:	Pflichtmodul
Voraussetzungen:	keine
Verwendbarkeit:	Studiengang Informatik (B.Sc.), Studiengang Informatik - Game Engineering (B.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übung in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Unterricht 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Teilnahmepflicht in den Übungen, Schriftliche Prüfung 90 Minuten am Ende des Semesters. Erfolgreiche Teilnahme an den Übungen ist Zulassungsvoraussetzung
Zur Prüfung zugelassene Hilfsmittel:	handschriftliche Notizen, 1 DIN A4 Blatt, beidseitig beschrieben, keine Kopie

Lernergebnisse und Inhalte

Lernergebnisse:

Die Studierenden haben einen Überblick über wichtige Algorithmen und Datenstrukturen in der Informatik. Die Studierenden sind in der Lage, die Aufwände der Algorithmen zu quantifizieren und in Komplexitätsmaße zu fassen. Sie können eine gegebene Problemstellung in einen Algorithmus fassen und diesen in ein effizient laufendes Programm überführen.

Lehrinhalte:

- Definition und Darstellung Algorithmus
- Komplexität und O-Notation
- Greedy-Algorithmen, Rekursion, Divide and Conquer-Algorithmen
- Definition Datenstruktur / ADT
- verkettete Liste, Stapel und Schlangen
- Tabelle mit Zugriffsoperationen und Implementierungen
- elementare Sortierverfahren (Insertion-, Selection-, ExchangeSort)
- schnelle Sortierverfahren (Quicksort, Mergesort, Heapsort, Shellsort, Combsort)
- Baumstrukturen (Begriffsbildung, Suchbaum, Heap, Treap, ausgeglichene Bäume)
- Hashfunktion
- Graphen (Definition, Darstellung, Implementierung, Breiten/Tiefensuche, reflexive transitive Hülle, Kürzeste Wege)

Literatur:

- Sedgewick, R.; Wayne, K.: "Algorithmen", 4. Auflage, Pearson, 2014.
- Saake, G.; Sattler, K.-U.: "Algorithmen und Datenstrukturen", 5. Auflage, dpunkt.verlag, Heidelberg, 2014.
- Reeß, H.; Viebeck, G.: "Datenstrukturen und Algorithmen (C++)", Carl Hanser Verlag, 2002.
- Heun, V.: "Grundlegende Algorithmen (C)", Vieweg Verlag, 2. Auflage, 2003.
- Dietzfelbinger, M.; Mehlhorn, K., Sanders, P.: "Algorithmen und Datenstrukturen", Springer, Berlin Heidelberg, 2014.
- Edmonds: How to think about Algorithms, Cambridge, 2008

GEB1112 Diskrete Mathematik für Games

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Markus Preisenberger
Dozent(en):	Prof. Dr. Markus Preisenberger / Epple
Modultyp:	Pflichtmodul
Voraussetzungen:	Lineare Algebra
Verwendbarkeit:	Studiengang Informatik - Game Engineering (B.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übung
Leistungspunkte:	5
Arbeitsaufwand:	60 Stunden Präsenzzeit Unterricht/Übung 30 Stunden Selbststudium - Betreute Studierzeit 60 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Schriftliche Prüfung 90 Minuten am Ende des Semesters
Zur Prüfung zugelassene Hilfsmittel:	kein Taschenrechner, ansonsten ohne Einschränkung

Lernergebnisse und Inhalte

Lernergebnisse:

Nach erfolgreicher Teilnahme am Modul Diskrete Mathematik können die Studierenden

- wesentliche Begriffe der Diskreten Mathematik wiedergeben
- wesentliche Berechnungsmethoden der Diskrete Mathematik problemabhängig auswählen
- Berechnungen mit Methoden der Diskrete Mathematik durchführen

Lehrinhalte:

- Grundlegende Begriffe der Diskreten Mathematik
- Abzählmethoden
- Graphen, Bäume, aufspannende Bäume, kürzeste Wege, Euler- und Hamilton-Kreise
- Algorithmen von Kruskal und Dijkstra, MST-Heuristik
- Ringe: Matrizenring, Polynomring
- Körper: Endliche Körper, modulare Arithmetik, Komplexe Zahlen
- Algorithmische Geometrie

Literatur:

- Aigner, M.: Diskrete Mathematik, Vieweg Verlag, 2007
- Aigner, M.: Zahlentheorie : eine Einführung mit Übungen, Hinweisen und Lösungen, Vieweg-Teubner, 2012
- Brill, M.: Mathematik für Informatiker, Hanser Verlag, 2. Auflage, 2005
- Hachenberger, D.: Mathematik für Informatiker, Pearson Studium, 2. Auflage, 2008



-
- Hartmann, P.: Mathematik für Informatiker, Springer Vieweg Verlag, 2019
 - Klein, R.: Algorithmische Geometrie, Springer Verlag, 2. Auflage, 2004
 - Stingl, P.: Operations Research : Lineartoptimierung, Hanser Verlag, 2002
 - Stingl, P.: Mathematik für Fachhochschulen, Hanser Verlag, 2007



GEB1113 Datenbanken

Allgemeines

Die Studierenden haben einen Überblick über den Einsatz von Datenbanken und Informationssystemen. Sie kennen die Grundlagen und die praktische Anwendung von Datenmodellierung und Datenbankentwurf. Die Studierenden können Datenbanken mit SQL erstellen, die Daten manipulieren und komplexe Abfragen durchführen. Sie sind in der Lage Ausführungspläne zu analysieren und Optimierungen am physischen Entwurf vorzunehmen. Außerdem können sie transaktionsorientierte Anwendungen erstellen.

Modulverantwortliche(r):	Prof. Nikolaus Steger
Dozent(en):	Prof. Nikolaus Steger
Modultyp:	Pflichtmodul
Voraussetzungen:	keine
Verwendbarkeit:	Studiengang Informatik (B.Sc.), Studiengang Informatik - Game Engineering (B.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übung in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Vorlesung 30 Stunden Präsenzzeit Übung 90 Selbststudium
Leistungsnachweis und Prüfung:	Schriftliche Prüfung 90 Minuten am Ende des Semesters
Zur Prüfung zugelassene Hilfsmittel:	handschriftliche Notizen, 1 DIN A4 Blatt, beidseitig beschrieben, keine Kopie

Lernergebnisse und Inhalte

Lernergebnisse:

Nach erfolgreicher Beendigung der Veranstaltung sind die Studierenden in der Lage

- Datenmodelle aus gegebenen Anforderungen zu erstellen
- Einen Logischen Datenbankentwurf und einen Physischen Datenbankentwurf aus einem gegebenen Datenmodell zu entwerfen
- Daten in Datenbanken mit SQL zu manipulieren
- Komplexe Abfragen in Relationaler Algebra und in SQL auf einem gegebenen Datenbankschema zu erstellen
- SQL-Abfragen in Relationale Algebra zu übersetzen und mittels Heuristischer Optimierung zu optimieren
- Zugriffspläne zu Abfragen zu analysieren und daraus physische Optimierungen abzuleiten und umzusetzen
- Ein Datenbankschema mit gegebenen Funktionalen Abhängigkeiten bis zur Boyce-Codd-Normalform zu normalisieren
- Transaktionsorientierte Anwendungen mit SQL zu implementieren

Lehrinhalte:

- Grundlagen Datenbanken und Informationssysteme
- Entity Relationship-Datenmodelle und Datenmodellierung mit UML
- Das Relationale Datenmodell
- Relationale Algebra
- Datenbankentwurf
- SQL (DDL, DML und DCL)
- Datenbankzugriff aus Programmiersprachen
- Implementierung der Relationalen Algebra
- Query-Übersetzung und Optimierung
- Relationale Entwurfstheorie, Normalformen
- Das ACID-Prinzip

Literatur:

- Kemper, Eickler: Datenbanksysteme: Eine Einführung, 10. Auflage 2015, Oldenburg,
- Elmasri, Navathe: Fundamentals of Database Systems, Pearson, 7th Edition 2016, Pearson
- Garcia-Molina, Widom: Database Systems The Complete Book, Second Edition 2013, Pearson



GEB1114 Betriebssysteme

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Arnulf Deinzer
Dozent(en):	Prof. Dr. Arnulf Deinzer
Modultyp:	Pflichtmodul
Voraussetzungen:	keine
Verwendbarkeit:	Studiengang Informatik (B.Sc.), Studiengang Informatik - Game Engineering (B.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übungen/Praktikum in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Vorlesung 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Pflicht zur aktiven Teilnahme in Übungen/Praktikum, Schriftliche Prüfung 90 Minuten am Ende des Semesters.
Zur Prüfung zugelassene Hilfsmittel:	handschriftliche Notizen, 1 DIN A4 Blatt, beidseitig beschrieben, keine Kopie, nicht programmierbarer Taschenrechner

Lernergebnisse und Inhalte

Lernergebnisse:

Die Studierenden haben fundierte Kenntnisse aus Aufgaben und Realisierungen von Betriebssystemen. Sie sind sicher im Umgang mit Linux auf Nutzerniveau und sind nach erfolgreicher Beendigung der Veranstaltung in der Lage, in einem experimentellen Betriebssystem einfache Aufgaben aus Scheduling und Synchronisation von Prozessen umzusetzen.

Im Einzelnen sind die Studierenden nach erfolgreicher Beendigung der Veranstaltung in der Lage

- die gebräuchlichsten Verfahren für die Organisation von Sekundärspeichern darzustellen und die entsprechenden Verwaltungsaufwände zu berechnen
- verschiedene Schedulingstrategien darzustellen und unter Verwendung von Standardmetriken zu vergleichen
- die gebräuchlichsten Verfahren für die Organisation von Primärspeichern darzustellen und die entsprechenden Verwaltungsaufwände zu berechnen
- sicher mit grundlegenden Synchronisationsprimitives wie Semaphoren umzugehen, d.h. ein gegebenes Synchronisationsproblem mit Semaphoren zu lösen bzw. problematische Implementierungen zu verbessern
- Strategien zur Deadlockerkennung und -vermeidung darzustellen und in Vor- und Nachteilen zu vergleichen

Lehrinhalte:

- Einführung und Überblick
- Dateisysteme, IO-Devices, Primär-, Sekundär- und Tertiärspeicher
- Prozess- und Prozessorverwaltung
- Primärspeicherverwaltung
- Prozesskommunikation
- Sicherheit
- Kommandosprachen

Literatur:

- Tanenbaum, Andrew S.: Moderne Betriebssysteme, Prentice-Hall 2002
- Stallings, William: Betriebssysteme, 4. Auflage, 2003 Pearson
- Herrtwich R.G., Hommel G.: Kooperation und Konkurrenz, Springer 1989



GEB1115 Computergrafik

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Bernd Dreier
Dozent(en):	Prof. Dr. Bernd Dreier
Modultyp:	Pflichtmodul
Voraussetzungen:	Keine
Verwendbarkeit:	Studiengang Informatik - Game Engineering (B.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS betreutes Praktikum in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Vorlesung 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Teilnahmepflicht im Praktikum, Leistungsnachweise im Praktikum, 90 minütige schriftliche Prüfung am Ende des Semesters, Leistungsnachweise sind Zulassungsvoraussetzungen.
Zur Prüfung zugelassene Hilfsmittel:	handschriftliche Notizen, 1 DIN A4 Blatt, beidseitig beschrieben, keine Kopie

Lernergebnisse und Inhalte

Lernergebnisse:

- Die Studierenden kennen und verstehen die Abläufe, die für heutige 3D-Grafik in Echtzeit erforderlich sind.
- Die Studierenden verstehen die OpenGL-Pipeline und können 3D-Grafik-Programme auf der Basis von C++ und OpenGL 4.x mit dem Framework Qt konzipieren und implementieren.
- Die Studierenden verstehen die Konzepte der neueren OpenGL-Versionen inklusive Shaderprogrammierung und können diese anwenden.
- Die Studierende können problemorientiert passende Datenstrukturen zur Speicherung von polygonalen Netzen auswählen und umsetzen.
- Die Studierenden verstehen die Anforderungen realitätsnaher Computergrafik und können diese im Hinblick auf Beleuchtung, Texturierung und andere Mappingverfahren anwenden.

Lehrinhalte:

- Computergrafik-Pipeline, Koordinatensysteme, affine Transformationen, Kameramodell sowie deren Anwendung in OpenGL
- Quaternionen zur Behandlung von Orientierung/Rotation
- Culling und Clipping
- Verdeckungsrechnung und Rasterisierung
- Einführung in Shaderprogrammierung

-
- Grundlagen polygonaler Netze und passender Datenstrukturen
 - Lokale Beleuchtungsmodelle, Texturen und andere Mappingverfahren sowie deren Abbildung in OpenGL

Literatur:

- Edward Angel, Dave Shreiner: Interactive Computer Graphics, Addison-Wesley, 2011

GEB1116 Software Engineering / Softwaretechnik 1

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Georg Hagel
Dozent(en):	Prof. Dr. Georg Hagel
Modultyp:	Pflichtmodul
Voraussetzungen:	keine
Verwendbarkeit:	Studiengang Informatik (B.Sc.), Studiengang Informatik - Game Engineering (B.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übungen/Praktikum in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Vorlesung 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Teilnahmepflicht in der Übung, Leistungsnachweise in der Übung, schriftliche Prüfung 90 Minuten am Ende des Semesters. Leistungsnachweise sind Zulassungsvoraussetzungen.
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

Nach erfolgreicher Beendigung der Veranstaltung sind die Studierenden in der Lage,

- verschiedene Vorgehensmodelle mit ihren Stärken und Schwächen zu beschreiben.
- die UML in ihrer aktuellen Version zur Beschreibung von Ergebnissen in Analyse, Architektur und Design anzuwenden.
- alle Phasen der Softwareerstellung (Requirements Engineering, Analyse, Architektur und Design, Implementierung und Qualitätssicherung) zu beschreiben.
- Bekannte Muster in Analyse und Entwurf anzuwenden.
- Testfallermittlung und Metriken auf gegebene Problemstellungen anzuwenden.

Lehrinhalte:

- Vorgehensmodelle
- Modellierung mit Strukturdiagrammen
- Modellierung mit Verhaltensdiagrammen
- Modellierung mit Architekturdiagrammen
- Modellierung mit Interaktionsdiagrammen
- Requirements Engineering
- Analyse und Analysemuster
- Architekturbeschreibung

- Design-Beschreibung und Design-Muster
- Qualitätssicherung
- Ethik in der Informatik

Literatur:

- Sommerville, Ian: Software Engineering, Pearson Studium, 10. Auflage (2018)
- Van Vliet, Hans: Software Engineering - Principles and Practice, Wiley, 3. Aufl. (2008)
- Baltzert, Helmut: Lehrbuch der Softwaretechnik - Basiskonzepte und Requirements Engineering, Spektrum Verlag, 3. Aufl. (2009)
- Baltzert, Helmut: Lehrbuch der Softwaretechnik - Entwurf, Implementierung, Installation und Betrieb, Spektrum Verlag, 3. Aufl. (2011)
- Braude, Eric J.: Software Engineering - Modern Approaches, Wiley, 2. Aufl. (2016)



GEB1117 Human Computer Interaction / Softwaretechnik 2

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Bernd Dreier
Dozent(en):	Prof. Dr. Bernd Dreier
Modultyp:	Pflichtmodul
Voraussetzungen:	keine
Verwendbarkeit:	Studiengang Informatik (B.Sc.), Studiengang Informatik - Game Engineering (B.Sc.), Studiengang Gesundheits- und Pflegeinformatik (B.Sc.)
Angebot und Dauer:	Wintersemester und Sommersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS betreutes Praktikum in kleinen Gruppen mit Erstellung einer Studienarbeit
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Unterricht 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Teilnahmepflicht im Praktikum, der endnotenbildende Leistungsnachweis besteht aus einer benoteten Studienarbeit (30-50 Seiten).
Zur Prüfung zugelassene Hilfsmittel:	

Lernergebnisse und Inhalte

Lernergebnisse:

- Die Studierenden verstehen die Bedeutung der Human Computer Interaction und ihre Ziele.
- Die Studierenden kennen aktuelle Methoden zur benutzerzentrierten Softwareentwicklung (Usability Engineering) und können diese anwenden.
- Die Studierenden verstehen die relevanten Grundlagen der menschlichen Physiologie und Psychologie und können diese bei der Gestaltung von Benutzeroberflächen anwenden.
- Die Studierenden kennen die relevanten objektiven Kriterien (Normen und Richtlinien) und können diese anwenden.

Lehrinhalte:

- Begriffsdefinition und Einführung von Human Computer Interaction, Interaktionsdesign, Usability (Engineering)
- Verständnis und Verflechtung der Prozesse Analyse, Entwurf, Evaluation, Design
- Einführung in Usability Engineering
- Relevante Teile der Physiologie und Psychologie des Menschen sowie abgeleitete Modelle und Verfahren
- Objektive Kriterien zur Gestaltung und Beurteilung von User Interfaces, Richtlinien und Normen, insbesondere DIN EN ISO 9241



Literatur:

- David Benyon, Designing Interactive Systems, 3rd edition, 2014, Pearson
- Richter, Flückiger, Usability Engineering kompakt, 3. Auflage, 2013
- Norm DIN EN ISO 9241 in der aktuellen Fassung

GEB1118 Wahrscheinlichkeitsrechnung und Numerik

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Jochen Staudacher
Dozent(en):	Prof. Dr. Jochen Staudacher
Modultyp:	Pflichtmodul
Voraussetzungen:	keine
Verwendbarkeit:	Studiengang Informatik - Game Engineering (B.Sc.), Studiengang Informatik (B.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übung in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Unterricht 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Schriftliche Prüfung 90 min am Ende des Semesters
Zur Prüfung zugelassene Hilfsmittel:	kein Taschenrechner, ansonsten ohne Einschränkung

Lernergebnisse und Inhalte

Lernergebnisse:

Nach erfolgreicher Beendigung der Veranstaltung können die Studierenden

- wichtige Grundbegriffe der Wahrscheinlichkeitsrechnung und der Numerik definieren
- wichtige Verteilungen auf einfache fachspezifische Fragestellungen anwenden
- die Funktionsweise zentraler numerischer Algorithmen beschreiben
- Methoden zur Beurteilung der Komplexität, Stabilität und Genauigkeit numerischer Algorithmen in einfachen Fällen einsetzen

Lehrinhalte:

- Grundbegriffe der Wahrscheinlichkeitsrechnung
- Zufallsexperiment und Wahrscheinlichkeit
- Zufallsvariable, diskrete und stetige Verteilungen; Grenzwertsätze
- Kombinatorik
- Hypothesentest
- Zahldarstellung und Fehlerrechnung
- Stabilität und Komplexität numerischer Algorithmen
- Grundbegriffe der Numerischen Linearen Algebra
- Approximation, Interpolation und numerische Integration
- Iterative Verfahren zur Bestimmung von Nullstellen und Fixpunkten

Literatur:

-
- W. Dürr, H. Mayer: Wahrscheinlichkeitsrechnung und Statistik, Hanser Verlag, 8. Auflage, 2018
 - M. Sachs: Wahrscheinlichkeitsrechnung und Statistik, Hanser Fachbuchverlag, 5. Auflage, 2018
 - G. Fischer: Stochastik einmal anders: Parallel geschrieben mit Beispielen und Fakten, vertieft durch Erläuterungen, Vieweg+Teubner, 1. Auflage, 2005
 - Thomas Huckle, Stefan Schneider: Numerische Methoden: Eine Einführung für Informatiker, Naturwissenschaftler, Ingenieure und Mathematiker, Springer, 2. Auflage, 2006.
 - Lloyd N. Trefethen, David Bau III: Numerical Linear Algebra, SIAM, 1997.
 - Günther Hämmerlin, Karl-Heinz Hoffmann: Numerische Mathematik, Springer, 4. Auflage, 1994.
 - Jochen Werner: Numerische Mathematik 1, Vieweg, 1992.
 - Endre Süli, David Mayers: An Introduction To Numerical Analysis, Cambridge University Press, 2003.



GEB1120 Game Engineering

Allgemeines

Im Fach Game Engineering werden alle wesentlichen technischen Elemente eines Computerspiels vorgestellt und gemeinsam in einem Beispiellevel entwickelt. Für die Entwicklung wird beispielhaft einer Game Engine verwendet, die bereits Basisfunktionalität bietet.

Modulverantwortliche(r):	Prof. Dr.-Ing. René Bühling
Dozent(en):	Prof. Dr.-Ing. René Bühling / Prof. Dr. Tobias Breiner
Modultyp:	Pflichtmodul
Voraussetzungen:	Programmierkenntnisse
Verwendbarkeit:	Studiengang Informatik - Game Engineering (B.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übungen (verpflichtend)
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Unterricht 30 Stunden Präsenzzeit Übungen 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	schriftl. Prüfung 90 Minuten am Ende des Semesters.
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

Nach erfolgreicher Teilnahme an der Veranstaltung sind die Studierenden in der Lage ein 3D Game mithilfe einer aktuellen Game Engine eigenständig zu entwickeln. Demnach kennen die Studierenden die Funktionsmodule der Engine für alle wichtigen Komponenten eines Spiels und können diese anwenden. Dies sind im Wesentlichen: Benutzeroberflächen, Beleuchtung und Farbgebung, Kameraführung, physikalische Effekte, grundlegende Methoden zur Steuerung und Animation von Spieler und optional NPCs, Soundeinbindung, Organisation von Szenen bzw. Levels, und optional Schnittstellen zu Hardware wie Mixed Reality Brillen oder Cave.

Lehrinhalte:

- Anwendung der API und ggf. Benutzeroberfläche einer aktuellen Game Engine
- Aufbau von Szenen und Entwicklung einzelner Komponenten wie Kamerasteuerung, Physik, Navigation, Beleuchtung, GUI
- Design Patterns und Kombination/Zusammenspiel der einzelnen Komponenten

Literatur:

Dokumentation der jeweiligen Game Engine, z.B.

- www.vektoria-engine.com
- Unity



GEB1121 Ansteuerung virtueller Charaktere

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Klaus Ulhaas
Dozent(en):	Prof. Dr. Klaus Ulhaas
Modultyp:	Pflichtmodul
Voraussetzungen:	Computergrafik
Verwendbarkeit:	Studiengang Informatik - Game Engineering (B.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übungen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Unterricht 30 Stunden Präsenzzeit Übungen 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Schriftl. Prüfung 90 Minuten am Ende des Semesters.
Zur Prüfung zugelassene Hilfsmittel:	nicht programmierbarer Taschenrechner

Lernergebnisse und Inhalte

Lernergebnisse:

Nach erfolgreicher Teilnahme an der Lehrveranstaltung sind die Studierenden in der Lage ...

- Motion-Capture Daten auf einen virtuellen Character zu übertragen und in der Unreal Engine zu visualisieren.
- virtuelle Charaktere über ein Skelettmodell und Morph-Targets (Blend Shapes) zu animieren und diese Animationen mit C++ und GLSL zu programmieren.
- bei einem Auszug eines Collada-Formats die Zusammenhänge zur Character-Animation und Visualisierung zu erkennen.
- ein optisches Motion Capture System richtig aufzubauen, zu kalibrieren und zu bedienen.
- die technische Arbeitsweise eines Motion Capture Systems zu erklären und zu verstehen.
- die Vor- und Nachteile verschiedener Motion-Capture-Formate zu benennen und zu erklären.
- Techniken und Algorithmen zur inversen Kinematik anzuwenden und Vor- und Nachteile verschiedener Verfahren zu erkennen und zu benennen.
- das Animationsverhalten Virtueller Charaktere über State Machines in der Unreal Engine festzulegen.

Lehrinhalte:

- Einführung
- Arten der Character-Animation
- Animation Blending und Character Control mit der Unreal Engine 4

- Motion Capture Systeme (Technologien und Grundlagen)
- Funktionsweise optischer MoCap-Systeme
- Vorbereitung und Ablauf einer MoCap-Sitzung
- Motion Capture Datentypen und Dateiformate
- Kinematische Animation und Motion Blending
- Inverse Kinematik Techniken

Literatur:

Literatur und Empfehlungen zu Einzelthemen werden fortlaufend in der Vorlesung bekanntgegeben

- Alberto Menache. Understanding motion capture for computer animation, Morgan Kaufmann, Second Edition, 2011
- Ramakrishnan Mukundan. Advanced Methods in Computer Graphics With examples in OpenGL, Springer Verlag, London, 2012



GEB1122 Verteilte Softwaresysteme

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Stefan Frenz
Dozent(en):	Prof. Dr. Stefan Frenz
Modultyp:	Pflichtmodul
Voraussetzungen:	keine
Verwendbarkeit:	Studiengang Informatik (B.Sc.), Studiengang Informatik - Game Engineering (B.Sc.), Studiengang Wirtschaftsinformatik (B.Sc.), Studiengang Gesundheits- und Pflegeinformatik (B.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übungen/Praktikum in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Unterricht 30 Stunden Präsenzzeit Übungen/Praktikum 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	schriftl. Prüfung 90 Minuten am Ende des Semesters.
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

Die Studierenden verstehen die grundlegenden Herausforderungen an und Möglichkeiten von softwareseitig verteilten Systemen. Sie lernen das OSI-7-Modell, Client-/Server- und Peer-To-Peer-Architekturen sowie die erforderlichen theoretischen Hintergründe kennen. Die Studierenden implementieren verteilte Anwendungen über UDP, TCP und RMI.

Lehrinhalte:

- Kommunikationsgrundlagen
- Architektur verteilter Systeme
- Implementierung verteilter Anwendungen
- Sicherheitsaspekte verteilter Anwendungen

Literatur:

- Tanenbaum, Andrew S.; van Steen, Marten: Verteilte Systeme (2. Auflage), Pearson 2007
- Coulouris, George et al: Verteilte Systeme (3. Auflage), Pearson 2005
- Bengel, Günther: Grundkurs Verteilte Systeme (3. Auflage), vieweg 2004



GEB1123 Projektmanagement

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Peter Klutke
Dozent(en):	Prof. Dr. Peter Klutke
Modultyp:	Pflichtmodul
Voraussetzungen:	keine
Verwendbarkeit:	Studiengang Informatik (B.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übung in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Vorlesung 30 Stunden Präsenzzeit Übungen 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Eine 90 minütige schriftliche Prüfung, die am Ende des Semesters erfolgt.
Zur Prüfung zugelassene Hilfsmittel:	Skript (Ausdruck mit eigenen Notizen), nicht programmierbarer Taschenrechner

Lernergebnisse und Inhalte

Lernergebnisse:

Nach erfolgreicher Beendigung der Veranstaltung sind die Studierenden in der Lage, . . .

- die Bedeutung von Projektmanagement zu erläutern,
- Projektmanagement-Standards einzusetzen,
- Abhängigkeiten im Umfeld eines Projektes zu analysieren,
- den Ablauf eines Projektes und die zugehörigen Tätigkeiten des Projektmanagements miteinander zu kombinieren,
- wesentliche Techniken des Projektmanagements sicher situativ anzuwenden,
- vorausschauende, proaktive Tätigkeiten im Projektmanagement sicher einzusetzen und
- gesellschaftliche Auswirkungen und Sozialverträglichkeit von Lösungen und Innovationen im Projektmanagement besser zu beurteilen, etwa beim ganzheitlichen Projektmanagement, im Personalmanagement (Überstunden, Burnout) oder im gegenseitigen Umgang ("Klima") im Projekt.

Lehrinhalte:

- Motivation und Definitionen für das Projektmanagement
- Projektmanagementstandard PMBoK mit Projektphasen und Wissensgebieten
- Organisationsformen und Aufgabenbereich des Projektleiters
- Problemfeldanalyse, u.a. mit Nutzwertanalyse, Marginalrendite, Balanced Scorecard
- Projektinitiierung, IT-Projektdefinition und Projektstrukturplan
- Netzplantechnik (CPM und MPM) und Einsatzmittelplanung

- Kosten-, Kommunikations- und Personalmanagement
- Risikomanagement mit Wahrscheinlichkeitsbäumen
- Projektüberwachung, Projektsteuerung und Projektabschluss
- Tailoring und Zusammenarbeit im Projektmanagement
- Projektmanagement und Vorgehensmodelle; SCRUM-Einführung

Literatur:

- Burghardt, Manfred: "Projektmanagement: Leitfaden für die Planung, Überwachung und Steuerung von Projekten"; Verlag: Publicis; Auflage: 10. überarb. u. erw. (10. Januar 2018); ISBN-13: 978-3895784729
- Project Management Institute: "A guide to the Project Management Body of Knowledge (PMBOK Guide) (German version)"; Verlag: The Stationery Office Ltd; Auflage: 6th ed., 2017 (30. Januar 2018); ISBN-13: 978-1628251883
- Timinger, Holger : „Modernes Projektmanagement: Mit traditionellem, agilem und hybridem Vorgehen zum Erfolg", Wiley-VCH; 1. Edition (12. Juli 2017), ISBN-10: 3527530487, ISBN-13: 978-3527530489

GEB5200 GE-Lab

Allgemeines

GE-Lab ist ein anwendungsbezogenes Modul, in dem die Studierenden den Game Engineering-Prozess auf Anwendungsebene durchlaufen. In kleinen Gruppen werden Games oder Simulationen mit Hilfe von Game Engines erstellt.

Modulverantwortliche(r):	Alle Professoren der Fakultät
Dozent(en):	Alle Professoren der Fakultät
Modultyp:	Pflichtmodul
Voraussetzungen:	
Verwendbarkeit:	Studiengang Informatik - Game Engineering (B.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	Projektbesprechungen, agile Entwicklungsmethodik zur Teamorganisation
Leistungspunkte:	5
Arbeitsaufwand:	15 Stunden Präsenzzeit 135 Stunden selbstständiges Arbeiten
Leistungsnachweis und Prüfung:	Projektbericht Präsentation

Zur Prüfung zugelassene Hilfsmittel:

Lernergebnisse und Inhalte

Lernergebnisse:

Nach erfolgreicher Beendigung der Veranstaltung sind die Studierenden in der Lage

- In der Industrie gängige Engines, wie z.B. Unity oder Unreal, anzuwenden
- Ein konkretes Projekt in der Gruppe zu organisieren und umzusetzen
- Ergebnisse eines Games oder Simulation in Form eines Pitches zu präsentieren

Lehrinhalte:

Projektabhängig

Literatur:

Projektabhängig



GEB2101 Game Physik

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Georg Hagel
Dozent(en):	Prof. Dr. Georg Hagel
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	Grundkenntnisse Analysis Grundkenntnisse Lineare Algebra und Analytische Geometrie Programmierung C++
Verwendbarkeit:	Studiengang Informatik - Game Engineering (B.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übungen/Praktikum in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Unterricht 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Teilnahmepflicht in der Übung, Leistungsnachweise in der Übung, schriftliche Prüfung 90 Minuten am Ende des Semesters. Leistungsnachweise sind Zulassungsvoraussetzungen.
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

Nach erfolgreicher Beendigung der Veranstaltung sind die Studierenden in der Lage,□

- zu erklären, weshalb man als Game Engineer eine eigene Physics Engine erstellen können muss.
- die physikalischen Grundlagen für eine Game Physics Engine zu beschreiben.
- die Algorithmen für die Umsetzung der physikalischen Grundlagen zu beschreiben.
- die Algorithmen für die Umsetzung der physikalischen Grundlagen in C++ mit der Entwicklungsumgebung Visual Studio zu programmieren.
- den Aufbau einer Game Physik Engine anzugeben und eine Game Physik Engine zu erstellen.
- die Probleme der Schnittstellenanpassung bei der Anbindung einer Game Physik Engine an eine Game Engine beschreiben zu können.
- eine Game Physik Engine mit einer Game Engine zu verbinden.
- ein Programm unter Verwendung einer Game Physik Engine und einer Game Engine zu schreiben.

Lehrinhalte:

- Aufbau einer Game Physics Engine
- Anbindung einer Game Physik Engine an eine Game Engine

- Teilchenphysik
- Physik zusammenhängender Massen
- Festkörperphysik
- Kollisionserkennung

Literatur:

- Ian Millington: Game Physics Engine Development, Morgan Kaufmann; 2. Auflage, 2010
- Christer Ericson: Real-Time Collision Detection, Morgan Kaufmann, 2004
- Steve Rabin (Ed.): Introduction to Game Development, Cengage Learning, 2. Auflage, 2010
- John P. Flynt; Danny Kodicek: Mathematics and Physics for Programmers, Cengage Learning, 2. Auflage, 2012
- David M. Bourg; Bryan Bywalec: Physics for Game Developers, O'Reilly, 2. Auflage, 2013
- Mike McShaffry; David "rez" Graham: Game Coding Complete; Cengage Learning, 4. Auflage, 2013
- David M. Bourg; Bryan Bywalec: Physics for Game Developers; O'Reilly, 2. Auflage, 2012



GEB2104 Computergestützte Bildbearbeitung mit Photoshop

Allgemeines

Modulverantwortliche(r):	Prof. M. A. Sabine Buchwieser
Dozent(en):	Prof. M. A. Sabine Buchwieser
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	Keine
Verwendbarkeit:	Studiengang Informatik - Game Engineering (B.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS praktische Übungen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Unterricht 30 Stunden Präsenzzeit Übungen 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Schriftliche Prüfung 90 Minuten am Ende des Semesters. Als Vorleistung 10 Übungsnachweise, die während des Semesters erstellt werden.
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

Nach erfolgreicher Beendigung der Veranstaltung sind die Studierenden in der Lage, verschiedenste Visualisierungsformen von digitalen Medien selbst zu erstellen.

Lehrinhalte:

Photoshop Grundlagen:

- Zeichenwerkzeuge
- Ebenentechnik
- Freistellungswerkzeuge
- Bildbearbeitung
- Composing
- Textur/ Muster
- Verfremdung
- Hintergründe
- Textbearbeitung

Literatur:

Keine



GEB2106 Design interaktiver Anwendungen

Allgemeines

Modulverantwortliche(r):	Prof. M. A. Sabine Buchwieser
Dozent(en):	Prof. M. A. Sabine Buchwieser
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	keine
Verwendbarkeit:	Studiengang Informatik - Game Engineering (B.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS praktische Übungen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Unterricht 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Schriftliche Prüfung 90 Minuten am Ende des Semesters. Als Vorleistung 10 Übungsnachweise, die während des Semesters erstellt werden.
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

Nach erfolgreicher Beendigung der Veranstaltung sind die Studierenden in der Lage, interaktive Anwendungsbereiche als Layout in Illustrator zu simulieren. Sie besitzen Grundkenntnisse im Programm Illustrator.

Lehrinhalte:

Design (6 Vorlesungen), Adobe Illustrator (ca. 7 Vorlesungen)

- Farbe
- Form
- Typografie
- Icons und Logo für Games
- Grundlagen und Übungen mit dem Vektoren-Zeichenprogramm Illustrator

Literatur:

- MediaFarbe - analog und digital: Farbe In Der Medienproduktion, ISBN: 978-3642622618
- Screen Design und visuelle Kommunikation: Gestaltung interaktiver Oberflächen, ISBN: 978-3778527375
- Wahrnehmungspsychologie: Der Grundkurs, ISBN: 978-3642550737
- Wie Farben wirken: Farbpsychologie - Farbsymbolik - Kreative Farbgestaltung, ISBN-13: 978-3499609237
- Die Macht der Farben: Bedeutung & Symbolik, ISBN-13: 9783854364337

-
- Das Buch für Ideensucher: Tipps und Denkanstöße von einem Insider der Kreativbranche - für jeden, der auf gute Ideen kommen muss, ISBN-13: 978-3836242820

GEB2107 Künstliche Intelligenz für Games

Allgemeines

Die KI ist neben der Computergraphik und der Physik der dritte wichtigste technische Baustein eines erfolgreichen Computerspiels. Im Vergleich ist KI für Computerspiele noch eine relativ junge Disziplin, die sich aus Erkenntnissen in anderen Anwendungsfeldern der KI bedient jedoch auch eigene für Computerspiele bestimmte Algorithmen entwickelt.

Nach erfolgreicher Teilnahme an der Lehrveranstaltung können die Studierenden Spiele so gestalten, dass virtuelle Charaktere und Gruppen von Charakteren, die nicht von Spielern kontrolliert werden, für den Spieler intelligent wirken und angemessen auf die Spielzüge reagieren. Die Teilnehmer beherrschen dann die dazu wichtigsten Verfahren der Bewegung und Entscheidungsfindung und können diese in der Programmiersprache C++ praktisch umsetzen.

Modulverantwortliche(r):	Prof. Dr. Christoph Bichlmeier
Dozent(en):	Prof. Dr. Christoph Bichlmeier
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	Kenntnisse in C++ und OOP
Verwendbarkeit:	Studiengang Informatik - Game Engineering (B.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übungen (verpflichtend)
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Unterricht 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	schriftl. Prüfung 90 Minuten am Ende des Semesters Teilnahmepflicht im Praktikum, Leistungsnachweise im Praktikum Leistungsnachweise sind Zulassungsvoraussetzungen
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

Nach erfolgreicher Teilnahme an der Lehrveranstaltung können die Studierenden

- grundlegende Algorithmen der Bewegung und der Entscheidungsfindung benennen und implementieren
- diese Algorithmen zu einem komplexeren Verhalten kombinieren
- weitere Algorithmen für speziellere Anwendungsfelder benennen
- die technische Einbettung der KI Algorithmen in ein Computerspiel umsetzen
- Basisstrukturen und Architektur einer Game KI benennen und implementieren

Lehrinhalte:

Die Themen der Lehrveranstaltung sind Algorithmen

- der Bewegung für einzelne NPCs und Gruppen von NPCs
- der Entscheidungsfindung für einzelne NPCs und Gruppen von NPCs (Strategie)
- der Kombination der Algorithmen aus den vorher genannten Teilbereichen
- der Datenverwaltung von Wissen und Zuständen
- Basisstrukturen und Architektur einer Game KI

Literatur:

- Künstliche Intelligenz (Pearson Studium - IT) von Stuart Russell, Peter Norvig
- Artificial Intelligence for Games von Ian Millington , John Funge
- Programming Game AI by Example von Mat Buckland
- Game AI Pro 1-3: Collected Wisdom of Game AI Professionals



GEB2109 Digitale Medien

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Klaus Ulhaas
Dozent(en):	Prof. Dr. Klaus Ulhaas
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	Programmierkenntnisse, mathematische Grundlagen
Verwendbarkeit:	Studiengang Informatik (B.Sc.), Studiengang Informatik - Game Engineering (B.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übung in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Vorlesung 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Anwesenheitspflicht in den Übungen, Leistungsnachweise in den Übungen, Schriftliche Prüfung 90 Minuten am Ende des Semesters. Leistungsnachweise sind Zulassungsvoraussetzung.
Zur Prüfung zugelassene Hilfsmittel:	nicht programmierbarer Taschenrechner

Lernergebnisse und Inhalte

Lernergebnisse:

Die Studierenden sind in der Lage

- die Bedeutung und Relevanz Digitaler Medien und deren Kompressionsverfahren für Computer Games und Browser-Anwendungen einzustufen.
- Abtastungsprobleme bei Digitalen Medien zu erkennen und die Ursache hierfür zu identifizieren.
- den Informationsgehalt einer Nachricht zu bestimmen.
- verlustfreie Kompressionsverfahren zu benennen und die Verfahren auf unterschiedliche Nachrichten wie z.B. Bild, Ton oder Textinformationen anzuwenden.
- zu verschiedenen Kodierungen anzugeben, wie gut die jeweilige erreichte Kompression ist.
- eine Lauflängenkodierung, Huffman-Kodierung, arithmetische Kodierung, LZW-Kodierung anzuwenden und günstige und ungünstige Anwendungsgebiete zu benennen.
- Verlustbehaftete Kompressionsverfahren zu benennen.
- die Verfahren JPEG und MPEG-Kodierung anzuwenden und teilweise zu implementieren.
- Verfahren zum Schutz digitaler Medien zu benennen und einfache Verfahren programmiertechnisch umzusetzen.

Lehrinhalte:

- Begriffsdefinitionen und wirtschaftliche Sicht zu Digitalen Medien und Digitalen Gütern
- Motivation für den Einsatz Digitaler Medien in Games und Browseranwendungen
- Abtastung und Digitalisierung
- Informationstheoretische Grundlagen zum Verständnis der Kompressionstechnik
- Verlustfreie Kompressionsverfahren
- Verlustbehaftete Kompressionsverfahren
- Urheberrecht und Mediensicherheit

Literatur:

Literatur und Empfehlungen zu Einzelthemen werden fortlaufend in der Vorlesung bekanntgegeben



GEB2114 Operations Research

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Jochen Staudacher
Dozent(en):	Prof. Dr. Jochen Staudacher
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	Keine
Verwendbarkeit:	Studiengang Informatik (B.Sc.), Studiengang Informatik - Game Engineering (B.Sc.), Studiengang Wirtschaftsinformatik (B.Sc.), Studiengang Gesundheits- und Pflegeinformatik (B.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	3 SWS Seminaristischer Unterricht 1 SWS Übung in kleinen Gruppen (14tägig 90 Minuten)
Leistungspunkte:	5
Arbeitsaufwand:	45 Stunden Präsenzzeit Vorlesung 15 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Erfolgreiche Teilnahme an den Übungen, Leistungsnachweise zu Praktikumsaufgaben in den Übungen, schriftliche Prüfung 90 Minuten am Ende des Semesters. Leistungsnachweise sind Zulassungsvoraussetzung.
Zur Prüfung zugelassene Hilfsmittel:	kein Taschenrechner, ansonsten ohne Einschränkung

Lernergebnisse und Inhalte

Lernergebnisse:

Nach erfolgreicher Beendigung der Veranstaltung können die Studierenden

- die grundlegenden Konzepte der Linearen Optimierung definieren
- einfache betriebswirtschaftliche Aufgaben als mathematische Modelle formulieren
- Methoden der Linearen Optimierung auf einfache Fragestellungen aus der Informatik und den Wirtschaftswissenschaften anwenden und die Ergebnisse interpretieren
- einfache Sensitivitätsanalysen durchführen
- die Grenzen der Linearen Optimierung exemplarisch aufzeigen

Lehrinhalte:

- Mathematische Grundlagen der Linearen Optimierung
- Das Simplex-Verfahren und seine Varianten
- Dualitätstheorie
- Alternativen zum Simplex-Verfahren
- Spezialfälle der Linearen Optimierung
- Ganzzahlige lineare Optimierung

- Sensitivitätsanalysen, Parametrische Lineare Optimierung
- Einfache Zweipersonen-Nullsummenspiele

Literatur:

- A. Koop, H. Moock: Lineare Optimierung: Eine anwendungsorientierte Einführung in Operations Research, Spektrum Akademischer Verlag, 2. Auflage, 2018
- P. Stingl: Operations Research: Lineare Optimierung, Hanser Fachbuchverlag, 1. Auflage, 2002
- H.-J. Zimmermann: Operations Research: Methoden und Modelle. Für Wirtschaftsingenieure, Betriebswirte, Informatiker, Vieweg+Teubner, 2. Auflage, 2007
- J. Schwarze: Mathematik für Wirtschaftswissenschaftler. Band 3: Lineare Algebra, Lineare Optimierung und Graphentheorie, Nwb Verlag, 13. Auflage, 2011
- K. Neumann, M. Morlock: Operations Research, Hanser Fachbuchverlag, 2. Auflage, 2002
- P.R. Thie, G.E. Keough: An Introduction to Linear Programming and Game Theory, 3rd Edition, 2008.
- R.J. Vanderbei: Linear Programming, Springer, 4th Edition, 2014.



GEB2115 Modellierung und Animation 2

Allgemeines

Modulverantwortliche(r):	Hr. André Kettner
Dozent(en):	Hr. André Kettner
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	Keine
Verwendbarkeit:	Studiengang Informatik - Game Engineering (B.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS betreutes Praktikum in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Vorlesung 30 Stunden Präsenzzeit Praktikum 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Schriftliche Prüfung 90 Minuten am Ende des Semesters Teilnahmepflicht im Praktikum, Leistungsnachweise im Praktikum Leistungsnachweise sind Zulassungsvoraussetzungen
Zur Prüfung zugelassene Hilfsmittel:	handschriftliche Notizen, 1 DIN A4 Blatt, beidseitig beschrieben, keine Kopie

Lernergebnisse und Inhalte

Lernergebnisse:

- Die Studierenden sind in der Lage, mittels fortgeschrittener Modellierungsmethoden sowie prozeduraler Methoden, komplexe Geometrie zu erstellen.
- Die Studierenden sind in der Lage, mit Node-Setups Kompositionen von Materialien zu erstellen sowie bestehende Materialien je nach Renderkontext aufzubereiten.
- Die Studierenden verstehen, spezielle Anforderungen an das Szenensetup für fotorealistisches Rendern zu erfüllen und sind in der Lage bestehende Modelle, Materialien und Beleuchtungsmethoden hinsichtlich dieser zu beurteilen und zu optimieren.
- Die Studierenden sind in der Lage, fortgeschrittene Animationen zu erstellen und kennen gängige physikalisch begründete Animationsmodelle.
- Die Studierenden sind in der Lage, Szenen bezüglich ihrer Performanz zu beurteilen und eigene sowie bereits bestehende Szenen für performantes verteiltes Rendern zu optimieren.

Lehrinhalte:

- Grundlegende Prinzipien des fotorealistischen Renderns
- Überblick über globale Beleuchtungsverfahren und Image-Based Lighting
- Einführung in prozedurale Methoden
- Node-basiertes Materialsetup und Komposition komplexer Materialmodelle
- Ambient Occlusion und Occlusion Maps

- Alternative Texturierungsverfahren und Per-Face Texture Mapping
- Rendereffekte wie Depth of Field und Motion Blur und Überblick über gängige Post-Processing Filterverfahren
- Animation mittels Blendshapes
- Überblick über physikalisch basierte Animationsmodelle und Einführung in Fluidsimulationen
- Dynamische Modelle zur Animation von Textilien und Haare und Grundlagen Partikel-basierter Animationen
- Performance-Betrachtungen und Optimierungsmöglichkeiten

Literatur:

- Tomas Akenine-Möller et al.: Real-Time Rendering, CRC Press, Taylor and Francis Group, NY 2008
- Rick Parent: Computer Animation, Algorithms and Techniques, Morgan Kaufmann Publishers, San Francisco, 2002



GEB2116 Sound- und Musikproduktion für Games

Allgemeines

Modulverantwortliche(r):	Hr. Sebastian Kern
Dozent(en):	Hr. Sebastian Kern
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	Keine
Verwendbarkeit:	Studiengang Informatik - Game Engineering (B.Sc.)
Angebot und Dauer:	Wintersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS betreutes Praktikum in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Vorlesung 30 Stunden Präsenzzeit Praktikum 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Teilnahmepflicht im Praktikum, der endnotenbildende Leistungsnachweis besteht aus einer benoteten Dokumentation, benoteten Übungen und einem benoteten Abschlussprojekt.
Zur Prüfung zugelassene Hilfsmittel:	

Lernergebnisse und Inhalte

Lernergebnisse:

Die Studierenden sind in der Lage mit Hilfe einer DAW (DigitalAudioWorkstation) Soundeffekte und Geräusche selbst zu erzeugen oder aufzunehmen, zu bearbeiten und diese mittels Wwise ins Game zu implementieren. Sie besitzen Grundkenntnisse in Musiktheorie und Harmonielehre und in speziellen Kompositions- und Recording-Techniken für Games sowie in die Programme LogicX (oder ähnliche) und Wwise.

Lehrinhalte:

- Überblick über die Geschichte von Sound und Musik in Games
- Sounddesign im Game
- Implementierung von Sound im Game
- Grundbegriffe der Harmonielehre
- Kompositionstechniken für Games
- Grundlagen der Musikproduktion für Games
- Umgang mit den Programmen LogicX und Wwise

Literatur:

N.N.



GEB2117 Farb- und Formpsychologie

Allgemeines

Modulverantwortliche(r):	Prof. Dr. Tobias Breiner
Dozent(en):	Prof. Dr. Tobias Breiner
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	keine
Verwendbarkeit:	Studiengang Informatik - Game Engineering (B.Sc.), Studiengang Wirtschaftsinformatik (B.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	4 SWS Seminaristischer Unterricht
Leistungspunkte:	5
Arbeitsaufwand:	60 Stunden Präsenzzeit Unterricht 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	schriftl. Prüfung 90 Minuten am Ende des Semesters
Zur Prüfung zugelassene Hilfsmittel:	keine Hilfsmittel

Lernergebnisse und Inhalte

Lernergebnisse:

- Die Studierenden wissen um die Fallstricke der Wahrnehmung.
- Die Studierenden kennen gestalterische Farb- und Formprinzipien und können damit umgehen.
- Die Studierenden können ein Computerspiel optimal mit geeigneten Farb- und Formelementen designen.
- Die Studierenden können Daten effektiv, expressiv und angemessen in geeignete Visualisierungen umsetzen.

Lehrinhalte:

Optik für Computergraphiker

- Grundlegende Lichtbegriffe (Welle-Teilchen-Dualismus, Wellenlänge, Frequenz, Lichtgeschwindigkeit, Periode, Phase, Polarisierung, Kohärenz, Compton-Effekt)
- Umrechnungsformel Wellenlänge - Frequenz
- Licht als Teil des elektromagnetischen Spektrums
- physikalische Grundlagen der Lichtausbreitung (Brechungsindex, Refraktion und Reflexion, Blaustreuung)
- Lichtphänomene
- Chromatische Aberration und Farbstereoskopie

Aufbau der Sehpipeline

- Physiologische und wahrnehmungspsychologische Grundlagen des Sehens
- Aufbau des Visuellen Systems
- Sehpipeline (Aufbau Auge, Mentale Bildverarbeitung, Parietaler vs. Temporaler Strom etc.)

- Optische Täuschungen (insbesondere in Hinblick auf Modellierungsschwierigkeiten und Feindsichtungen bei First-Person-Shootern)
- Camouflage, Figur-Grund-Trennung)
- Ambiguous Images, Ambigramme, Links-rechts-Ambivalenz

Farbtheorien

- Farbeffekte (Bezold-Brücke-Effekt, Braun-Effekt etc.)
- Grassmannsche Gesetze
- Stereosehen und Formwahrnehmung
- photosensitive Epilepsie durch Farbabfolgen

Farbsysteme

- Historische Farbmodelle (Farbmodelle v. Aristoteles, Kircher, Goethe, Runge, ...)
- Künstlerfarbmodelle (HSV, HLS,...)
- Technische Farbmodelle (RGB, CMY, CMYK, YUV, ...)
- CIE-Farbmodelle (CIE-XYZ, CIE-LUV/LAB)
- Normfarbsysteme (HKS, RAL, ...)
- psychologisches Farbsystem

Wirkung von Farbe und Form auf das Verhalten

- Auswirkung von Farben im Sport
- Auswirkung von Farben im Gaming
- Wechselseitige Interdependenzen zwischen Farbe und Sprache

Farbassoziationen

- empirisch gewonnene Assoziationen zu Einzelfarben und Farbkombinationen
- kulturelle Besonderheiten bei Farbassoziationen
- konnotative Überschreibungen von Farbe

Einsatz von Farbe

- Farbskalen und ihre Anwendung in der Visualisierung bzw. Games
- Farbkodierung nach Bauer, Colin Ware, Schuhmann und Müller u.a.
- Wahrnehmungsmäßige Gleichabständigkeit
- Auswahl von Farbskalen und Farbkombinationen für Visualisierungen und Games
- Game-Spezifika für Farbenblinde
- Einsatz von Farbe in der Kunst

Anatomie und Physiognomie für Computerspiel-Charaktere

- Gesichtserkennung
- Männliche vs. weibliche Gesichtsmerkmale
- Physiognomie von Helden
- Sympathische vs. Antipathische Gesichter
- Thatcher-Effekt
- Kindchenschema
- Klassische Körperproportionen
- Körperproportionen von Helden

- Körperproportionen von Fabelwesen
- Muskelaufbau von Menschen

Einsatz von Farbe und Form in der Werbung

- Die Phasen der Werbung
- Erzeugung von Werbekonnotationen durch Farbe und Form
- Negation und Provokation in der Werbung
- Auswirkung von sexuellen Symbolen in der Werbung
- subliminale Werbeimpulse

Wahrnehmungsstörungen durch Formen und Bewegungen

- Inattentional Blindness
- Wunschsehen
- Motion Blindness

Formassoziationen

- Assoziationen zu bestimmten Symbolen
- Formkombinationen
- Goldener Schnitt
- Goldener Winkel

Gestaltpsychologie

Visualisierung (optional)

- Visualisierungspipeline
- Ausgewählte Verfahren der wissenschaftlich-technischen Visualisierung und der Informationsvisualisierung (z.B. Isoflächen)
- Virtual Environments
- Grundlagen der Datenvisualisierung (wichtige Chart- und Diagrammart, Ziele und Probleme der Datenvisualisierung)
- Effektivität, Expressivität und Angemessenheit von Visualisierungen
- Monoskopie vs. Stereoskopie

Arbeiten mit Bildverarbeitungsprogrammen (optional, nur Theorie)

- Pixelgrafikbearbeitungsprogramme (am Beispiel Adobe Photoshop)
- Vektorgrafikbearbeitungsprogramme (am Beispiel Adobe Illustrator) , nur Prinzip
- Morphing-Programme (am Beispiel WinMorph)

Literatur:

- Breiner: Breiner, Tobias (2018): Farb- und Formpsychologie, Springer-Verlag, Berlin (die diesbez. klausurrelevanten Informationen finden sich auch in den Vorlesungsbegleitenden Folien, so dass das Buch nicht gekauft werden muss)



GEB2118 Logik

Allgemeines

Die Logik ist ein wesentliches Werkzeug menschlichen Denkens. Gerade in der Informatik ist es wichtig, die wesentlichen Aspekte dieses Werkzeugs zu benennen und zu sehen wie man sie formalisieren und automatisieren kann. Die Veranstaltung geht auf diese Aspekte ein und zeigt ihre Grenzen auf, welche von ganz grundsätzlicher Natur sind und Rückschlüsse auch auf das prinzipielle Denkvermögen von Menschen erlauben.

Modulverantwortliche(r):	Prof. Dr. Elmar Böhler
Dozent(en):	Prof. Dr. Elmar Böhler
Modultyp:	Wahlpflichtmodul
Voraussetzungen:	Programmieren, Theoretische Informatik, Algorithmen und Datenstrukturen
Verwendbarkeit:	Studiengang Informatik (B.Sc.), Studiengang Informatik - Game Engineering (B.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übung in kleinen Gruppen
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit Vorlesung 30 Stunden Präsenzzeit Übung 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Teilnahmepflicht an den Übungen; schriftliche Prüfung (90 Minuten) am Ende des Semesters; erfolgreiche Teilnahme an den Übungen ist Zulassungsvoraussetzung
Zur Prüfung zugelassene Hilfsmittel:	handschriftliche Notizen, 1 DIN A4 Blatt, einseitig beschrieben, keine Kopie

Lernergebnisse und Inhalte

Lernergebnisse:

Die Studierenden können logische Aussagen erkennen und formalisieren. Sie erkennen inwiefern Formeln und Schaltkreise Codierungen von Wahrheitswertetabellen sind und werden sich der mitunter einhergehenden kombinatorischen Explosion bewusst. Sie verstehen den Unterschied und Zusammenhang zwischen Formeln und Booleschen Funktionen. Sie lernen wesentliche Familien von BF zu erkennen und vollständige Mengen von BF zu identifizieren. Die Begriffe des Folgerns und Schließens werden konkretisiert, wobei klar wird warum jede Aussage, auf die geschlossen werden kann, auch nach mechanischen Prinzipien automatisch abgeleitet werden kann (Vollständigkeitssatz). Die Studierenden lernen die Grenzen des logischen Folgerns kennen (Unvollständigkeitssatz). Sie werden in Mechanismen des automatischen Folgerns eingeführt, mit entsprechenden Werkzeugen bekannt gemacht und lernen ihre Grenzen kennen.

Lehrinhalte:

- Aussagenlogik
- Formulieren logischer Aussagen



- Formeln und Schaltkreise
- Wahrheitstabelle und Boolesche Funktionen
- Abgeschlossene Mengen von BF und Postsches Vollständigkeitskriterium
- Äquivalenz und Normalformen von Formeln
- Folgern und Schließen
- Vollständigkeitssatz/Unvollständigkeitssatz
- Hornformeln
- Resolution
- Prädikatenlogik
- Normalformen
- Unifikation
- Termersetzungssysteme/Automatische Beweissysteme
- Logikprogrammierung, Prolog
- Zusammenhang von Größe der WW-Tabelle und der Schaltkreisgröße
- Komplexitätsbetrachtungen z.B. der Probleme der Schaltkreisauswertung und der Schaltkreiserfüllbarkeit

Literatur:

- U. Schöning: Logik für Informatiker
- K. Wagner: Theoretische Informatik
- S. Russel, P. Norvig: Artificial Intelligence
- H. Vollmer: Introduction to Circuit Complexity

GEB3100.1 Praktisches Studiensemester

Allgemeines

Modulverantwortliche(r):	Praxisbeauftragter Prof. Dr. Bernd Dreier
Dozent(en):	Praxisbeauftragter Prof. Dr. Bernd Dreier
Modultyp:	Pflichtmodul
Voraussetzungen:	Zulassung zum Vertiefungsstudium und in den Fächern des Vertiefungsstudiums mind. 20 ECTS erreicht
Verwendbarkeit:	Studiengang Informatik - Game Engineering (B.Sc.)
Angebot und Dauer:	Wintersemester/Sommersemester
Lehrformen:	Praktische Tätigkeit
Leistungspunkte:	25
Arbeitsaufwand:	Zusammenhängender Zeitraum von mind. 20 Wochen (einschließlich der praxisbegleitenden Lehrveranstaltungen), wobei die tägliche Arbeitszeit der üblichen Arbeitszeit der Ausbildungsstelle entspricht.
Leistungsnachweis und Prüfung:	Praxisbericht
Zur Prüfung zugelassene Hilfsmittel:	

Lernergebnisse und Inhalte

Lernergebnisse:

Im praktischen Studiensemester widmen sich Studierende deutlich berufsbezogenen Tätigkeiten. Die praktische Ausbildung stellt die Verbindung zwischen Studium und Berufspraxis her und ist auf die Erfordernisse in Industrie, Wirtschaft, Verwaltung etc. ausgerichtet. Die praktische Ausbildung wird durch praxisbegleitende Lehrveranstaltungen der Fakultät ergänzt. Sie dienen der Integration von Praxis und Theorie sowie der Auswertung und Vertiefung der praktischen Tätigkeiten innerhalb des praktischen Studiensemesters.

Lehrinhalte:

Literatur:

GEB3100.2 Praxisbegleitende Lehrveranstaltung

Allgemeines

Modulverantwortliche(r):	Fr. Bodmer
Dozent(en):	Fr. Bodmer / Fr. Breitfeld
Modultyp:	Pflichtmodul
Voraussetzungen:	Keine
Verwendbarkeit:	Studiengang Informatik - Game Engineering (B.Sc.)
Angebot und Dauer:	Wintersemester, Sommersemester, Blocklehrveranstaltung
Lehrformen:	2 SWS Seminaristischer Unterricht 2 SWS Übungen in kleinen Gruppen (Workshop) Blocklehrveranstaltung
Leistungspunkte:	5
Arbeitsaufwand:	60 Stunden Präsenzzeit 90 Stunden Selbststudium
Leistungsnachweis und Prüfung:	Präsentation
Zur Prüfung zugelassene Hilfsmittel:	

Lernergebnisse und Inhalte

Lernergebnisse:

Nach erfolgreicher Teilnahme können die Studierenden

- die soziale Kompetenz als Handlungsprämisse erkennen
- ihre soziale und fachliche Kompetenz verknüpfen
- Sachverhalte sach- und situationsangemessen darstellen,
- verschiedene Präsentationstechniken zielgruppenorientiert anwenden,
- Argumentationsmuster und Argumentationsstrategien analysieren und beurteilen,
- sprachliche, sprecherische und rhetorische Gestaltungsmittel von Botschaften gezielt einsetzen.

Lehrinhalte:

- Kommunikationsmuster und Gesprächsführung
- Umgang mit Konflikten
- Grundlagen der Teamarbeit
- Persönlichkeit als Ressource/Persönlichkeitsmodelle
- Motivation als Erfolgsfaktor
- Grundlagen der Präsentation
- Selbst- und Fremdwahrnehmung
- Praktische Übungen

Literatur:

GEB4100 Seminar

Allgemeines

Modulverantwortliche(r):	Professoren und Lehrbeauftragte
Dozent(en):	Professoren und Lehrbeauftragte
Modultyp:	Pflichtmodul
Voraussetzungen:	Keine
Verwendbarkeit:	Studiengang Informatik - Game Engineering (B.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	Seminar
Leistungspunkte:	5
Arbeitsaufwand:	15 Stunden Präsenzzeit Vorträge und anschließender Diskussion 135 Stunden selbständiges Arbeiten, Vorbereitung der Präsentation, Ausarbeitung der Studienarbeit
Leistungsnachweis und Prüfung:	Studienarbeit und / oder Kolloquium
Zur Prüfung zugelassene Hilfsmittel:	

Lernergebnisse und Inhalte

Lernergebnisse:

Nach dem erfolgreichen Besuch der Lehrveranstaltung sind die Studierenden in der Lage, ...

- sich in ein vorgegebenes Thema einzuarbeiten, entsprechende Literatur zu recherchieren und aufzuarbeiten
- einen Vortrag zu einem vorgegebenen Thema zu erarbeiten, zu präsentieren und bei der nachfolgenden Diskussion zu vertreten
- eine schriftliche Ausarbeitung zu einem vorgegebenen Thema zu erstellen

Lehrinhalte:

Themenabhängig

Literatur:

Themenabhängig

GEB5100 Projektarbeit

Allgemeines

Modulverantwortliche(r):	Professoren der Fakultät
Dozent(en):	Professoren der Fakultät
Modultyp:	Pflichtmodul
Voraussetzungen:	Keine
Verwendbarkeit:	Studiengang Informatik - Game Engineering (B.Sc.)
Angebot und Dauer:	Sommersemester, ein Semester
Lehrformen:	Projekt
Leistungspunkte:	15
Arbeitsaufwand:	15 Stunden Präsenzzeit Unterricht 435 Stunden selbständiges Arbeiten
Leistungsnachweis und Prüfung:	Projektbericht Präsentation
Zur Prüfung zugelassene Hilfsmittel:	

Lernergebnisse und Inhalte

Lernergebnisse:

Nach erfolgreicher Beendigung der Veranstaltung sind die Studierenden in der Lage, ...

- ein Projekt zu planen, durchzuführen und sorgfältig zu dokumentieren
- ihr erworbenes Wissen in Standard-Anwendungsszenarien einzusetzen und erfolgreich anzuwenden
- die Entwicklungsergebnisse in Form von Postersessions und Kurzpräsentationen vorzustellen

Lehrinhalte:

Projektabhängig

Literatur:

Projektabhängig



GEB6100.1/6100.2 Bachelorarbeit / -seminar

Allgemeines

Modulverantwortliche(r):	Betreuender Professor
Dozent(en):	Betreuender Professor
Modultyp:	Pflichtmodul
Voraussetzungen:	Es gelten die in der Studien- und Prüfungsordnung genannten Voraussetzungen.
Verwendbarkeit:	Studiengang Informatik - Game Engineering (B.Sc.)
Angebot und Dauer:	Es gelten insbesondere die Regelungen der Allgemeinen Prüfungsordnung der Hochschule Kempten.
Lehrformen:	-
Leistungspunkte:	15
Arbeitsaufwand:	
Leistungsnachweis und Prüfung:	Abschlussarbeit und Vortrag über das Thema der Abschlussarbeit im Rahmen eines Seminars. Der Vortrag findet entweder im Unternehmen oder im Rahmen eines Seminars an der Hochschule statt.
Zur Prüfung zugelassene Hilfsmittel:	

Lernergebnisse und Inhalte

Lernergebnisse:

Mit der Bachelorarbeit soll der Studierende beweisen, dass er in der Lage ist, eine Problemstellung - praktischer oder theoretischer Natur - innerhalb eines begrenzten und definierten Zeitraums nach wissenschaftlichen Methoden zu bearbeiten. Die Abschlussarbeit darf mit Zustimmung der Prüfungskommission in einer Einrichtung außerhalb der Hochschule ausgeführt werden.

Lehrinhalte:

Literatur:

In Absprache mit dem betreuenden Professor