

**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**  
**DERS BİLGİ FORMU**

Dersin Ayrıntıları				
Dersin Kodu	Sınıfı			Yarıyılı
INF101	1			Güz
Dersin Adı	T	U	L	AKTS
Bilgisayar Bilimi ve Programlamaya Giriş	2	0	2	6
Dersin Dili	Almanca			
Dersin Düzeyi	Lisans	X	Yüksek Lisans	Doktora
Bölümü/Programı	Bilgisayar Mühendisliği			
Eğitim Türü	Yüzyüze ders anlatımı, kişisel çalışma.			
Dersin Türü	Zorunlu	X	Seçmeli	
Dersin Amacı	Bu dersi başarıyla tamamladıktan sonra, öğrenciler bilgisayar biliminin temel kavramlarını ve yöntemlerini tanımlayabilirler. Programlama bilgisine ve temel veri yapıları hakkında bilgiye sahip olurlar. Sorunları algoritmik olarak programlara dönüştürebilir ve C ve C ++ programlama dillerini kullanabilirler.			
Dersin İçeriği	<b>Bilgisayar Bilimine Giriş</b> - bilgisayarlarda veri gösterimi - kodlama teorisi  <b>Programlamaya Giriş</b> - algoritma, program kavramları - veri tipleri, değişkenler, operatörler - mantıksal ifadeler, akış kontrolü, döngüler - fonksiyonlar, geçerlilik alanları - işaretçiler (pointer) - numaralandırmalar, yapılar, alanlar - Arduino ile mikroişlemci programlama (ilgilenen öğrenciler için isteğe bağlı)  Öğrenciler, önceden belirlenmiş, ilgili programlama ödevlerini bağımsız olarak çözerek, programlayarak ve teslim ederek bu konuları ele alırlar.			
Ön Koşulları	Yok			
Dersin Koordinatörü	Dr.-Ing. Volkan Gezer			
Dersi Verenler	Dr.-Ing. Volkan Gezer			
Dersin Yardımcıları	-			
Dersin Staj Durumu	Yok			
Ders Kaynakları				
Ders Notu	- Hartmut Ernst, Jochen Schmidt, Gerd Beneken. Grundkurs Informatik. Springer Viewek, 2016			
Diğer Kaynaklar	- Helmut Erlenkötter. C: Programmieren von Anfang an. Rowohlt Taschenbuch Verlag, 1999.			

**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**  
**DERS BİLGİ FORMU**

**Materyal Paylaşımı**

Dokümanlar	-
Ödevler	-
Sınavlar	-

**Dersin Yapısı**

Matematik ve Temel Bilimler	20	%
Mühendislik Bilimleri	20	%
Mühendislik Tasarımı		%
Sosyal Bilimler		%
Eğitim Bilimleri		%
Fen Bilimleri		%
Sağlık Bilimleri		%
Alan Bilgisi	60	%

**Değerlendirme Sistemi**

	Sayısı	Katkı Oranı (%)
Ara Sınav	1	40
Kısa Sınav		
Ödev		
Devam		
Uygulama		
Proje		
Yarıyıl Sonu Sınavı	1	60
<b>Toplam</b>		<b>100</b>

**AKTS İş Yüğü Dağılımı Tablosu**

	Sayısı	Süresi	Toplam İş Yüğü (Saat)
Ders Süresi	14	2	28
Sınıf Dışı Ç. Süresi	1	60	60
Ödevler			
Sunum/Seminer Hazırlama			
Ara Sınavlar	1	3	3
Uygulama			
Laboratuvar	15	4	60
Proje			
Yarıyıl Sonu Sınavı	1	10	10
<b>Toplam İş Yüğü</b>			<b>161</b>

**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**  
**DERS BİLGİ FORMU**

AKTS Kredisi (Toplam İş Yüğü / 28)		6					
<b>Dersin Öğrenim Çıktıları</b>							
1	Bilgisayarlarda farklı veri türlerinin nasıl görüntülediğini bilmek.						
2	Bilgisayarlarda sayı aritmetiği bilgisine sahip olmak.						
3	Hataya dayanıklı, sıkıştırma ve şifreleme kodlama yöntemleri bilgisine sahip olmak.						
4	Algoritma geliştirebilme ve C programlama dilinde program yazabilme.						
<b>Ders Konuları</b>							
1	Bilgisayar bilimine giriş, tarihçe, bilgisayarlarda veri saklama						
2	Sayı sistemleri ve ikili aritmetik						
3	C programlama (temel terimler: algoritma, akış şeması)						
4	C programlama (veri tipleri, değişkenler)						
5	C programlama (matematiksel ve mantıksal operatörler)						
6	C programlama (if ifadeleri, akış kontrolü)						
7	C programlama (goto döngü yapısı)						
8	C programlama (döngüler)						
9	Ara Sınav						
10	Kodlama ve şifreleme						
11	C programlama (diziler ve yapılar)						
12	C programlama (değişkenlerin fonksiyonları ve kapsamı)						
13	C programlama (özyinelemeli fonksiyonlar)						
14	C programlama (fonksiyonlar, değere göre çağrı, referansla çağrı)						
15	C programlama (pointer)						
<b>Dersin Program Çıktılarına Katkısı (1-5)</b>							
	<b>P1</b>	<b>P2</b>	<b>P3</b>	<b>P4</b>	<b>P5</b>	<b>P6</b>	<b>P7</b>
1	5	5	5			3	1
2	5	5	5			3	1
3	5	5	5			3	1
4	5	5	5			3	1
<b>Katkı Oranı: 1: Çok Düşük 2: Düşük 3: Orta 4: Yüksek 5: Çok Yüksek</b>							
<a href="http://bm.tau.edu.tr/program-yeterlikleri">http://bm.tau.edu.tr/program-yeterlikleri</a>							
<b>Hazırlayan:</b>	Ayşe Betül Yüce						
<b>Güncelleme Tarihi:</b>	24.06.2022						

DEPARTMENT OF COMPUTER SCIENCE  
COURSE SYLLABUS

Course Details				
<b>Code</b>	<b>Academic Year</b>			<b>Semester</b>
INF101	1			Fall
<b>Title</b>	<b>T</b>	<b>A</b>	<b>L</b>	<b>ECTS</b>
Introduction to Computer Science and Programming	2	0	2	6
<b>Language</b>	German			
<b>Level</b>	<b>Undergraduate</b>	X	<b>Graduate</b>	<b>Postgraduate</b>
<b>Department / Program</b>	Computer Science			
<b>Forms of Teaching and Learning</b>	Lecture, Individual Study.			
<b>Course Type</b>	<b>Compulsory</b>	X	<b>Elective</b>	
<b>Objectives</b>	After successfully completing this module, students are able to describe elementary concepts and methods of computer science. You have knowledge of imperative programming and basic knowledge of basic data structures. They are able to algorithmically convert problems into programs and use the programming languages C and C ++.			
<b>Content</b>	<p><b>Introduction to Computer Science</b></p> <ul style="list-style-type: none"> <li>- data representation in computers</li> <li>- coding theory</li> </ul> <p><b>Introduction to Programming</b></p> <ul style="list-style-type: none"> <li>- algorithm, specification, program</li> <li>- data types, variables, operators</li> <li>- logical expressions, flow control, loops</li> <li>- functions, areas of validity</li> <li>- pointers</li> <li>- enumerations, structures, fields</li> <li>- microprocessor programming with Arduino (optional for interested students)</li> </ul> <p>Students deal with these concepts by independently solving, programming and handing in predetermined, relevant programming tasks.</p>			
<b>Prerequisites</b>	None			
<b>Coordinator</b>	Dr.-Ing. Volkan Gezer			
<b>Lecturer(s)</b>	Dr.-Ing. Volkan Gezer			
<b>Assistant(s)</b>	-			
<b>Work Placement</b>	None			
Recommended or Required Reading				
<b>Books / Lecture Notes</b>	- Hartmut Ernst, Jochen Schmidt, Gerd Beneken. Grundkurs Informatik. Springer Viewek, 2016			
<b>Other Sources</b>	- Helmut Erlenkötter. C: Programmieren von Anfang an. Rowohlt Taschenbuch Verlag, 1999.			

DEPARTMENT OF COMPUTER SCIENCE  
COURSE SYLLABUS

Additional Course Material			
Documents	-		
Assignments	-		
Exams	-		
Course Composition			
Mathematics und Basic Sciences	20		%
Engineering	20		%
Engineering Design			%
Social Sciences			%
Educational Sciences			%
Natural Sciences			%
Health Sciences			%
Expert Knowledge	60		%
Assessment			
Activity	Count		Percentage (%)
Midterm Exam	1		40
Quiz			
Assignments			
Attendance			
Recitations			
Projects			
Final Exam	1		60
		<b>Total</b>	<b>100</b>
ECTS Points and Work Load			
Activity	Count	Duration	Work Load (Hours)
Lectures	14	3	28
Self-Study	1	60	60
Assignments			
Presentation / Seminar Preparation			
Midterm Exam	1	3	3
Recitations			
Laboratory	15	4	60
Projects			
Final Exam	1	10	10
		<b>Total Work Load</b>	<b>161</b>

DEPARTMENT OF COMPUTER SCIENCE  
COURSE SYLLABUS

ECTS Points (Total Work Load / 28)		6					
<b>Learning Outcomes</b>							
1	Know how different types of data are displayed in computers.						
2	Knowledge of number arithmetic in computers.						
3	Knowledge of fault-tolerant, compressing and encrypting coding methods						
4	Independent development of algorithms in pseudo code and implementation in the programming language C.						
<b>Weekly Content</b>							
1	Introduction to computer science, history, data display in computers						
2	Number systems and binary arithmetic						
3	Programming in C (basic terms: algorithm, flow chart)						
4	Programming in C (data types, variables)						
5	Programming in C (mathematical and logical operators)						
6	Programming in C (if statements, flow control)						
7	Programming in C (goto loop construction)						
8	Programming in C (loops)						
9	Midterm exams						
10	Coding and encryption						
11	Programming in C (arrays and structures)						
12	Programming in C (functions and scope of variables)						
13	Programming in C (recursive functions)						
14	Programming in C (functions, call-by-value, call-by-reference)						
15	Programming in C (pointer)						
<b>Contribution of Learning Outcomes to Program Objectives (1-5)</b>							
	<b>P1</b>	<b>P2</b>	<b>P3</b>	<b>P4</b>	<b>P5</b>	<b>P6</b>	<b>P7</b>
1	5	5	4			3	1
2	5	5	4			3	1
3	5	5	4			3	1
4	5	5	4			3	1
<b>Contribution Level</b>		1: Low 2: Low-intermediate 3: Intermediate 4: High 5: Very High					
<a href="http://bm.tau.edu.tr/learning-objectives-of-the-program">http://bm.tau.edu.tr/learning-objectives-of-the-program</a>							
<b>Compiled by:</b>		Ayşe Betül Yüce					
<b>Date of Compilation:</b>		24.06.2022					

## STUDIENGANG INFORMATIK MODULBESCHREIBUNG

Details zum Modul										
<b>Code</b>	INF101		<b>Studienjahr</b>	1	<b>Studiensemester</b>	WiSe				
<b>Bezeichnung</b>	Einführung in die Informatik und Programmierung		<b>VL</b>	2	<b>UE</b>	0	<b>LU</b>	2	<b>ECTS</b>	6
<b>Sprache</b>	Deutsch									
<b>Studium</b>	<b>Bachelor</b>	X	<b>Master</b>		<b>Doktor</b>					
<b>Studiengang</b>	Informatik									
<b>Lehr- und Lernformen</b>	Vorlesung, Selbststudium									
<b>Modultyp</b>	<b>Pflichtfach</b>	X	<b>Wahlfach</b>							
<b>Lernziele</b>	<p>Nach erfolgreichem Abschluss dieses Moduls sind Studierenden in der Lage, elementare Konzepte und Methoden der Informatik zu beschreiben. Sie haben Kenntnisse der imperativen Programmierung sowie Grundkenntnisse der grundlegenden Datenstrukturen. Sie sind befähigt Problemstellungen algorithmisch in Programme zu überführen und die Programmiersprachen C und C++ anzuwenden.</p>									
<b>Lerninhalte</b>	<p><b>Einführung in die Informatik:</b></p> <ul style="list-style-type: none"> <li>- Informationsdarstellung, Darstellung von Zahlen und Zeichen im Rechner</li> <li>- Codierungstheorie</li> </ul> <p><b>Einführung in die Programmierung:</b></p> <ul style="list-style-type: none"> <li>- Algorithmus, Spezifikation, Programm</li> <li>- Datentypen, Variablen, Operatoren</li> <li>- Logische Ausdrücke, Flusskontrolle, Schleifen</li> <li>- Funktionen, Geltungsbereiche</li> <li>- Zeiger</li> <li>- Aufzählungen, Strukturen, Felder</li> <li>- Mikroprozessorprogrammierung mit Arduino (optional für interessierte Studierende)</li> </ul> <p>Studierende beschäftigen sich mit diesen Konzepten in dem sie regelmäßig vorgegebene, relevante Aufgabenstellungen eigenständig lösen, programmieren und abgeben.</p>									
<b>Teilnahmevoraussetzungen</b>	Keine									
<b>Koordination</b>	Dr.-Ing. Volkan Gezer									
<b>Vortragende(r)</b>	Dr.-Ing. Volkan Gezer									
<b>Mitwirkende(r)</b>	-									
<b>Praktikumsstatus</b>	Keine									
Fachliteratur										
<b>Bücher / Skripte</b>	- Hartmut Ernst, Jochen Schmidt, Gerd Beneken. Grundkurs Informatik. Springer Viewek, 2016									
<b>Weitere Quellen</b>	- Helmut Erlenkötter. C: Programmieren von Anfang an. Rowohlt Taschenbuch Verlag, 1999.									
Lernmaterialien										

**STUDIENGANG INFORMATIK  
MODULBESCHREIBUNG**

Dokumente	-		
Hausaufgaben	-		
Prüfungen	-		
<b>Zusammensetzung des Moduls</b>			
Mathematik und Grundlagenwissenschaften	20	%	
Ingenieurwesen	20	%	
Konstruktionsdesign		%	
Sozialwissenschaften		%	
Erziehungswissenschaften		%	
Naturwissenschaften		%	
Gesundheitswissenschaften		%	
Fachkenntnis	60	%	
<b>Bewertungssystem</b>			
<b>Aktivität</b>	<b>Anzahl</b>	<b>Gewichtung in Endnote (%)</b>	
Zwischenprüfungen	1	40	
Quiz			
Hausaufgaben			
Anwesenheit			
Übung			
Projekte			
Abschlussprüfung	1	60	
	<b>Summe</b>	<b>100</b>	
<b>ECTS Leistungspunkte und Arbeitsaufwand</b>			
<b>Aktivität</b>	<b>Anzahl</b>	<b>Dauer</b>	<b>Gesamtaufwand (Stunden)</b>
Vorlesungszeit	14	2	28
Selbststudium	1	60	60
Hausaufgaben			
Präsentation / Seminarvorbereitung			
Zwischenprüfungen	1	3	3
Übung			
Labor	15	4	60
Projekte			
Abschlussprüfung	1	10	10
	<b>Summe Arbeitsaufwand</b>		<b>161</b>
	<b>ECTS Punkte (Gesamtaufwand / 28)</b>		<b>6</b>



## STUDIENGANG INFORMATIK MODULBESCHREIBUNG

### Lernergebnisse

1	Wissen wie unterschiedliche Arten von Daten in Rechnern dargestellt werden.
2	Kenntnisse über die Zahlenarithmetik in Rechnern
3	Kennen von Methoden zur fehlertoleranten, komprimierenden und verschlüsselnden Codierung
4	Eigenständiges Entwickeln von Algorithmen in Pseudo Code und Implementieren in der Programmiersprache C

### Wöchentliche Themenverteilung

1	Einführung in die Informatik, Geschichte, Datendarstellung in Rechnern
2	Zahlensysteme und Binärarithmetik
3	Programmierung in C (Grundbegriffe: Algorithmus, Ablaufdiagramm)
4	Programmierung in C (Datentypen, Variablen)
5	Programmierung in C (Mathematische und Logische Operatoren)
6	Programmierung in C (if-Anweisungen, Flusskontrolle)
7	Programmierung in C (goto-Schleifenaufbau)
8	Programmierung in C (Schleifen)
9	Zwischenprüfungen
10	Codierung und Verschlüsselung
11	Programmierung in C (Arrays und Strukturen)
12	Programmierung in C (Funktionen und Geltungsbereiche von Variablen)
13	Programmierung in C (Rekursive Funktionen)
14	Programmierung in C (Funktionen, Call-by-Value, Call-by-Reference)
15	Programmierung in C (Zeiger)

### Beitrag der Lernergebnisse zu den Lernzielen des Programms (1-5)

	P1	P2	P3	P4	P5	P6	P7
1	5	5	5			3	1
2	5	5	5			3	1
3	5	5	5			3	1
4	5	5	5			3	1

Beitragsgrad: 1: Sehr Niedrig 2: Niedrig 3: Mittel 4: Hoch 5: Sehr Hoch

<http://bm.tau.edu.tr/lernziele-des-programms>

Erstellt von: Ayşe Betül Yüce

Datum der Aktualisierung: 24.06.2022