



YAKIN DOĞU ÜNİVERSİTESİ DIŞA AÇIK DERSLER KOORDİNATÖRLÜĞÜ

Okul/Fakülte: MÜHENDİSLİK FAKÜLTESİ

Bölüm/Program: Bilgisayar Mühendisliği - İngilizce

Ders Dili:	English	Ders Kodu:	COM162
DersTürkçe İsmi:	PROGRAMLAMA VE SORU ÇÖZME		
Ders İngilizce İsmi:	PROGRAMMING PROBLEM SOLVING		
Dersi Verecek:	Yard .Doç. Dr. Hüseyin SEVAY		
Dersin Türü:	ZORUNLU	Dersin Seviyesi:	
Yıl	1	Semester	2
		Dönem:	2
Ders Kredisi:	4	AKTS Kredisi:	6
		Ders İşleme Biçimi:	YÜZ YÜZE
Teori(saat/hafta):	4,00	Uygulama(saat/hafta):	0,00
		Laboratuar(saat/hafta):	2,00
Dersin İçeriği:			

This course provides an introduction to fundamental concepts of programming and use of built-in data structures in solving problems using the Python general-purpose programming language. In this course, students study how to write user-defined functions using iteration as well as recursion. This course also stresses the importance of programming tools such as programming editors and debuggers. The students are expected to work within a GNU/Linux environment. The course provides a basic introduction into object-oriented programming.

1. Teach the fundamental principles of computer programming via the Python programming language
2. Teach a very basic introduction to object-oriented programming using Python
3. Teach how to write algorithms, and implement and test those algorithms
4. Convey that using fundamental programming-related tools well is essential to programming efficiently. These tools include the syntax of the Python language, a capable text editor suited for programming such as VIM/GVIM, the Python debugger, and the GNU/Linux operating system as a productive learning environment
5. Teach problem solving techniques and teach how to solve problems using programming
6. Teach that testing is an essential part of programming
7. Encourage students to incorporate all tools mentioned into their learning of programming.

Upon the completion of this course, a student will be able to:

1. Learn basic programming concepts and importance of testing software
2. Develop an understanding of how real-life problems can/may be solved using programming
3. Understand the important role programming plays in our lives
4. Write basic Python programs mainly using built-in data structures to solve problems
5. Develop basic algorithms in pseudo-code format for given problems
6. Write test code to test functions and methods in Python
7. Use input files and write output files in binary/text format
8. Recognize the breadth of Python libraries in solving many real-life problems
9. Understand basic object-oriented programming concepts
10. Understand the difference between mutable and immutable data types
11. How data structures are used to represent objects in memory
12. Understand how the stack is used in executing functions and methods
13. Understand scoping rules
14. Recognize the trade-off between space and time
15. Understand the importance of abstraction
16. Understand the concept of refactoring and code reuse
17. Understand the importance of right data structure selection for implementation

Öğrenme Kazanımları:

İlgili kavramları/kuramları anlayabilecek
İlgili kavram/kuramların geçerliliğini tartışabilecek
İlgili kavram/kuramların, gerçek hayattaki muhtemel uygulamalarını tartışabilecek ve öneriler sunabilecek
İlgili kavram/kuramları gerçek hayata/verilen diğer durumlara/vakalara uygulayabilecek
İlgili kavram/kuramların gerçek hayatta var olan uygulamalarını eleştirel olarak analiz edebilecek
Farklı kavram ve kuramları kendi özgün yaklaşımlarını yaratbilmek için sentezleyebilecek
İlgili kavramlarla ilgili özgün bir yaklaşım geliştirebilecek
Sunum(lara)a hazırlık
Verilen ölçütlere göre kendi çalışmalarını değerlendirebilecek
Verilen ölçütlere göre arkadaşlarının çalışmalarını değerlendirebilecek
Yeni yaklaşım geliştirebilecek/yaratabilecek
Verilen parametreler çerçevesinde yeni bir ürün geliştirebilecek/yaratabilecek
Verilen çalışmayı bağımsızca yürütebilecek
Verilen bir çalışma üzerinde grup halinde çalışabilecek
İlgili kavramları sayabilecek ve açıklayabilecek
Öğrenmenin değerini takdir edecek
Akademik bir makale üretmek için seçilen kaynak gösterme biçiminin ilkelerini uygulayabilecek
Hedeflenen becerileri geliştirebilecek
.

Dersin Amaçları:

Belirlenen kavram(ları) açıklamak/anlatmak
İlgili kavram(lar)la alakalı farkındalık yaratmak ve bunu geliştirmek.
Belirlenen kavram(lar)ın geçerliliğini tartışmak.
Seçilen/belirlenen becerileri geliştirmek
Seçilen konuların derinlemesine/detaylı bir şekilde incelemek
Belirlenen kavram/kuram/konularla ilgili öğrencilerin var olan bilgilerini geliştirmek
Seçilen kavramlar bağlamında öğrencilerin fikirlerini/bilgilerini/kavrayışlarını geliştirmek
Belirlenen kavram/kuram/konularla ilgili öğrencilerle var olan bilgilerini yenilemek
Yeniliği teşvik etmek
Eleştirel düşünceyi geliştirmek

Öğrenci İş Yüğü:

Ders saatleri
Ara sınav
Final sınavı
Sınıf içi tartışma(lar)
Ödev(ler)
Ders planlama
Materyal uyarlama
Materyal geliştirme

AKTS Formülü:

	Activities Quantity Duration (hour) Total Workload Course duration in class (including Exam weeks) 16 4 64 Labs and Tutorials Homework 5 3 15 Project/Presentation/Report E-learning activities Quizzes Midterm Examination Study 1 5 5 Final Examination Study 1 5 5 Self Study 40 2.5 100 Total Workload (hours) 189 Total Workload / 30 (hours) 6.3 ECTS Credit of the Course 6
Kaynaklar:	Python for Software Design: How to Think Like a Computer Scientist, Allen B. Downey, 2009, Cambridge University Press, and additional written resources.
Değerlendirme:	Midterm: 30% Long and short homeworks: 10% Lab: 15% Final : 45%
İşe Yerleştirme(Staj):	---
Ön Koşul Ders Kodları:	COM141
1. Hafta (19 – 23 Eylül)	Introduction to programming and programming tools
2. Hafta (26 – 30 Eylül)	Variables, expressions, & statements

3. Hafta (3 – 7 Ekim)	Functions
4. Hafta (10 – 14 Ekim)	Interface Design
5. Hafta (17 – 21 Ekim)	Conditionals & Recursion
6. Hafta (24 – 28 Ekim)	Advanced Functions
7. Hafta (31 - 4 Kasım)	Iteration
8. Hafta (7 - 11 Kasım)	Strings
9. Hafta (14 – 18 Kasım)	Solving Practical Problems
10. Hafta (21 – 25 Kasım)	Lists
11. Hafta (28 - 2 Aralık)	Dictionaries
12. Hafta (5 – 9 Aralık)	Tuples
13. Hafta (12 -16 Aralık)	Data Structure Selection
14. Hafta (19 - 23 Aralık)	Files
15. Hafta (24 – 30 Aralık)	FINAL SINAVLARI HAFTASI
16. Hafta	Classes & Objects
17. Hafta	
18. Hafta	
19. Hafta	
20. Hafta	
21. Hafta	
22. Hafta	
23. Hafta	
24. Hafta	
25. Hafta	
26. Hafta	
27. Hafta	
28. Hafta	
