**ITU**

**DERS KATALOG FORMU**

**(Course Catalogue Form)**

| **Dersin Adı:**  Algoritma Analizi II | **Course Name:**  Analysis of Algorithms II |
| --- | --- |

| **Kodu (Course Code)** | **Yarıyıl (Semester)** | **Kredisi (Local Credits)** | **AKTS Kredisi (ECTS Credits)** | **Ders Uygulaması, Saat/Hafta** | | |
| --- | --- | --- | --- | --- | --- | --- |
| **Ders (Theoretical)** | **Uygulama (Tutorial/Recitation)** | **Laboratuvar (Laboratory)** |
| BLG336E | 6 | 3 | 5 | 3 | - | - |

| **Bölüm/Program**  **(Department/Program)** | Bilgisayar Mühendisliği / Computer Engineering |
| --- | --- |

| **Dersin Türü**  **(Course Type)** | Temel Mühendislik  (Basic Engineering) | **Dersin Dili (Course Language)** | İngilizce  (English) |
| --- | --- | --- | --- |
| **Ders Zorunluluğu (Course Compulsion)** | | Zorunlu (Compulsory) | |

| **Dersin Önkoşulları (Course Prerequisites)** | BLG335/E Algorithm Analysis I  or  BLG381/E Advanced Data Structures | | | |
| --- | --- | --- | --- | --- |
| **Dersin Mesleki Bileşene Yüzde Katkısı**  **(Course Category by Content Percentage)** | Temel Bilim  (Basic Science) | Temel Mühendislik (Engineering Science) | Mühendislik Tasarım (Engineering Design) | İnsan ve Toplum Bilim (General Education) |
| 20% | 10% | 70% | - |

| **Dersin İçeriği (Course Description)** | Bu ders zaman iyi zaman performansına sahip algoritmaların tasarım ve performans analizlerinin öğrenilmesini hedefler. Basics of algorithm analysis. Asymptotic Notation; Graphs. Greedy algorithms; Divide and conquer; Dynamic programming; Network Flow; NP and computational intractability-I |
| --- | --- |
| This course aims to study the methods for designing efficient algorithms and to evaluate their performance (mainly in term of time). Algoritma Analizinin Temelleri; Asimptotik Notasyon; Çizgeler; Açgözlü Algoritmalar; Parçala-Yen; Dinamik Programlama ; Ağ Akışı; NP ve Hesapsal Çetinlik |
| **Dersin Amacı (Course Objective)** | 1. Algoritma alanında matematiksel düşünme pratiği vermek 2. Algoritma seviyesinde karmaşıklığın önemini anlamak 3. Değişik algoritma türlerini tanımak 4. Bazı problemler ve onların algoritmik çözümleri hakkında bilgi sahibi olmak. 5. Değişik algoritma türlerinin ortak özellikleri hakkında bilgi sahibi olmak. |
| 1. Practice mathematical thinking in the domain of algorithms. 2. Understand the importance of complexity at the algorithmic level. 3. Be familiar with different classes of algorithms. 4. Be familiar with various problems and their algorithmic solutions. 5. Be familiar with the common properties of the different classes of algorithms. |
| **Dersin Öğrenme Çıktıları (Course Learning Outcomes)** | 1. Verilen bir algoritma (ya da problemin) karmaşıklığını hesaplayabilmek 2. Verilen bir algoritmanın zaman karmaşıklığını hesaplayabilmek 3. Değişik amaçlar için kullanılan algoritmaların bazı hedefler ve durumlar için karmaşıklıklarını hesaplamak 4. Verilen bir problemi çözen bir yazılımı, belli bir performans kriterini göz önünde tutarak gerçekleme. |
| 1. Ability to formulate the complexity of a given algorithm (problem) . 2. Ability to analyze the complexity of a given algorithm in terms of time. 3. Ability to compare the complexities of some algorithms used for the same purpose under various conditions . 4. Ability to develop a program to solve a specific problem considering some performance criteria . |

| **Ders Kitabı (Textbook)** | Introduction to Algorithms , 2nd Edition,T.H.Cormen, C.E. Leiserson, R.L. Rivest, MIT J. Kleinberg and E. Tardos, Algorithm Design, Addison Wesley, 2006. |
| --- | --- |
| **Diğer Kaynaklar (Other References)** | Introduction to Algorithms, Cormen, Leiserson and Rivest, The MIT Pres/McGraw-Hill. |

| **Ödevler ve Projeler (Homeworks & Projects)** | C++’da projeler. |
| --- | --- |
| Projects using C++ |
| **Laboratuvar Uygulamaları (Laboratory Work)** | - |
| - |
| **Bilgisayar Kullanımı (Computer Use)** | Zorunlu |
| Compulsory |
| **Diğer Uygulamalar (Other Activities)** | - |
| - |

| **Başarı Değerlendirme Sistemi**  **(Assessment Criteria)** | **Faaliyetler (Activities)** | **Adedi (Quantity)** | **Değerlendirmedeki Yüzde Katkısı**  **(Effects on Grading by Percentage)** |
| --- | --- | --- | --- |
| **Yıl İçi Sınavları (Midterm Exams)** | - | - |
| **Kısa Sınavlar (Quizzes)** | - | - |
| **Ödevler (Homework)** | 3 | 75% |
| **Projeler (Projects)** | - | - |
| **Dönem Ödevi/Projesi (Term Paper/Project)** | - | - |
| **Laboratuvar Uygulaması (Laboratory Work)** | - | - |
| **Diğer Uygulamalar (Other Activities)** | - | - |
| **Final Sınavı (Final Exam)** | 1 | 25% |

**DERS PLANI**

**(Course Plan)**

| **Hafta** | **Konu** | **Dersin Çıktıları** |
| --- | --- | --- |
| **1** | Giriş, örnek problemler | 4 |
| **2** | Giriş, örnek problemler | 4 |
| **3** | Algoritma Analizinin Temelleri | 4 |
| **4** | Çizgeler | 1,2 |
| **5** | Açgözlü Algoritmalar-I | 1,2 |
| **6** | Açgözlü Algoritmalar-II | 1,2,3 |
| **7** | Parçala-Yen-I | 1,2,3 |
| **8** | Parçala-Yen-II | 1,2,3 |
| **9** | Dinamik Programlama | 1,2 |
| **10** | Ağ Akışı-I | 1,2 |
| **11** | Ağ Akışı-II | 1,2,3 |
| **12** | NP ve Hesapsal Çetinlik-I | 1,2 |
| **13** | NP ve Hesapsal Çetinlik-I | 1,2,3 |
| **14** | Genel tekrar | 1, 2, 3, 4 |

| **Week** | **Topic** | **Course Outcome** |
| --- | --- | --- |
| **1** | Introduction. Some representative problems. | 4 |
| **2** | Introduction. Some representative problems. | 4 |
| **3** | Basics of algorithm analysis. | 4 |
| **4** | Graphs. | 1,2 |
| **5** | Greedy algorithms-I. | 1,2 |
| **6** | Greedy algorithms-II. | 1,2,3 |
| **7** | Divide and conquer-I | 1,2,3 |
| **8** | Divide and conquer-II | 1,2,3 |
| **9** | Dynamic programming. | 1,2 |
| **10** | Network Flow-I | 1,2 |
| **11** | Network Flow-II | 1,2,3 |
| **12** | NP and computational intractability-I | 1,2 |
| **13** | NP and computational intractability-II | 1,2,3 |
| **14** | Course Recap | 1, 2, 3, 4 |

**DERSİN BİLGİSAYAR MÜHENDİSLİĞİ ÖĞRENCİ ÇIKTILARI İLE İLİŞKİSİ**

**Relationship between the Course and Student Outcomes**

**(1: “Little”, 2: “Partial”, 3: “Full”, Leave blank if your answer is “None”)**

| **Computer Engineering Department Program Outcomes and Performance Criteria** | | **Level of Contribution** | | |
| --- | --- | --- | --- | --- |
| **1** | **2** | **3** |
| 1 | an ability to identify, formulate, and solve complex engineering problems by applying principles of engineering, science, and mathematics |  |  | X |
| 2 | an ability to apply engineering design to produce solutions that meet specified needs with consideration of public health, safety, and welfare, as well as global, cultural, social, environmental, and economic factors |  | X |  |
| 3 | an ability to communicate effectively with a range of audiences | X |  |  |
| 4 | an ability to recognize ethical and professional responsibilities in engineering situations and make informed judgments, which must consider the impact of engineering solutions in global, economic, environmental, and societal contexts | X |  |  |
| 5 | an ability to function effectively on a team whose members together provide leadership, create a collaborative and inclusive environment, establish goals, plan tasks, and meet objectives | X |  |  |
| 6 | an ability to develop and conduct appropriate experimentation, analyze and interpret data, and use engineering judgment to draw conclusions |  | X |  |
| 7 | an ability to acquire and apply new knowledge as needed, using appropriate learning strategies |  |  | X |

**HAZIRLANMA BİLGİSİ**

**Edition Information**

| **Prepared by** | **Date** | **Signature** |
| --- | --- | --- |
| **Dr.İlkay Öksüz** | **24.11.2020** |  |
| **Approved by** | **Date** | **Signature** |
| **Dr.Tolga Ovatman** | **26.11.2020** |  |