

GENEL TANIM / GENERAL DESCRIPTION

Ders Adı / Course Name	Distributed Algorithms / Distributed Algorithms	
Ders Kodu / Course Code	9105055452012	
Ders Türü / Course Type		
Ders Seviyesi / Course Level	Second Cycle / Second Cycle	
Ders Akts Kredi / ECTS	8.00	
Haftalık Ders Saati (Kuramsal) / Course Hours For Week (Theoretical)	3.00	
Haftalık Uygulama Saati / Course Hours For Week (Objected)	0.00	
Haftalık Laboratuar Saati / Course Hours For Week (Laboratory)	0.00	
Dersin Verildiği Yıl / Year	1	
Öğretim Sistemi / Teaching System	Face to Face / Face to Face	
Eğitim Dili / Education Language	Turkish / Turkish	
Ön Koşulu Olan Ders(ler) / Precondition Courses	Yok	None
Amacı / Purpose	Bu dersin amacı öğrencilerin; dağıtık algoritmaları kavramaları, analiz edebilmeleri, dağıtık algoritmaların doğrulukları ve performansları hakkında yorum yapabilmeleri, limitlerini kavrayabilmelerini ve yeni dağıtık algoritmalar geliştirebilmelerini sağlamaktır.	The objectives of this course are to teach students distributed algorithm concepts, theoretical analysis, proof of correctness and performance analysis. Also the course aims to design new distributed algorithms.
İçeriği / Content	Dağıtık çizge algoritmaları: Dağıtık hesaplama modelleri, çizge teorisine ve çizge teorik algoritmaların tekrarı, köşe ve ağaç kapsama (cover) algoritmaları, dağıtık herkese gönderim (broadcast), kapsayan ağaç algoritmaları, GHS algoritması, dağıtık DFS ve dağıtık BFS algoritmaları, eşleme, bağımsız küme ve hâkim küme, kümeleme. Temel dağıtık algoritmalar: Zaman senkronizasyonu, dağıtık karşılıklı dışlama algoritmaları, sonlanma tespiti, dağıtık ortamlarda ölü-kilit problemleri ve örnek çözüm algoritmaları, anlaşma protokolleri, oto-stabilizasyon.	Distributed graph algorithms: Distributed computing models, revisiting of graph theory and graph algorithms, edge and vertex cover algorithms, distributed broadcast, spanning tree algorithms, GHS algorithms, distributed DFS and distributed BFS algorithm, matching, independent set and dominating set, clustering. Fundamental Distributed Algorithms: Time synchronization, distributed mutual exclusion, termination detection, distributed deadlock problems and example algorithms, consensus protocols, self-stabilization.
Önerilen Diğer Hususlar / Recommended Other Considerations	Yok	None
Staj Durumu / Internship Status	Yok	None

Kitap / Malzemesi / Önerilen Kaynaklar / Books / Materials / Recommended Reading	1.Sukumar Ghosh, Distributed Systems : An Algorithmic Approach, Chapman and Hall, (2006) . 2.Vijay Garg, Elements of Distributed Computings, John Wiley, (2002). 3.Gerard Tel, Introduction to Distributed Algorithms, 2nd Ed., Cambridge University Press, (2000). 4.David Peleg, Distributed Algorithms, Locality Sensitive Approach, SIAM, Philadelphia, PA, (2000). 5.Jon Kleinberg and Eva Tardos, Algorithm Design, Addison Wesley, (2006). 6.Douglas West, Introduction to Graph Theory, 2nd Edition, Prentice-Hall, (2000). 7. J.Gross and J. Yellen, Graph Theory and Its Applications, 2nd Edition, Chapman & Hall, (2005).	1.Sukumar Ghosh, Distributed Systems : An Algorithmic Approach, Chapman and Hall, (2006) . 2.Vijay Garg, Elements of Distributed Computings, John Wiley, (2002). 3.Gerard Tel, Introduction to Distributed Algorithms, 2nd Ed., Cambridge University Press, (2000). 4.David Peleg, Distributed Algorithms, Locality Sensitive Approach, SIAM, Philadelphia, PA, (2000). 5.Jon Kleinberg and Eva Tardos, Algorithm Design, Addison Wesley, (2006). 6.Douglas West, Introduction to Graph Theory, 2nd Edition, Prentice-Hall, (2000). 7. J.Gross and J. Yellen, Graph Theory and Its Applications, 2nd Edition, Chapman & Hall, (2005).
Öğretim Üyesi (Üyeleri) / Faculty Member (Members)	Doç. Dr. Orhan Dağdeviren	

ÖĞRENME ÇIKTILARI / LEARNING OUTCOMES

1	1.Dağıtık Algoritma kavramını anlayabilmek.	1.Understanding distributed algorithm concept
2	2.Problemlere farklı bir bakış açısıyla dağıtık bir şekilde çözüm üretebilmek.	2.Designing distributed algorithm approaches for problems.
3	3.Bilinen dağıtık problemleri ve çözümler hakkında bilgi sahibi olmak.	3.Acquiring the knowledge of distributed problems and solutions.
4	4.Dağıtık algoritmaların analizini yapabilmek.	4.Analyzing distributed algorithms.
5	5.Dağıtık ortamda oluşabilecek problemleri fark edebilmek.	5.Realizing problems on distributed environments.
6	6.Problemleri dağıtık bir şekilde düşünerek dağıtık algoritmasını tasarlayabilmek.	6.Learning distributed algorithm design methods.
7	7.Dağıtık yazılım geliştirmek için çağdaş teknolojileri kullanabilmek.	7.Using modern technologies to develop distributed software.
8	8.Bilinen seri çalışan bir algoritmayı dağıtık hale çevirebilmek.	8.Designing the distributed version of central algorithms.

HAFTALIK DERS İÇERİĞİ / DETAILED COURSE OUTLINE

Hafta / Week					
1	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Dağıtık sistemlerin tanımı, dağıtık uygulama örnekleri, mesaj gönderimi modelleri, paylaşımlı bellek modeli.	Programlama Ödevi			
	Distributed system definition, distributed application example, message transmission models, shared memory models.	Programming Homework			
2	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Sonlu durum makineleri, görevler, threadler, thread çeşitleri ve semafor örnekleri.				
	Finite state machines, threads, thread types, semaphore examples.				
3	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Çizge teorisi ve temel çizge algoritmaları: BFS algoritması, DFS algoritması, çizge bağlılığı, topolojik sıralama algoritması.	Programlama Ödevi			
	Graph theory, fundamental graph algorithms: BFS algorithm, DFS algorithm, connectivity, topological sorting.	Programming Homework			
4	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Köşe ve ağaç boyama algoritmaları, dağıtık köşe ve ağaç boyama algoritması, dağıtık ağaç tabanlı algoritmalar: broadcast (herkese gönderim) ve convergecast (herkesten toplama) algoritmaları.				
	Vertex and tree coloring algorithms, distributed vertex and tree coloring algorithms, distributed spanning tree algorithms: broadcast and convergecast algorithms				
5	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	BFS ağaç oluşturma, yayma (flooding), seri ve dağıtık MST algoritmaları, döngü (cycle) ve kesim (cut) kavramları.	Programlama Ödevi			
	BFS tree construction, flooding, serial and distributed MST algorithms, cycle and cut concepts.	Programming Homework			

	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
6	Dağıtık sistemlerde zaman senkronizasyonu, Berkeley zaman protokolü, mantıksal saatler, Lamport'un mantıksal saat algoritması, vektör saatleri, matris saatleri	Araştırma Ödevi			
	Distributed time synchronization, Berkeley time synchronization protocol, logical clocks, Lamport's logical clock algorithm, vector clocks, matrix clocks	Research Homework			
7	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Kaynak paylaşırma, kritik bölge problemi, donanımsal senkronizasyon, semaforlar, semafor problem örnek kod incelemesi.	Literatür Tarama			
	Resource sharing, critical section problem, hardware synchronization, semaphores.	Literature Review			
8	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Ara Sınav				
	Midterm				
9	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Karşılıklı dışlama algoritmaları: Lamport'un algoritması, Ricart-Agrawala'nın algoritması, Suzuki-Kasami'nin algoritması, Raymond'ın algoritması, Maekawa'nın algoritması.	Araştırma Ödevi			
	Mutual Exclusion Algorithms: Lamport's algorithm, Ricart-Agrawala's algorithm, Suzuki-Kasami's algorithm, Raymond's algorithm, Maekawa's algorithm.	Research Homework			
10	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Dağıtık sistemlerin global durumu, Chandy Lamport'un algoritması, Lai Yang'ın algoritması, snapshot.	Literatür Tarama			
	Global states of distributed systems, Chandy Lamport's algorithm, Lai Yang'ın algorithm, snapshot.	Literature Review			
11	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Ölü kilitler ve sonlanma tespiti, Dijkstra-Scholten algoritması, dağıtık ölü kilit, Chandy Misra Haas ölü kilit önleme algoritması.	Programming Homework			
	Deadlocks and termination detection, Dijkstra-Scholten algorithm, distributed deadlock, Chandy Misra Haas deadlock prevention algorithm.	Programming Homework			

12	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Lider seçimi algoritmaları: Bully algoritması, LeLann'ın algoritması, Chang Roberts algoritması, Senkronizörler.				
	Leader election algorithms: Bully algorithm, LeLann algorithm, Chang Roberts algorithm, Synchronizers.				
13	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Anlaşma protokolleri, Bizans generalleri problemi, konsensus.				
	Consensus protocols, byzantine generals problems, consensus.				
14	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Anlaşma protokolleri, oto stabilizasyon.				
	Self-stabilization				
15	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Proje Sunumları				
	Project Presentations				
16	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Final Sınavı				
	Final Exam				

DEĞERLENDİRME / EVALUATION

Yarıyıl (Yıl) İçi Etkinlikleri / Term (or Year) Learning Activities	Sayı / Number	Katkı Yüzdesi / Percentage of Contribution (%)
Ara Sınav / Midterm Examination	1	100
Toplam / Total:	1	100
Başarı Notuna Katkı Yüzdesi / Contribution to Success Grade(%):		40

Yarıyıl (Yıl) Sonu Etkinlikleri / End Of Term (or Year) Learning Activities	Sayı / Number	Katkı Yüzdesi / Percentage of Contribution (%)
Final Sınavı / Final Examination	1	100
Toplam / Total:	1	100
Başarı Notuna Katkı Yüzdesi / Contribution to Success Grade(%):		60

Etkinliklerinin Başarı Notuna Katkı Yüzdesi(%) Toplamı / Total Percentage of Contribution (%) to Success Grade:	100
Değerlendirme Tipi / Evaluation Type:	

İŞ YÜKÜ / WORKLOADS

Etkinlikler / Workloads	Sayı / Number	Süresi (Saat) / Duration (Hours)	Toplam İş Yüğü (Saat) / Total Work Load (Hour)
Ara Sınav / Midterm Examination	1	3.00	3.00
Final Sınavı / Final Examination	1	3.00	3.00
Derse Katılım / Attending Lectures	14	3.00	42.00
Rapor Hazırlama / Report Preparation	8	3.00	24.00
Proje Hazırlama / Project Preparation	9	4.00	36.00
Proje Sunma / Project Presentation	1	3.00	3.00
Ödev Problemleri için Bireysel Çalışma / Individual Study for Homework Problems	7	7.00	49.00
Ara Sınav için Bireysel Çalışma / Individual Study for Mid term Examination	1	21.00	21.00
Final Sınavı için Bireysel Çalışma / Individual Study for Final Examination	1	30.00	30.00
Okuma / Reading	7	3.00	21.00
Toplam / Total:	50	80.00	232.00

Dersin AKTS Kredisi = Toplam İş Yüğü (Saat) / 30.00 (Saat/AKTS) = 232.00/30.00 = 7.73 ~ / Course ECTS Credit = Total Workload (Hour) / 30.00 (Hour / ECTS) = 232.00 / 30.00 = 7.73 ~

PROGRAM VE ÖĞRENME ÇIKTISI / PROGRAM LEARNING OUTCOMES

Öğrenme Çıktıları / Learning Outcomes	Program Çıktıları / Program						
	1.1.1	1.1.2	1.1.3	1.1.4	1.1.5	1.1.6	1.1.7
1.1.Dağıtık Algoritma kavramını anlayabilmek. / 1.Understanding distributed algorithm concept							
2.2.Problemlere farklı bir bakış açısıyla dağıtık bir şekilde çözüm üretebilmek. / 2.Designing distributed algorithm approaches for problems.							
3.3.Bilinen dağıtık problemleri ve çözümler hakkında bilgi sahibi olmak. / 3.Acquiring the knowledge of distributed problems and solutions.							
4.4.Dağıtık algoritmaların analizini yapabilmek. / 4.Analyzing distributed algorithms.							
5.5.Dağıtık ortamda oluşabilecek problemleri fark edebilmek. / 5. Realizing problems on distributed environments.							
6.6.Problemleri dağıtık bir şekilde düşünerek dağıtık algoritmasını tasarlayabilmek. / 6.Learning distributed algorithm design methods.							
7.7.Dağıtık yazılım geliştirmek için çağdaş teknolojileri kullanabilmek. / 7.Using modern technologies to develop distributed software.							
8.8.Bilinen seri çalışan bir algoritmayı dağıtık hale çevirebilmek. / 8. Designing the distributed version of central algorithms.							

Katkı Düzeyi / Contribution Level : 1-Çok Düşük / Very low, 2-Düşük / Low, 3-Orta / Moderate, 4-Yüksek / High, 5-Çok Yüksek / Very high