

Öğrenci No		Bölüm	Bilgisayar Mühendisliği		
Adı Soyadı		Yıl / Yarıyıl	2025-2026	<input type="checkbox"/> Güz	<input checked="" type="checkbox"/> Bahar
İmza		Tarih / Saat / Süre	09/06/2026	10.00	60 dakika
Dersin Kodu / Adı	BILM-202 ALGORİTMALAR	Sınav Türü	<input type="checkbox"/> Ara Sınav	<input checked="" type="checkbox"/> Final	<input type="checkbox"/> Bütünleme
Dersin Sorumlusu	Dr. Öğr. Üyesi Sercan KÜLCÜ		<input type="checkbox"/> Mazeret	<input type="checkbox"/> Diğer:	

Öğrenme Çıktıları	ÖÇ5	ÖÇ4	ÖÇ4	Toplam
Sorular	1	2	3	3 Soru
Puanlar	30	20	50	100 Puan
Alınan Puan				

Sınavla İlişkin Açıklamalar

- Sınavın ilk 30 dakikasında salondan çıkmak yasaktır.
- Sınavda cep telefonu, akıllı saat vb. kullanmak; kalem, silgi alışverişi yapmak yasaktır.
- Cevaplarınız adım adım, anlaşılır ve düzenli olmalıdır. Anlaşılmayan düzensiz cevaplar puanlanmayacaktır.
- Sınav kurallarına mutlaka uyunuz. Sınav kurallarına uymayanlar hakkında disiplin soruşturması açılacaktır.

SORULAR

Soru 1: Metin: “gel de geleyim”, Örüntü: “geleyim”. Metin içerisinde örüntü, Brute Force Naive, Knuth Morris Pratt ve Boyer Moore algoritmalarına göre aratılmıştır.

- Her bir algoritma toplamda kaç karşılaştırma yaparak sonuca ulaşır?
- Algoritmaların çalışma mantıklarını açıklayınız.

Brute Force / Naive: Toplam 17 karşılaştırma yapar.

KMP: Arama sırasında 15 karşılaştırma yapar.

Boyer-Moore: Toplam 8 karşılaştırma yapar.

Brute Force: Metindeki her konumdan başlayarak örüntüyü karakter karakter karşılaştırır. Uyumsuzluk olursa bir sonraki konuma geçer.

KMP: Önce LPS tablosu oluşturur. Uyumsuzluk olduğunda örüntüyü baştan başlatmak yerine LPS tablosuna göre kaydırır. Böylece gereksiz karşılaştırmaları azaltır.

Boyer-Moore: Örüntüyü sağdan sola karşılaştırır. Uyumsuzlukta kötü karakter ve iyi sonek kurallarına göre büyük kaydırmalar yapabilir. Uzun metinlerde genellikle çok etkilidir.

Soru 2: Aşağıdaki ifadelerin doğru mu yanlış mı olduğunu belirtiniz.

- BFS, ağırlıksız çizgelerde en kısa yolu bulmak için kullanılabilir.

Doğru. BFS, ağırlıksız çizgelerde en kısa yolu bulmak için kullanılabilir.

- DFS her zaman en kısa yolu bulur.

Yanlış. DFS her zaman en kısa yolu bulmaz.

- Kruskal algoritması minimum kapsayan ağaç bulmak için kullanılır.

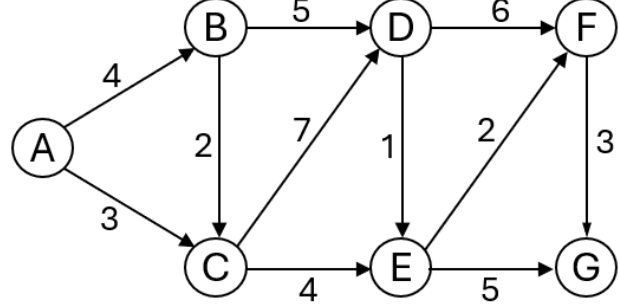
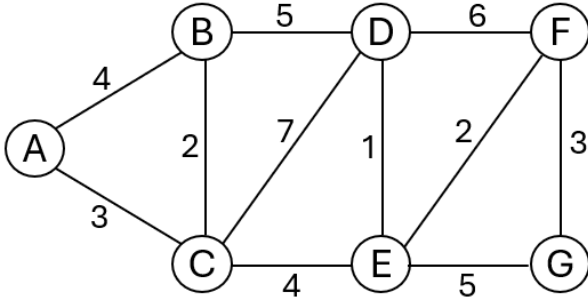
Doğru. Kruskal algoritması minimum kapsayan ağaç bulmak için kullanılır.

- Dijkstra algoritması negatif ağırlıklı kenarlarda güvenli şekilde çalışır.

Yanlış. Dijkstra negatif ağırlıklı kenarlarda güvenli şekilde çalışmaz.

- Yönsüz bir çizgede her kenar iki düğüm arasında çift yönlü bağlantı ifade eder.

Doğru. Yönsüz çizgede her kenar iki düğüm arasında çift yönlü bağlantı ifade eder.



Soru 3: Aşağıdaki soruları yukarıda verilen çizgelere göre cevaplayınız. Maksimum akış sorusunda yönlü çizmeyi esas alınız. Diğer sorularda yönsüz çizmeyi esas alınız.

a) Düğüm sayısını ve kenar sayısını bulunuz.

Düğüm sayısı: 7 Kenar sayısı: 11

b) Başlangıç düğümü A olsun. Komşular alfabetik sıraya göre gezilecektir. Buna göre BFS gezinti sırasını yazınız.

$A \rightarrow B \rightarrow C \rightarrow D \rightarrow E \rightarrow F \rightarrow G$

c) Başlangıç düğümü A olsun. Komşular alfabetik sıraya göre gezilecektir. Buna göre DFS gezinti sırasını yazınız.

$A \rightarrow B \rightarrow C \rightarrow D \rightarrow E \rightarrow F \rightarrow G$

d) Dijkstra algoritmasını kullanarak A düğümünden G düğümüne en kısa yolu/yolları ve maliyetini bulunuz.

$A \rightarrow C \rightarrow E \rightarrow G$

$A \rightarrow C \rightarrow E \rightarrow F \rightarrow G$

Toplam maliyet: 12

e) İsteddiğiniz algoritmayı kullanarak verilen çizgenin minimum kapsayan ağacı ve toplam ağırlığını bulunuz. Prim ve Kruskal algoritmalarının çalışma mantığını karşılaştırınız.

MST kenarları: D – E, B – C, E – F, A – C, F – G, C – E,

Minimum kapsayan ağaç toplam ağırlığı: $1 + 2 + 2 + 3 + 3 + 4 = 15$

Prim algoritması, bir başlangıç düğümünden başlar ve ağacı adım adım büyütür. Her adımda ağaca en küçük ağırlıklı uygun kenarı ekler.

Kruskal algoritması, tüm kenarları küçükten büyüğe sıralar. Döngü oluşturmayan en küçük kenarları seçerek minimum kapsayan ağacı oluşturur.

f) İsteddiğiniz algoritmayı kullanarak A düğümünden G düğümüne maksimum akışı bulunuz. Artırma yollarını yazınız. Her artırma yolundan gönderilen akış miktarını belirtiniz. Maksimum akış değerini bulunuz.

1. Artırma yolu $A \rightarrow C \rightarrow E \rightarrow G$ Kapasiteler: 3, 4, 5 Gönderilen akış: 3

2. Artırma yolu $A \rightarrow B \rightarrow D \rightarrow F \rightarrow G$ Kapasiteler: 4, 5, 6, 3 Gönderilen akış: 3

3. Artırma yolu $A \rightarrow B \rightarrow D \rightarrow E \rightarrow G$ Kalan kapasiteler üzerinden gönderilen akış: 1

Toplam maksimum akış: $3 + 3 + 1 = 7$