Graph

Çizge (graph), düğümlerle bu düğümleri birbirine bağlayan kenarlardan oluşan ve ağ

görünümünde olan bir tür veri yapısıdır. Temel olarak birbirleriyle ilişkili verileri tem-

sil etmek için kullanılırlar. Çizgelerin ulaşım, bilgisayar ağları, elektrik devreleri, sosyal

ağlar gibi günlük hayattaki birçok alanda uygulamaları mevcuttur. Örneğin, bir sosyal

ağ uygulamasını göz önüne aldığımızda, burada kullanıcılar ve kullanıcılar arasında ar-

kadaşlık ilişkilerinden bahsedebiliriz. Her bir kullanıcı ayrı bir düğüm noktası olarak ele

alındığında, bazı düğümler diğer düğümlerle bağlantılı durumda olacaktır. Herhangi bir

çizgenin içerdiği düğümler, özel arama algoritmalarıyla bir noktadan başlanarak belirli

sırada ziyaret edilebilmekte ve listelenebilmektedir. Çizgeler üzerinde bu işlemin yapıla-

bilmesi için temel olarak enine arama (breadth-first search) ve önce derinliğine arama

(depth-first search) algoritmaları mevcuttur. Bunun yanısıra, çizgelerin içerdiği düğüm

noktaları arasındaki en kısa mesafelerin hesaplanması da söz konusu olabilir. Örneğin,

bir şehirden başka bir şehire araçla en hızlı şekilde ulaşılmak istenildiğini varsayalım. Bu

durumda, ilgili haritadaki tüm şehirler birer düğüm noktası olur ve belli düğüm noktaları

arasında bağlantılar bulunur. Bu konuda uygun çözümlere ulaşmak için çeşitli “en kısa

yol bulma” algoritmaları mevcuttur. Ünitemizde, bu tipte en yaygın algoritmalardan birisi

olan Dijkstra en kısa yol algoritması anlatılmaktadır. Bu algoritma, çizgenin içerisindeki

herhangi bir düğüm noktasından diğer tüm düğüm noktalarına olan en kısa mesafeleri

hesaplamak ve bu mesafelerin elde edilmesi için izlenecek yolları belirlemek için kullanı-

lır.

ÇİZGELERLE İLGİLİ TEMEL KAVRAMLAR

Çizgeler, matematiksel olarak aşağıdaki formüldeki gibi temsil edilirler. Bu formülde, Ç

ifadesi çizgeyi, D ve K ifadeleri ise sırasıyla düğümleri ve kenar bağlantılarını temsil et-

mektedir.

Ç = (D, K)

Çizgeler, düğümler arasındaki kenar bağlantılarının tipine göre yönlü çizge (directed

graph) ve yönsüz çizge (undirected graph) olmak üzere ikiye ayrılırlar.

Yönlü çizge (directed graph):

Yönlü çizge, kenar bağlantılarının yönleri temsil eden oklarla gösterildiği çizgedir.

Çizge içerisindeki birbirine bağlı iki düğüm noktası arasında, sadece ilgili okun işaret

ettiği yönde ilerlenebilmesi mümkündür.

Yönsüz çizge (undirectedgraph):

Yönsüz çizge, kenar bağlantılarının yönleri temsil eden oklar ile gösterilmediği çift yönlü

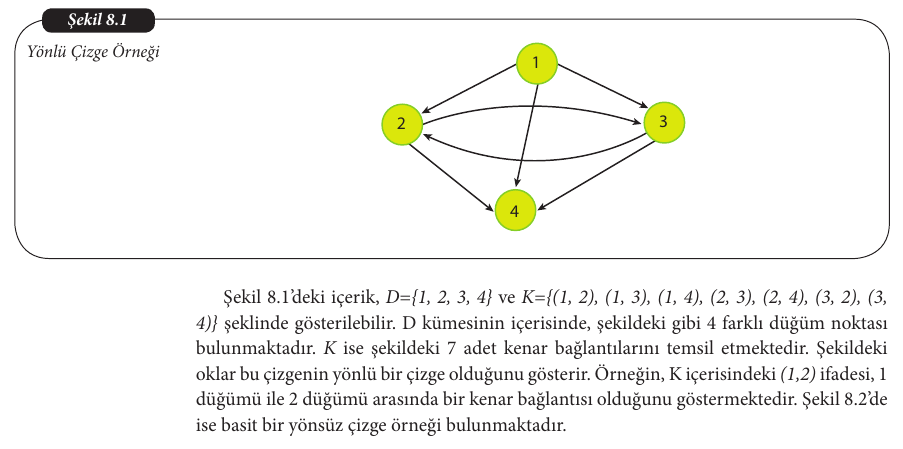
olan çizgedir. Çizge içerisindeki birbirine bağlı iki düğüm noktası arasında her iki yönde de

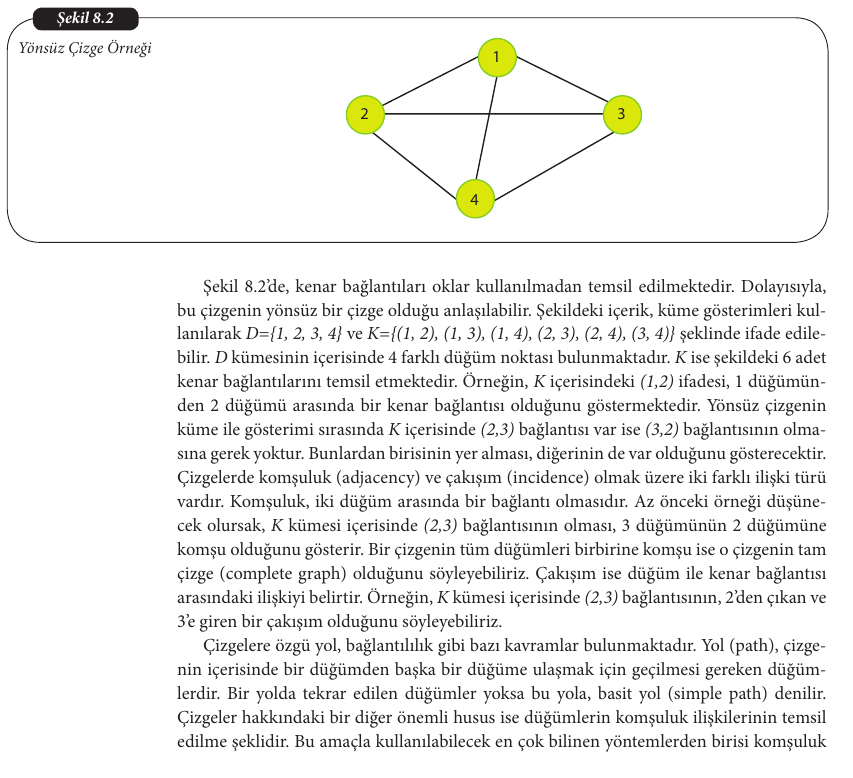
ilerlenebilmesi mümkündür. Bir anlamda, düğümler arasındaki bağlantıların simetrik olduğu

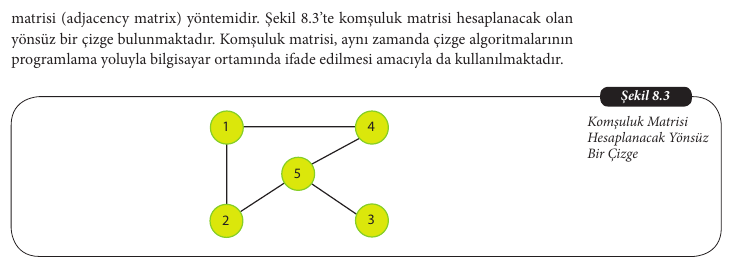
söylenebilir.

Ağırlıklandırılmış çizge(weighted graph):

Düğümler arasındaki kenar bağlantıları üzerinde sıfırdan farklı sayısal değerlerin yer aldığı çizge türüdür







ENİNE ARAMA ALGORİTMASI

Enine arama, çizgenin bir düğümünden başlanarak, söz konusu düğümün komşu dü-

ğümlerinin ve onların da komşularının sırayla ziyaret edildiği arama algoritmasıdır. Bu

algoritmanın çalışması sırasında, öncelikle başlangıç düğümünün tüm komşuları ziyaret

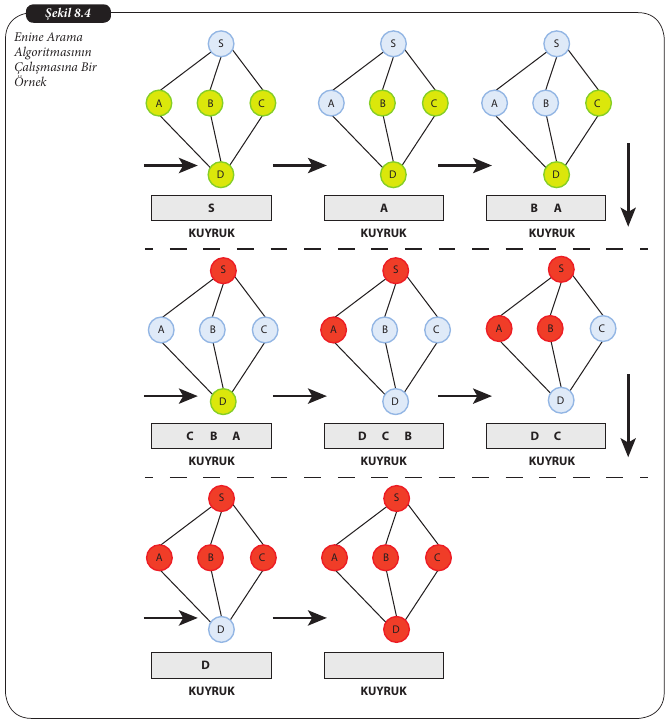
edilir. Daha sonra, başlangıç düğümünün komşuları ile komşu olan düğümlerden devam

edilir. Algoritmanın uygulanması esnasında kuyruk (queue) veri yapısından faydalanılır.

Şekil 8.4’te enine arama algoritmasının çalışmasına bir örnek verilmiştir. Bu örnekte, bir

düğümün birden fazla komşusu olduğunda bu komşular alfabetik sırada küçükten büyüğe

doğru ziyaret edilmektedir.



ÖNCE DERİNLİĞİNE ARAMA ALGORİTMASI

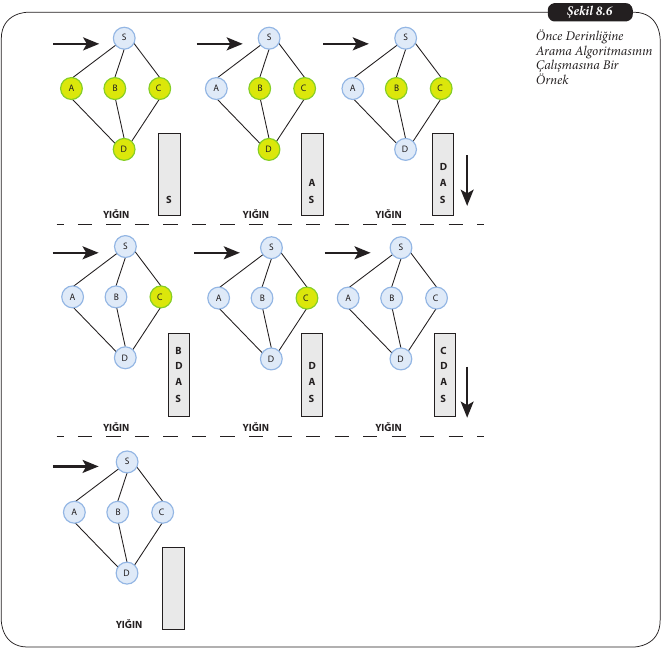
Önce derinliğine arama, çizgenin bir düğümünden başlanarak bu düğümün komşusu

üzerinden gidilebilecek en uzak düğüme kadar olan noktaların ziyaret edildiği ve daha

sonra geri dönülerek aynı işlemlerin ziyaret edilmemiş düğümler için sürdürüldüğü bir

arama algoritmasıdır. Algoritmanın uygulanması esnasında yığın (stack) veri yapısından

faydalanılır. Şekil 8.6’da enine arama algoritmasının çalışmasına bir örnek verilmiştir.



DİJKSTRA EN KISA YOL ALGORİTMASI

Dijkstra algoritması, ağırlıklandırılmış çizgelerde bir başlangıç düğümü ile diğer düğüm-

ler arasındaki en kısa mesafeyi tespit etmek için kullanılır. Bu algoritmanın amacını açık-

lamak için örnek olarak Türkiye haritasındaki şehirlerin bir çizge üzerinde düğümler ola-

rak gösterildiğini varsayalım. Düğümler arasındaki ağırlıklandırılmış kenar bağlantıları

ise her iki şehir çifti arasındaki uzaklığı temsil edecektir. Bu bilgilerin yardımıyla Dijkstra

algoritması kullanılarak, belirtilen bir şehir ile diğer bütün şehirler arasındaki en kısa yol-

lar bulunabilir. Sonuçta, çıktı olarak farklı şehirler arasındaki en kısa mesafeler ve izlen-

mesi gereken yollar elde edilecektir. Bu algoritmanın başlangıcında, başlangıç düğümü ile

diğer düğümler arasındaki uzaklıkların tamamının sonsuz olduğu varsayılır. Algoritma,

düğümler arasındaki en kısa yolları aradığı için düğümler arasında daha kısa yollar bulun-

dukça, sonsuz değeri ilgili yolun uzunluğu ile değiştirilir. Şekil 8.8’de Dijkstra algoritması

ile en kısa yolun bulunacağı bir çizge örneği verilmiştir. Bu örnekte, 0 düğümü ile diğer

düğümler arasındaki en kısa yol aranacağı için 0 düğümü mavi ile gösterilmektedir.

