**T.C. İSTANBUL Okan Üniversitesi**

**Meslek Yüksek Okulu**

**Elektronik ve Otomasyon Bölümü**

**Mobil Teknolojileri Programı**

****

**YAPAY ZEKA**

**PROJE ÖDEVİ**

**PROJE İSMİ : EN KISA YOL PROBLEMİ**

**Hazırlayan: GÜLER FATMA AÇIKGÖZ**

İçindekiler

[GİRİŞ (ARAÇLAR) 3](#_Toc38554583)

[VISUAL STUDIO 3](#_Toc38554584)

[PROGRAMLAMA DİLİ: C++ 3](#_Toc38554585)

[GELİŞME 3](#_Toc38554586)

[EN KISA YOL PROBLEMİ 3](#_Toc38554587)

[DIJKSTRA ALGORİTMASI 4](#_Toc38554588)

[BELLMAN VE FORD 4](#_Toc38554589)

[FLOYD ALGORİTMASI 4](#_Toc38554590)

[SONUÇ 4](#_Toc38554591)

[ÇÖZECEĞİMİZ PROBLEM 4](#_Toc38554592)

[KODLAR 5](#_Toc38554593)

[EKRAN ÇIKTISI 6](#_Toc38554594)

[KAYNAK 6](#_Toc38554595)

# GİRİŞ (ARAÇLAR)

## VISUAL STUDIO

Birçok programlama dilini destekleyen bir uygulama geliştirme platformudur. Bu platform içerisinde uygulama, program veya web sitesi yapılabilir. Visual Studio, Windows API, Windows Forms, Windows Presentation Foundation, Windows Store ve Microsoft Silverlight gibi Microsoft yazılım geliştirme platformlarını kullanır. Visual Studio, C, C++, VB.NET, C#, F# gibi birçok dili desteklemektedir. Biz en kısa yol problemimizi çözüyorken C++ dilini kullanıyor olacağız.

## PROGRAMLAMA DİLİ: C++

1979 yılında Bjarne Stroustrup tarafından Bell Labs’[da](https://wmaraci.com/nedir/da) geliştirilen nesne yönelimli programlama dilidir. C++’ın ilk ismi “C with classes” dir ve C programlama dilinin bir eklentisi olarak kullanılmaktadır.

C++’ın yaratıcısı Bjarne Stroustrup bu programlama dilini öğrenciyken geliştirmiştir. C++ programlama dilinde C tarzında veya nesne yönelimli tarzda kesin senaryolarla kodlamalar yapılabilmektedir. Bu açıdan C++ en önemli ve işlevsel hibrit programlama dillerinden biri olma niteliğine de sahiptir.

# GELİŞME

## EN KISA YOL PROBLEMİ

En kısa yolu bulmak, iki düğüm arasında n maliyetle gidilebilen bir yolun varlığını belirleme problemidir. En kısa yol problemi herhangi bir düğümden bir başka düğüme, her bir düğümden tüm düğümlere ya da tüm düğümler için hesaplanabilir.

En kısa yol problemlerini çözebilmek için farklı algoritmalar kullanılır. Bu algoritmalar: DIJKSTRA ALGORİTMASI, BELLMAN VE FORD, FLOYD ALGORİTMASI

Şeklinde üçe ayrılır. Biz uygulamamızı yazarken Dijkstra Algoritmasını kullanıyor olacağız.

## DIJKSTRA ALGORİTMASI

Dikstra algoritması, bir düğümden diğer düğümlere en kısa yolları hesaplar. Bir başka ifadeyle belirli bir başlangıçtan noktasına göre en kısa yolu belirleyen algoritmadır. Ağırlıklı ve yönlü graflar için geliştirilmiş olup kenarların ağırlık değeri sıfır ya da sıfırdan büyük bir değer olmalıdır. Eğer kenarların değerleri sıfırdan küçük oluyorsa daha genel algoritmalardan Bellman-Ford kullanılabilir. Dijkstra algoritmasının zaman karmaşıklığı genel olarak 0(MlogN) şeklinde hesaplanmıştır.

Dijkstra algoritması en kısa yolu belirlerken Greedy yaklaşımını kullanır. Yani bir düğümden diğer bir düğüme geçerken olası en iyi yerel çözümü göz önüne alır. Her seferinde bir sonraki düğüme ilerleme Greedy yaklaşımına göre yapılır.

## BELLMAN VE FORD

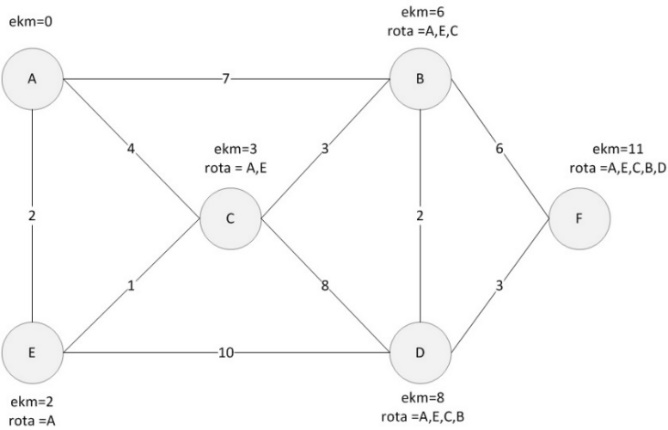
Dijkstra gibi bir düğümden diğer düğümlere olan en kısa yolları belirlemek için kullanılan bir algoritmadır. Bu algoritmanın Dijkstradan farkı; negatif değerli graflar için doğru çalışıyor olmasıdır.

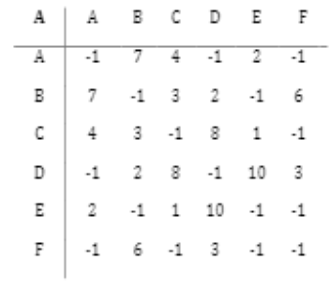
## FLOYD ALGORİTMASI

Bu algoritmada ise graf üzerindeki her bir düğüm için geriye kalan tüm düğümlere olan en kısa yol belirlenir. En kısa yolu bulmak için en sık kullanılan algoritmadır. Çoğunlukla yoğun graflarda kullanılmak için tercih edilir. Graf komşuluk matrisi olarak tutulursa bu algoritma 0() karmaşıklığındadır.

# SONUÇ

## ÇÖZECEĞİMİZ PROBLEM





Yukarıda maliyetli grafı verilen problemimizin komşuluk matrisi ile çözümü de belirtilmiştir. Dijkstra algoritması ile çözümü sağlanmıştır.

## KODLAR

#include "stdafx.h"

#include "cstring"

#define N 6

#define EBAS 0x7FFFFFFF

void Dijkstra();

int dizi[N][N] = { { -1, 7, 4, -1, 2, -1 },

{ 7, -1, 3, 2, -1, 6 }, { 4, 3, -1, 8, 1, -1 },

{ -1, 2, 8, -1, 10, 3 }, { 2, -1, 1, 10, -1, -1 },

{ -1, 6, -1, 3, -1, -1 } };

int maliyet[N];

char ROTA[N][N] = { NULL };

void \_tmain(int argc, \_TCHAR\* argv[])

{

int i;

Dijkstra();

puts("\nKomsuluk Matrisi\n");

for (i = 0; i < N; i++)

{

for (int j = 0; j < N; j++)

printf("%5d", dizi[i][j]);

printf("\n\n\n");

}

puts("\nMaliyetler\n");

for (i = 0; i < N; ++i)

printf("maliyet[%d]=%d\n", i, maliyet[i]);

puts("\nRotalar\n");

for (i = 0; i < N; ++i)

printf("ROTA[%d]=%s\n", i, ROTA[i]);

getchar();

}

void Dijkstra()

{

char \*ptr, ELEALINDI[N] = { 0 };

int i, j, ead, ek;

maliyet[0] = 0;

for (i = 1; i < N; ++i)

maliyet[i] = EBAS;

ead = 0;

for (i = 0; i < N; ++i)

{

for (j = 0; j < N; ++j)

if (!ELEALINDI[j])

if (dizi[ead][j] != -1)

if (maliyet[j]>dizi[ead][j] + maliyet[ead])

{

maliyet[j] = dizi[ead][j] + maliyet[ead];

strcpy(ROTA[j], ROTA[ead]);

ptr = ROTA[j];

while (\*ptr != NULL)

++ptr;

\*ptr = 'A' + ead;

}

ek = EBAS;

for (j = 1; j < N; ++j)

if (!ELEALINDI[j])

if (maliyet[j] < ek)

{

ek = maliyet[j];

ead = j;

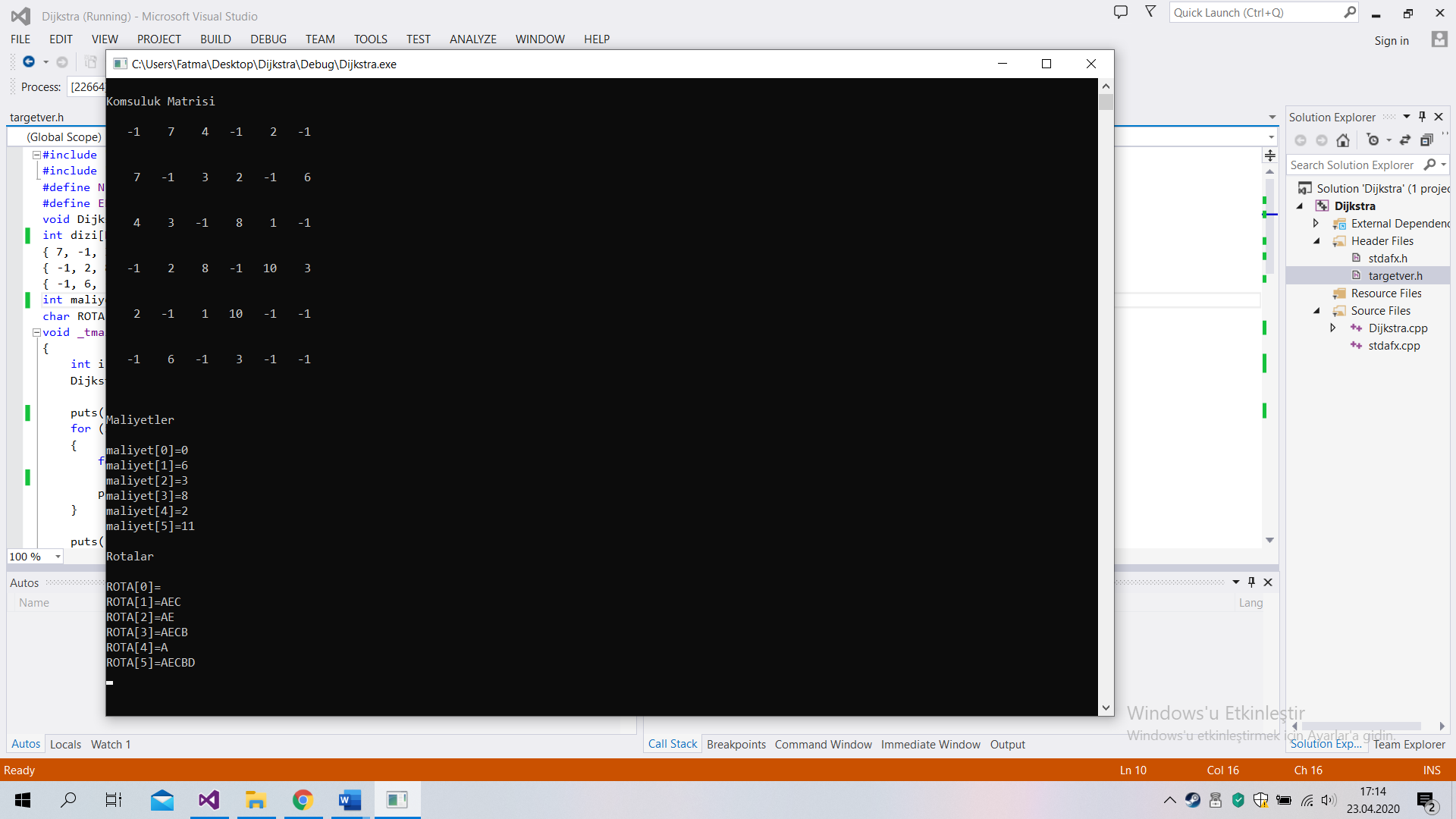
}

ELEALINDI[ead] = 1;

}

}

## EKRAN ÇIKTISI



## KAYNAK

<http://www.zafercomert.com/IcerikDetay.aspx?zcms=70>

<https://wmaraci.com/nedir/cplusplus>