

Google Maps ve Genetik Algoritmalarla GSP Çözümü İçin Öneri

Onur KARASOY¹ , Serkan BALLI²

¹ Muğla Sıtkı Koçman Üniversitesi Bilgi İşlem Dairesi Başkanlığı

² Muğla Sıtkı Koçman Üniversitesi Bilişim Sistemleri Mühendisliği Anabilim Dalı



akademik bilişim

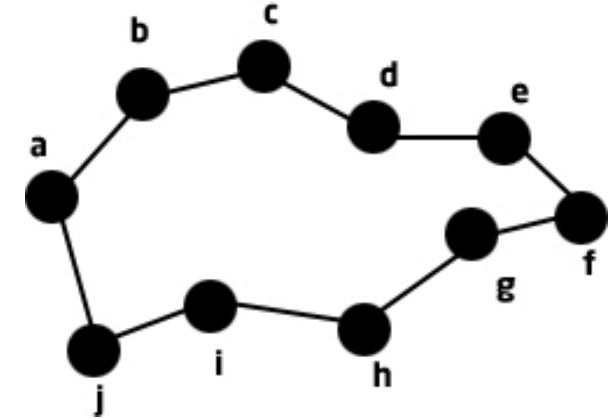
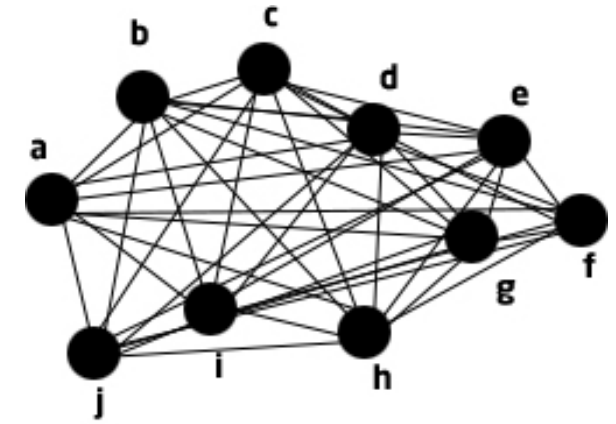
Giriş

- Gezgin Satıcı Problemi
- Genetik Algoritmalar
 - Başlangıç Popülasyonu
 - Uygunluk Değerinin hesaplanması
 - Yeniden Üretme
 - Çaprazlama
 - Mutasyon
- Google Maps (Google Haritalar)
- Uygulama



Gezgin Satıcı Problemi

- GSP'de amaç bir satıcının belli bir şehirden başlayıp her şehre sadece bir defa uğradıktan sonra başladığı şehre dönmek için en kısa mesafeyi hesaplamaktır.
- Hamilton Turu { $(n-1)!/2$ }
 - 1. Çizgenin tüm Hamilton turlarını bul.
 - 2. Her turun uzunluğunu hesapla.
 - 3. Turlar arasından en kisasını seç.
- 10 şehir içeren bir GSP için bulunması gereken tur sayısı 181.440
- 20 Şehir $\approx 6,08 \times 10^{16}$



Genetik Algoritma

- Genetik Algoritma yaklaşımının ortaya çıkışı 1970'lerin başında olmuştur. **1975'te John Holland'ın** makine öğrenmesi üzerine yaptığı çalışmalarda **canlılardaki evrimden** ve değişimden etkilenerek, bu genetik evrim sürecini bilgisayar ortamına aktarması ve böylece bir tek mekanik yapının öğrenme yeteneğini geliştirmek yerine, çok sayıdaki böyle yapıların tamamını “**çiftleşme, çoğalma, değişim...**” gibi genetik süreçler sonunda üstün yeni bireylerin elde edilebileceğini gösteren çalışmasından çıkan sonuçların yayınlanmasından sonra geliştirdiği yöntemin adı “**Genetik Algoritmalar**” olarak tanınmıştır.
- Bir probleme olası pek çok çözümün içerisinde **en uygununu** (en iyisini) bulmaya çalışan algoritmalarıdır.
- Popülasyon nesilden nesile geliştikçe **kötü çözümler yok olma, iyi çözümler** ise daha iyi çözümler oluşturmak için **kullanılma** eğilimindedirler.



Genetik Algoritma

- Çözüm uzayının **tamamını değil** yalnızca bir kısmını tararlar.
- Böylece etkin arama yaparak çok daha **kısa bir sürede** çözüme ulaşırlar
- Önemli bir üstünlüğü ise çözümlerden oluşan popülasyonu eş zamanlı inceleyerek **yerel en iyi çözümlere** takılmamaları.
- Genellikle çözüm alanı oldukça geniş
- Problemi etkileyen faktörlerin fazla olduğu durumlarda kullanılır.
- Genetik algoritmalar olasılık kurallarına göre çalışır ve ne kadar iyi çalışacağı önceden bilinemez.



Genetik Algoritma- Temel Kavramlar

GEN :

- “ *Kendi başına anlamı olan ve genetik bilgi taşıyan en küçük genetik birimdir*”. Kısmi bilgi taşıyan bu küçük yapıların bir araya gelmesiyle **tüm bilgileri içeren kromozomlar** meydana gelir.
- Bir gen A,B gibi bir karakter olabileceği gibi 0 veya 1 ile ifade edilen bir bit veya bit dizisi olabilir. Örneğin bir cismin x koordinatındaki yerini gösteren bir gen **101** şeklinde ifade edilebilir.

KROMOZOM :

- Bir yada birden fazla **genin** bir araya gelmesiyle oluşurlar. Probleme ait tüm bilgileri içerirler.
- Kromozomlar toplumdaki **bireyler** yada **üyelere** karşılık gelirler.
- Ele alınan problemde **alternatif çözüm adayıdır**.



Genetik Algoritma- Temel Kavramlar

KROMOZOM :

- Örneğin kromozom bir problemde açı, boyut ve koordinat değişkenlerinden veya bir dikdörtgen prizmasının ölçülerinden (yükseklik, genişlik, derinlik) oluşabilir.

001 101 111 -> **1,5,7** değerleri kromozomu oluşturan genlerdir.

- Genetik algoritma işlemlerinde kromozomları kullandığı için **kromozom tanımları** çok **iyi** ifade edilmelidir.
- En kısa yol hesaplama örneğinde şehirleri temsil eden sayılar oluşan olası çözüm kümelerinin herbiri.

3 5 1 2 4



Genetik Algoritma- Temel Kavramlar

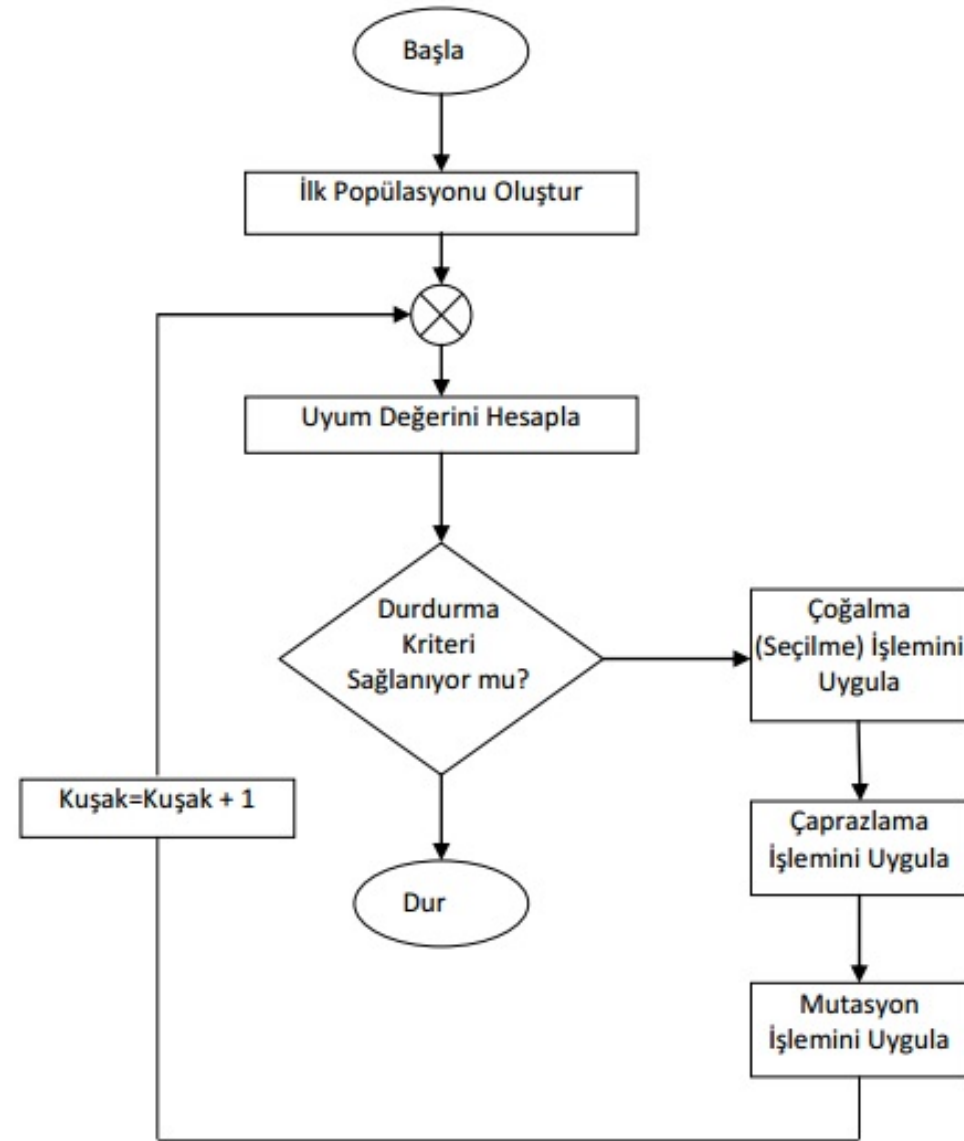
POPÜLASYON (Topluluk):

- Kromozomlar veya bireyler topluluğudur. *Popülasyon üzerinde durulan problem için alternatif çözümler kümesidir.*
- Aynı anda bir popülasyonda ki birey sayısı sabit ve probleme göre kullanıcı tarafından belirlenir. *(Zayıf olan bireylerin yerini kuvvetli yeniler almaktadır)*
- Genetik operatörlerle sağlanan yenilemeler sayesinde probleme daha uygun bireyler bulunabilmektedir.

3 5 1 2 4 , 3 4 5 1 2 , 3 2 4 5 1



Akış Diyagramı



Şekil 1. Genetik algoritmanın akış diyagramı.



Kodlama

- Kromozomlarla temsil edilen çözümlerin nasıl oluşturulması gerektiğini gösterir.
- a) İkili Kodlama : Her bir kromozom 0 ve 1'lerden oluşan bit dizisidir ve ikili dizi ile ifade edilir.

Kromozom A \rightarrow 10011111
Kromozom B \rightarrow 11011110

- b) Permütasyon Kodlama: Bu kodlama gezgin satıcı problemi ve iş sıralama problemleri gibi sıralama problemlerinde kullanılır.

Kromozom A \rightarrow 3 5 1 2 4 7 6 0
Kromozom B \rightarrow 5 3 4 2 1 0 6 7



Kodlama

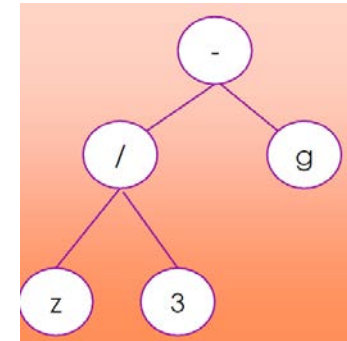
c) Değer Kodlama : Bu kodlama gerçel gibi kompleks sayıların yer aldığı problemlerde kullanılır.

Kromozom A \rightarrow 3.2 6.5 0.8 9.4

Kromozom B \rightarrow ABCDE

Kromozom C \rightarrow (geri), (sağ), (ileri), (sol)

d) Ağaç Kodlama : Değişen gelişen programlar veya değerler için kullanılır. (Bir kelime bir işlem)

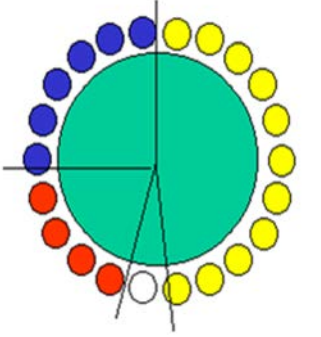


Seçilim

- Yeni topluluğu oluşturmak için mevcut topluluktan **çaprazlama** ve **mutasyon** işlemine tabi tutulacak bireylerin seçilmesi gerekir.
- Teoriye göre iyi olan bireyler yaşamını sürdürmeli ve bu bireylerden yeni bireyler oluşturulmalıdır.
- Bu nedenle tüm seçim yöntemlerinde **uygunluk değeri fazla** olan bireylerin seçilme olasılığı daha yüksektir.
- En bilinen seçim yöntemleri **Rulet Seçilimi**, **Turnuva Seçilimi** ve **Sıralı Seçilimdir**.



Seilim



- **Rulet Seilimi:**

Topluluktaki tm bireylerin uygunluk deęerleri toplanır ve her bireyin seilme olasılıęı, uygunluk deęerinin bu toplam deęere oranı kadardır.

- **Sıralı Seilim:**

En kt uygunlukta olan kromozoma 1 deęeri verilir, ondan daha iyi olana 2, daha iyisine 3 deęeri verilerek devam edilir.

- **Turnuva Seilimi:**

Topluluk ierisinden rastgele k adet (3,5,7..) birey alınır. Bu bireylerin ierisinden uygunluk deęeri en iyi olan birey seilir.



Çaprazlama

- Amaç, **ata** kromozomun yerlerini değiştirerek **çocuk kromozomlar üretmek** ve böylelikle zaten uygunluk değeri yüksek olan ata kromozomlardan daha yüksek uygunluklu çocuk kromozomlar üretmektir.

A) Tek noktalı çaprazlama :

- Kromozom-1 : 11000|00100110110
- Kromozom-2 : 11011|11000011110
- Çocuk-1 : 1101111000011110
- Çocuk-2 : 1100000100110110



Çaprazlama

B) Çift noktalı çaprazlama

- Kromozom-1 : 11000|00100|110110
- Kromozom-2 : 11011|11000|011110
- Çocuk-1 : 1100011000110110
- Çocuk-2 : 1101100100011110



Çaprazlama

B) Sıralı Çaprazlama

- A = 9 8 4 | 5 6 7 | 1 3 2 10

- B = 8 7 1 | 2 3 10 | 9 5 4 6

- A = 9 8 4 | 2 3 10 | 1 H H H

- B = 8 H 1 | 5 6 7 | 9 H 4 H

- AI = 9 8 4 | 2 3 10 | 1 5 6 7

- BI = 8 2 1 | 5 6 7 | 9 3 4 10

Sıra ile soldan sağa doğru



Mutasyon

- Kromozomların kendi **genleri** veya genleri oluşturan küçük birimleri üzerinde **değişiklik** yapılmasını sağlayan işlemcidir.
- GA'da değişimin sağladığı avantaj, problemin çözüm alanını araştırmada **yön değişikliklerini** sağlayarak

'Mutasyon(Değişim) yardımıyla araştırmanın kısır döngüye girmesini önlemektir. ' (Lokal Maksimum)

- **Pozisyona göre değişim** : Rasgele seçilen karakterlerin(genlerin) yerleri değiştirilerek gerçekleştirilir.
- **Sıraya göre değişim** : Kromozomun rasgele seçilen iki karakterinden ikincisinin, birincinin önüne getirilmesiyle olur



Mutasyon

	Önce	Sonra
Pozisyona göre değişim	<u>A</u> B C D E <u>F</u>	<u>F</u> B C D E <u>A</u>
Sıraya göre değişim	<u>A</u> B C D E <u>F</u>	<u>F</u> <u>A</u> B C D E
Kromozom (Rast)	1 1 0 1 0 1 <u>1</u> 0	1 1 0 1 0 1 <u>0</u> 0



GA'da Performansı Etkileyen Faktörler

- **Popülasyon büyüklüğü / Kromozom sayısı:** Kromozom sayısını arttırmak çalışma zamanını arttırırken, azaltmak da kromozom çeşitliliğini yok eder.
- **Mutasyon Oranı:** Kromozomlar birbirine benzemeye başladığında hala çözüm noktalarının uzağında bulunuyorsa mutasyon işlemi GA'nın sıkıştığı yerden (tüm kromozomlar aynı platoda) kurtulmak için tek yoludur. Ancak yüksek bir değer vermek GA'nın kararlı bir noktaya ulaşmasını engelleyecektir.
- **Kaç Noktalı Çaprazlama Yapılacağı:** Normal olarak çaprazlama tek noktada gerçekleştirilmekle beraber yapılan araştırmalar bazı problemlerde çok noktalı çaprazlamanın çok yararlı olduğunu göstermiştir.
- **Çaprazlamanın sonucu elde edilen bireylerin nasıl değerlendirileceği:** Elde edilen iki bireyin birden kullanılıp kullanılmayacağı bazen önemli olmaktadır.



GA'da Performansı Etkileyen Faktörler

- **Durum kodlanmasının nasıl yapıldığı:** Bir parametrenin doğrusal yada logaritmik kodlanması GA'nın performansında önemli bir farka yol açabilir.
- **Başarı değerlendirmesinin nasıl yapıldığı:** Akıllıca yazılmamış bir değerlendirme işlevi, çalışma zamanını uzatabileceği gibi çözüme hiçbir zaman ulaşılamamasına da neden olabilir.



Google Maps

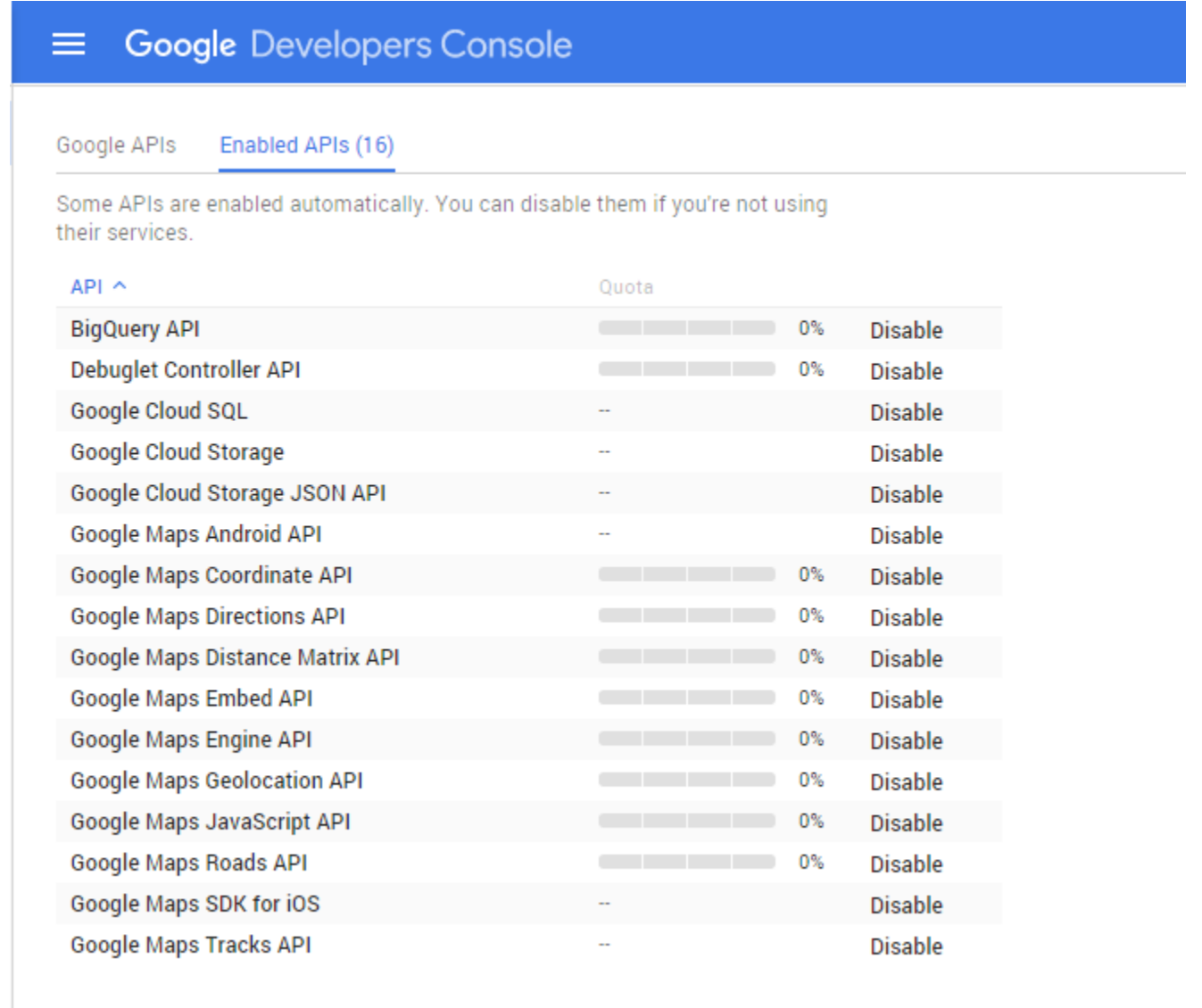
- 08 Şubat 2005
- Web tabanlı harita sistemi
 - İş yerleri, kütüphaneler, okullar
 - Kültürel etkinlikler
 - Emlak uygulamaları
 - Yol bulma uygulamaları
 - Hava durumu
 - Kent suç haritaları gösterimi
 - ..

gibi servislerde kullanılır.



Google Developer Console

- Bu konsoldan yeni bir proje oluşturularak kullanılacak apilerin enable edilmesi gerekiyor.
- Google Map 3.0 ile Directions (Yol Tarifi)
- Google Maps Distance Matrix API (Noktalar Arası Mesafe Matrisi)
- Google Maps Geolocation API (Şuanki Konumu Alma)
- Google Maps JavaScript API (Google Haritaları kullanmak için gerekli olan javascript kütüphanesi)



Google Developers Console

Google APIs [Enabled APIs \(16\)](#)

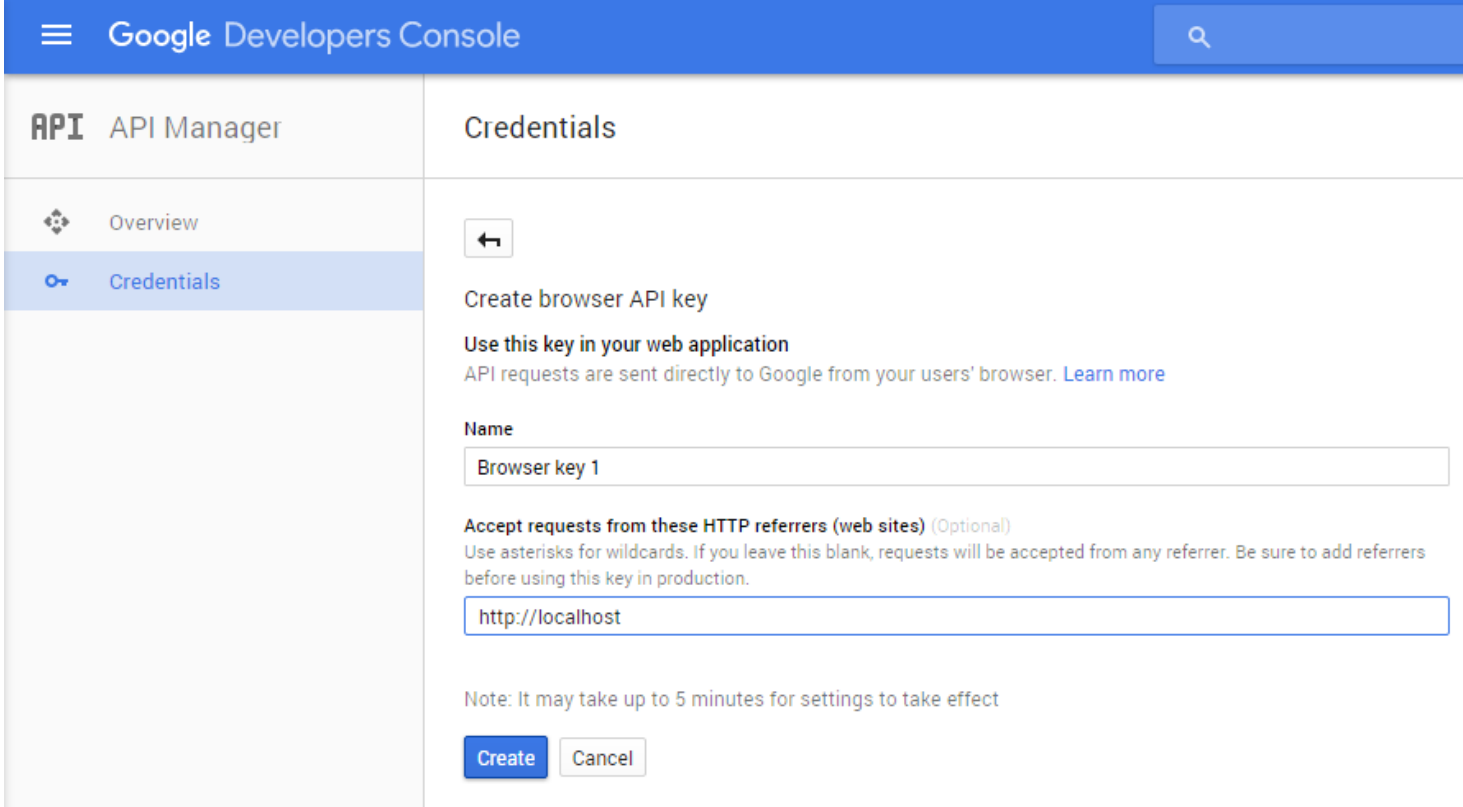
Some APIs are enabled automatically. You can disable them if you're not using their services.

API ^	Quota	
BigQuery API	<div style="width: 0%;"></div> 0%	Disable
Debuglet Controller API	<div style="width: 0%;"></div> 0%	Disable
Google Cloud SQL	--	Disable
Google Cloud Storage	--	Disable
Google Cloud Storage JSON API	--	Disable
Google Maps Android API	--	Disable
Google Maps Coordinate API	<div style="width: 0%;"></div> 0%	Disable
Google Maps Directions API	<div style="width: 0%;"></div> 0%	Disable
Google Maps Distance Matrix API	<div style="width: 0%;"></div> 0%	Disable
Google Maps Embed API	<div style="width: 0%;"></div> 0%	Disable
Google Maps Engine API	<div style="width: 0%;"></div> 0%	Disable
Google Maps Geolocation API	<div style="width: 0%;"></div> 0%	Disable
Google Maps JavaScript API	<div style="width: 0%;"></div> 0%	Disable
Google Maps Roads API	<div style="width: 0%;"></div> 0%	Disable
Google Maps SDK for iOS	--	Disable
Google Maps Tracks API	--	Disable



Api Kullanımı için Key alma

- Developer console 'da Credential sekmesinden yeni bir **Browser Key** oluşturmamız gerekiyor.



The screenshot shows the Google Developers Console interface. The top navigation bar is blue with the text 'Google Developers Console' and a search icon. Below the navigation bar, there is a sidebar on the left with 'API Manager' and 'Credentials' sections. The 'Credentials' section is active, showing a list of credentials. The main content area displays the 'Create browser API key' form. The form includes a 'Name' field with the value 'Browser key 1', an 'Accept requests from these HTTP referrers (web sites) (Optional)' field with the value 'http://localhost', and a 'Note' stating 'It may take up to 5 minutes for settings to take effect'. There are 'Create' and 'Cancel' buttons at the bottom of the form.

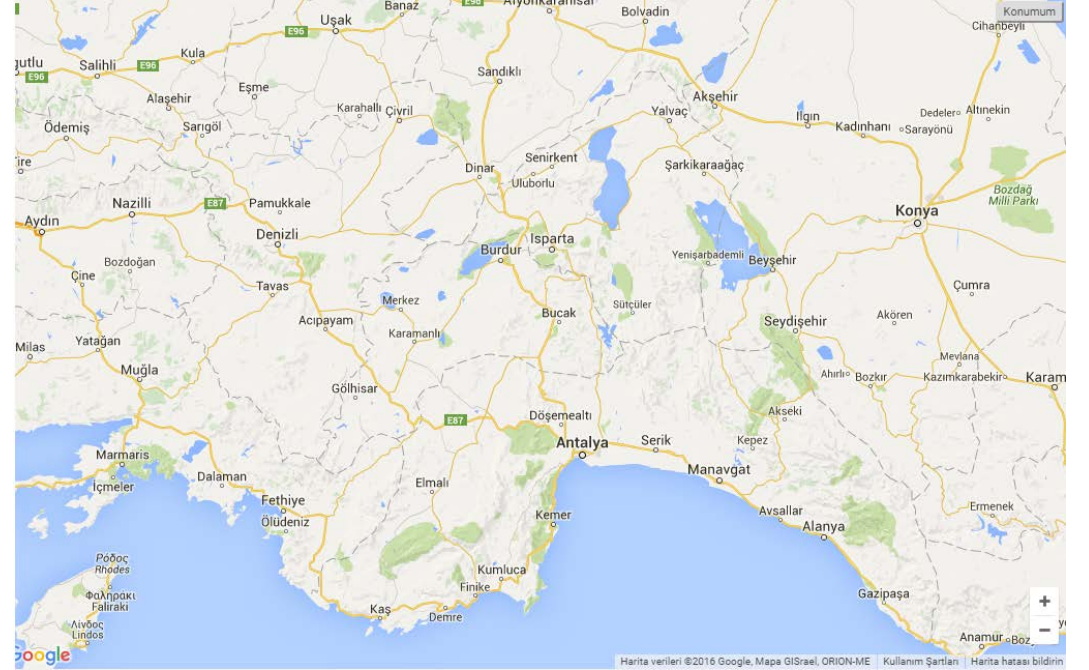
- Apinin hangi adreste kullanılacağını belirtmek gerekiyor.
- Birden fazla adres belirtilebiliyor.



Harita Oluřturma

Başlangıç ayarları

```
var opts = {  
  center: new google.maps.LatLng(37.316667, 28.7836667),  
  zoom: 8,  
  streetViewControl: false,  
  mapTypeControl: false  
};
```



Harita olarak tanımlanan html element

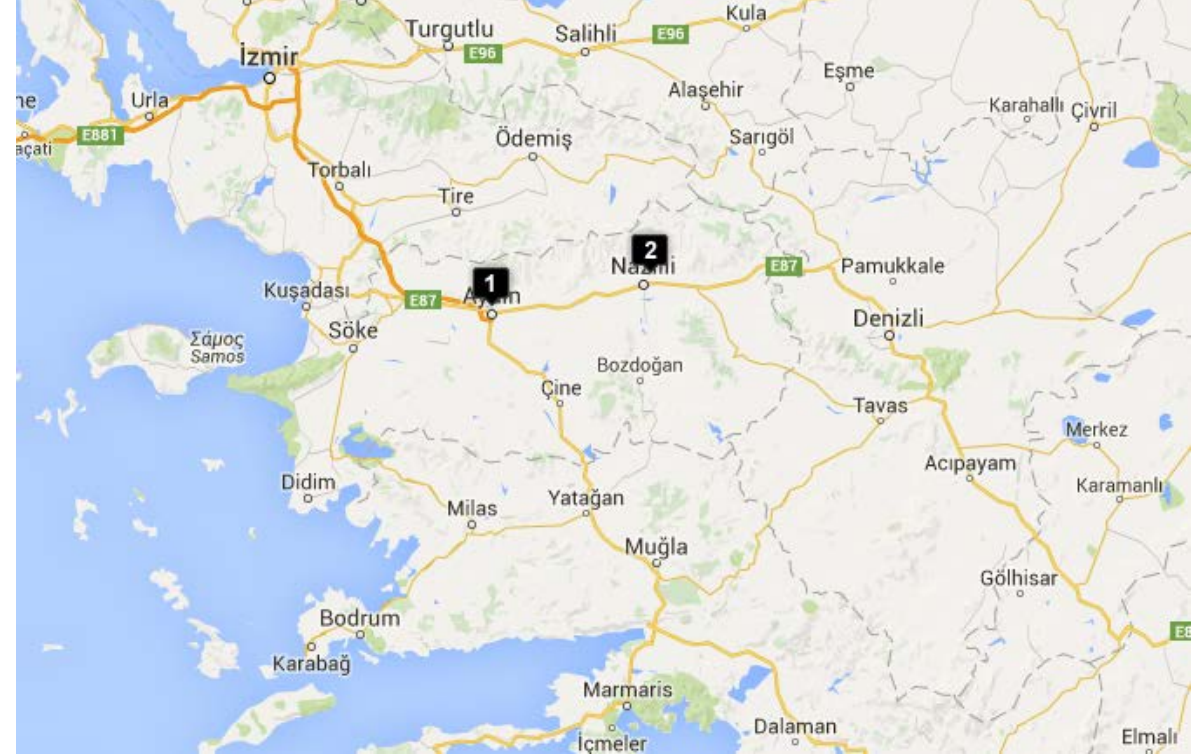
```
map = new google.maps.Map(document.getElementById('map-canvas'), opts);
```



Konumların Belirlenmesi

Tıklanan Konumun Diziye atılması

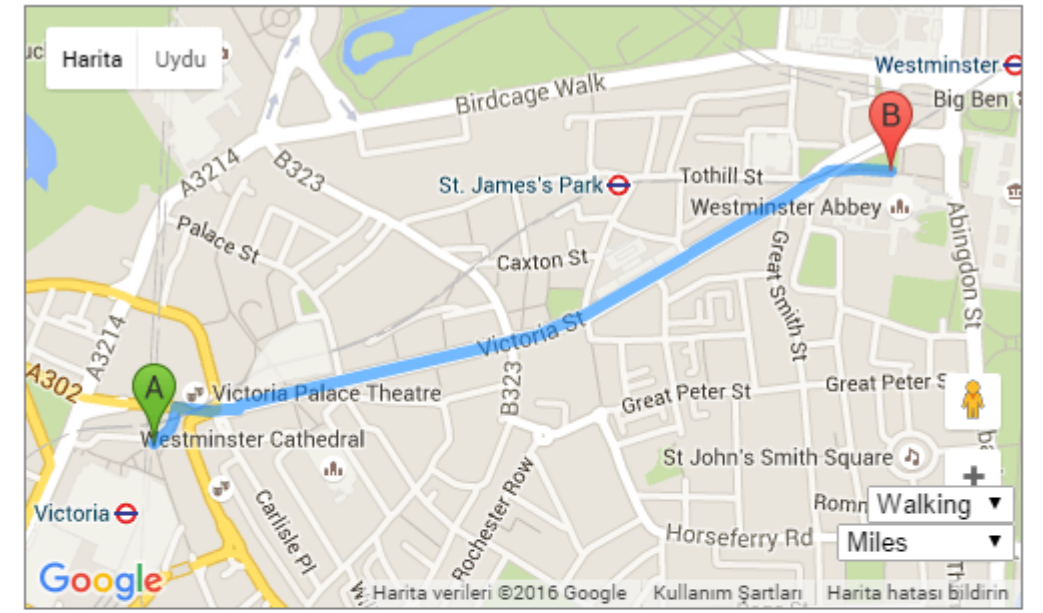
```
google.maps.event.addListener(map, 'click', function(event) {  
  
    nodes.push(event.latLng);  
  
});
```



Uzaklıkların Hesaplanması

DistanceMatrixService

```
function getDistances(callback) {  
  var service = new google.maps.DistanceMatrixService();  
  service.getDistanceMatrix({  
    origins: nodes,  
    destinations: nodes,  
    travelMode: google.maps.TravelMode[('#travel-type').val()],  
    avoidHighways: parseInt($('#avoid-highways').val()) > 0 ? true : false, //  
    avoidTolls: false  
  }, function(distanceData) { ..}
```

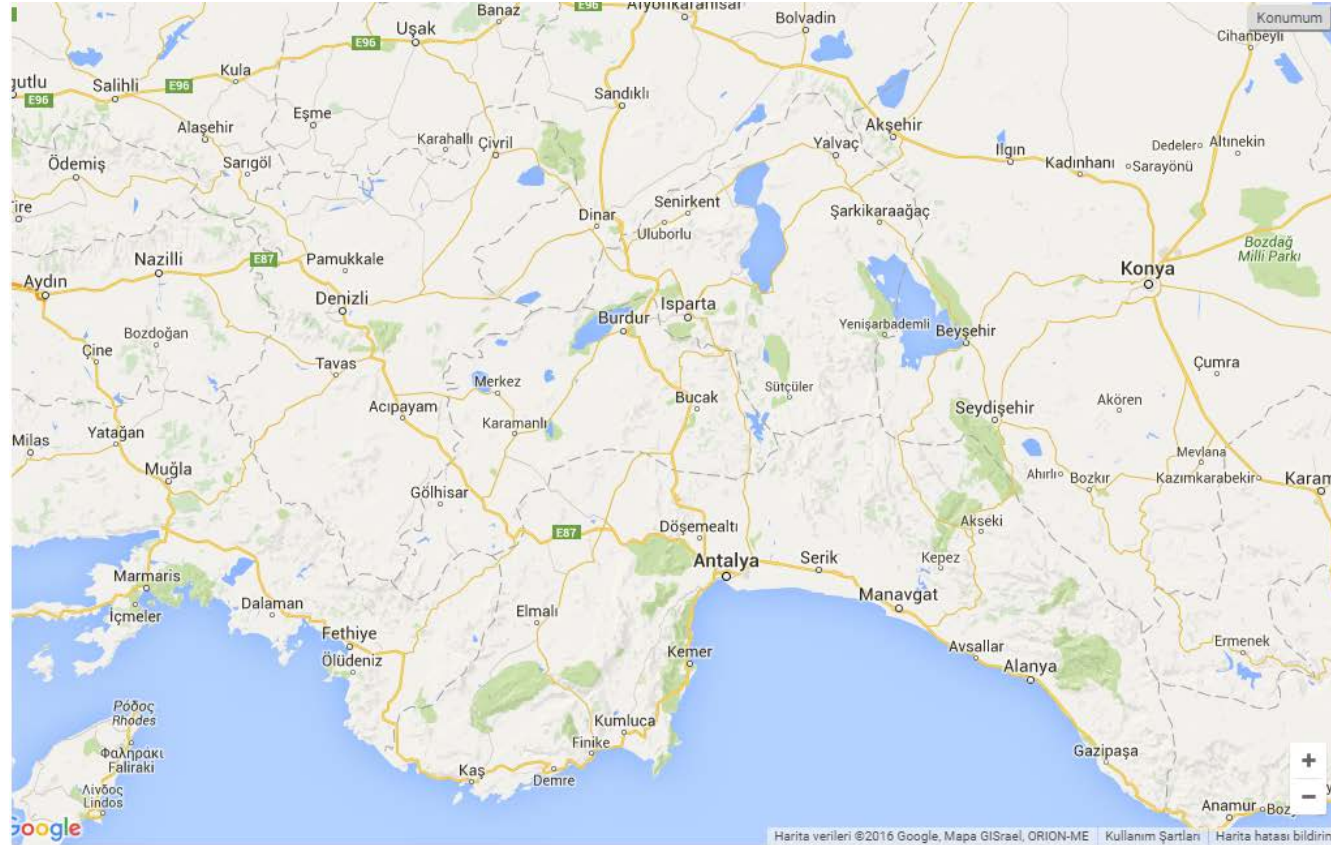


	Buckingham Palace	Houses of Parliament	Tower Bridge	Trafalgar Square	Westminster Abbey
Euston	0,3 mil 6 dakika	2,3 mil 45 dakika	3,2 mil 1 saat 6 dakika	1,7 mil 33 dakika	2,3 mil 46 dakika
Kings Cross	0,4 mil 9 dakika	2,6 mil 53 dakika	3,1 mil 1 saat 2 dakika	2,0 mil 41 dakika	2,6 mil 54 dakika
Liverpool St	2,8 mil 57 dakika	2,7 mil 54 dakika	0,8 mil 17 dakika	2,3 mil 47 dakika	2,7 mil 55 dakika
Paddington	2,5 mil 51 dakika	2,8 mil 56 dakika	4,8 mil 1 saat 38 dakika	2,6 mil 51 dakika	2,8 mil 55 dakika
St. Pancras	0,3 mil 6 dakika	2,6 mil 53 dakika	3,1 mil 1 saat 3 dakika	2,0 mil 41 dakika	2,6 mil 54 dakika
Victoria	3,1 mil 1 saat 3 dakika	0,8 mil 16 dakika	3,3 mil 1 saat 7 dakika	1,2 mil 26 dakika	0,7 mil 15 dakika
Waterloo	2,2 mil 46 dakika	0,8 mil 16 dakika	1,7 mil 34 dakika	0,8 mil 17 dakika	0,8 mil 17 dakika



Uygulama (Açılış Ekranı)

- kodsey.com/map/ga.html



Ayarlar

Başla Temizle

Genetik Algoritma Ayarları

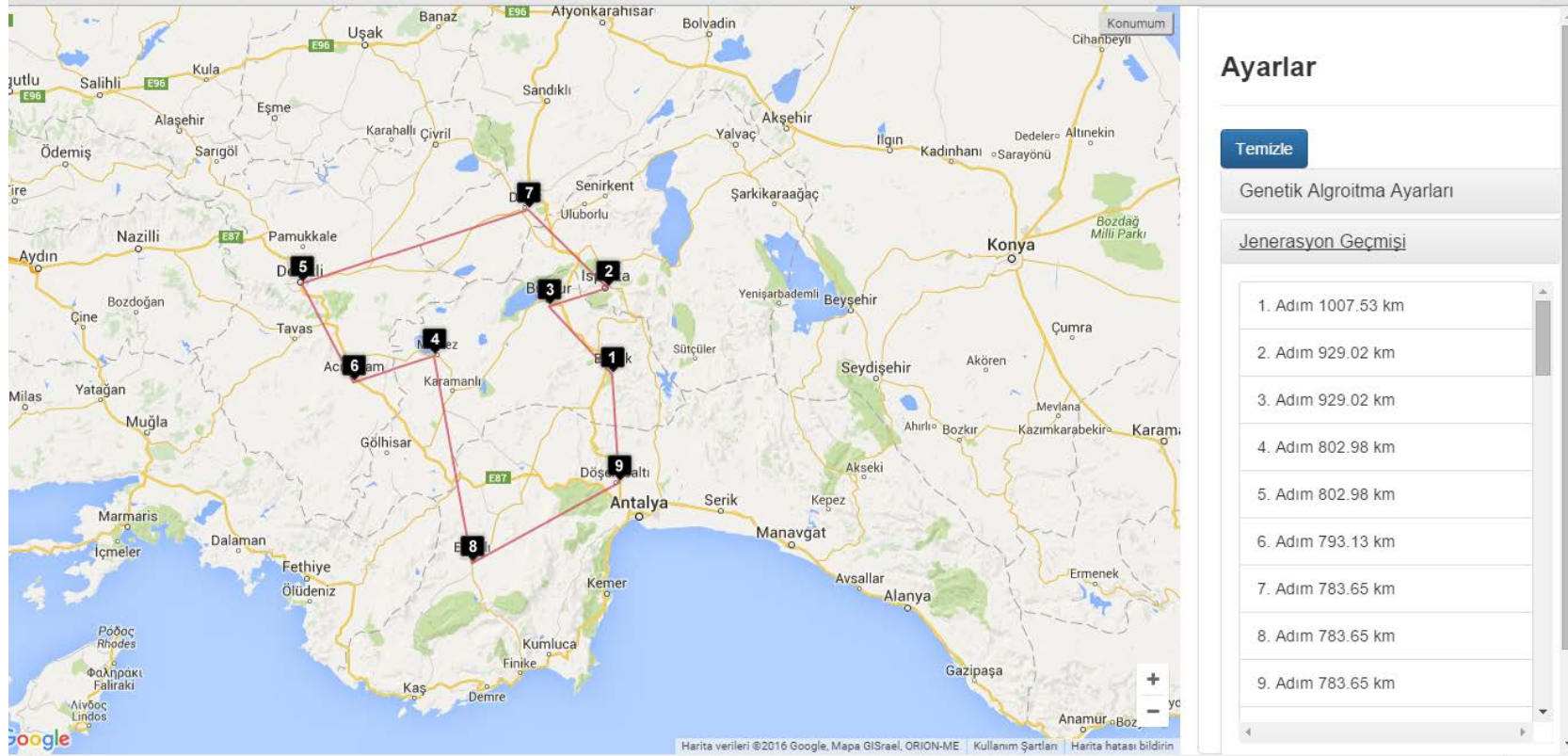
Seyahat Şekli:	Araba ▼
Otoyolları Kullanma	Hayır ▼
Popülasyon:	50 ▼
Mutasyon Oranı	0.1 ▼
Çaprazlama Oranı:	0.5 ▼
Elitist seçim	Açık ▼
Maksimum Jenerasyon:	50 ▼
Debug Info	
Hedef Sayısı:	0

Jenerasyon Geçmişi

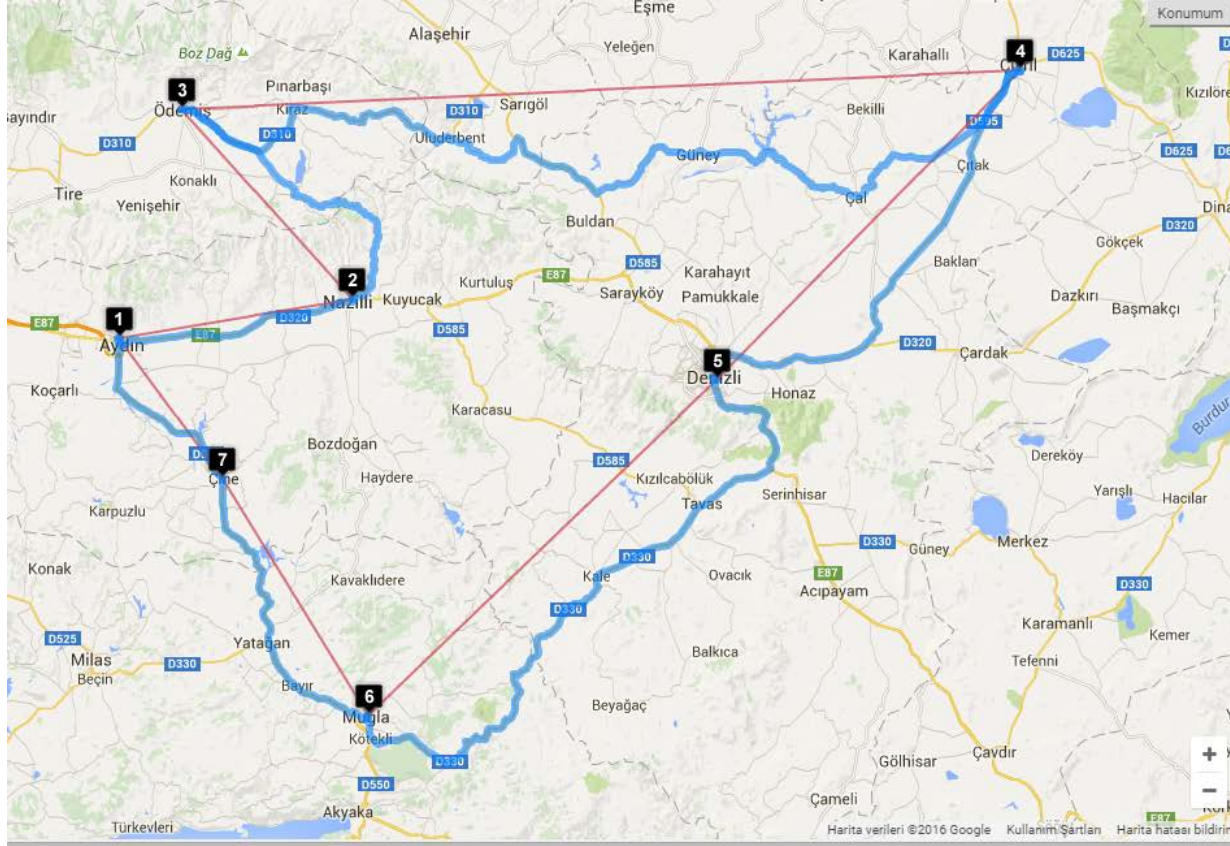
Adres Ekle



Uygulama (Ekran 2)



Uygulama (Ekran 3)



Ayarlar

Temizle

Genetik Algoritma Ayarları

Jenerasyon Geçmişi

1. Adım 751.45 km
2. Adım 751.45 km
3. Adım 740.03 km
4. Adım 740.03 km
5. Adım 740.03 km
6. Adım 740.03 km
7. Adım 740.03 km
8. Adım 740.03 km
9. Adım 740.03 km





akademik bilişim

Teşekkürler