

126912

**TEMEL JEODEZİK HESAPLARIN  
İNTERNET ORTAMINDA  
İNTERAKTİF UYGULAMASI**

**YÜKSEK LİSANS TEZİ**  
**Müh. Ali İhsan KURT**  
**(501991099)**

**T.C. YÜKSEKÖĞRETİM KURULU**  
**DEKÜBANTISYON BİRİMİ**

**Tezin Enstitüye Verildiği Tarih : 2 Ocak 2002**  
**Tezin Savunulduğu Tarih : 14 Ocak 2002**

**Tez Danışmanı :** Doç.Dr. Rahmi Nurhan ÇELİK  
**Diğer Jüri Üyeleri** Prof.Dr. Rasim DENİZ (İ.T.Ü.)  
Prof.Dr. Hüseyin DEMİREL (Y.T.Ü.)

*Rahmi Nurhan Çelik*  
*Rasim Deniz*  
*Hüseyin Demirel*

126912

**OCAK 2002**

## ÖNSÖZ

Temel Jeodezik Hesapların İnternet Ortamında İnteraktif Uygulaması konulu bu çalışmada, tez danışmanlığımı üstlenen ve tezin her aşamasında bilgi ve deneyimini esirgemeyen sayın hocam Doç.Dr. Rahmi Nurhan ÇELİK'e çok teşekkür ederim.

Bu çalışmada, haritacılıkta kullanılan temel jeodezik programların internet ortamında kullanılmasına ilişkin bir uygulama yapılmıştır. Benim için yeni bir konu olan internet programcılığı konusunda bana destek olan sayın Mustafa OCAK'a teşekkürü bir borç bilirim.

Ayrıca, çalışmalarım sırasında bana sürekli yardımcı olan ve her türlü desteği sağlayan sevgili kardeşim Erhan'a çok teşekkür ederim.

Ocak 2002

Ali İhsan KURT

## İÇİNDEKİLER

<b>KISALTMALAR</b>	<b>iv</b>
<b>ŞEKİL LİSTESİ</b>	<b>v</b>
<b>SEMBOL LİSTESİ</b>	<b>vi</b>
<b>ÖZET</b>	<b>vii</b>
<b>SUMMARY</b>	<b>viii</b>
<b>1. GİRİŞ</b>	<b>1</b>
1.1. Giriş ve Çalışmanın Amacı	1
<b>2. İNTERAKTİF WEB UYGULAMALARI</b>	<b>3</b>
2.1. Genel Uygulamalar	4
2.2. Jeodezik Amaçlı Web Uygulamaları	5
2.3. İnteraktif Web Tasarımı için Programlama Dili Seçimi	8
2.3.1. Java programlama dili	8
2.3.2. Neden Java	8
2.3.3. Java'nın özellikleri	9
2.3.4. Java'nın çalışma biçimi	11
2.3.5. Java'nın diğer programlama dilleriyle karşılaştırılması	15
<b>3. TEMEL JEODEZİK HESAPLAMALAR İÇİN WEB TASARIMI</b>	<b>19</b>
3.1. Hedef Kitle ve Gereksinimlerin Tespiti	19
3.2. Temel Jeodezik Hesapların Web'de Ele Alınma Biçimi	20
3.3. Temel Jeodezik Hesapların Web'deki Yapısı	24
3.4. Web Sayfasının Kullanım Klavuzu	29
<b>4. TEMEL JEODEZİK HESAPLAR</b>	<b>32</b>
4.1. I. Temel Ödev Çözümü	32
4.2. II. Temel Ödev Çözümü	34
4.3. Kartezyen-Coğrafik Koordinatlar Dönüşümü ve Tersisi	37
4.4. Coğrafik-UTM Koordinatlar Dönüşümü ve Tersisi	39
4.5. Dilimden Dilime Dönüşümler	44
4.5.1. Üç derecelik sistemde dilimden dilime dönüşüm	44
4.5.2. Altı derecelik sistemde dilimden dilime dönüşüm	45
4.6. Datum Dönüşümü Hesabı	46
4.7. Julian günü, GPS Haftası Hesabı	48
<b>5. SONUÇLAR VE ÖNERİLER</b>	<b>50</b>
<b>KAYNAKLAR</b>	<b>52</b>
<b>ÖZGEÇMİŞ</b>	<b>54</b>

## KISALTMALAR

<b>OS</b>	: Operating System
<b>JAR</b>	: Java Archive
<b>HTML</b>	: Hyper Text Markup Language
<b>OOP</b>	: Object Oriented Programming
<b>CGI</b>	: Common Gateway Interface
<b>ASP</b>	: Active Server Pages
<b>KBPS</b>	: Kilo Bit Per second
<b>MBPS</b>	: Mega Bit Per second
<b>AWT</b>	: Abstract Window Toolkit
<b>CPU</b>	: Central Processing Unit
<b>ISDN</b>	: Integrated Services Digital Network
<b>HTTP</b>	: Hyper Text Transfer Protocol
<b>JDK</b>	: Java Development Kit
<b>UTM</b>	: Universal Transverse Mercator
<b>GPS</b>	: Global Positioning System
<b>UT</b>	: Universal Time
<b>DOM</b>	: Dilim Orta Meridyeni
<b>ED-50</b>	: European Datum – 1950
<b>WGS-84</b>	: World Geodetic System – 1984

## ŞEKİL LİSTESİ

	<u>Sayfa No</u>
Şekil 2.1	: İnteraktif GPS veri değerlendirmesi uygulamasına örnek sayfa .. 5
Şekil 2.2	: İnteraktif astronomik hesapların internet sayfası ..... 6
Şekil 2.3	: I. Temel ödev çözümü yapan interaktif web sayfası örneği ..... 7
Şekil 2.4	: Java işletim sistemi elemanları ve aralarındaki ilişkiler ..... 13
Şekil 2.5	: Java kaynak kodunun internet üzerinden transferi ..... 14
Şekil 2.6	: Java uygulamasının kullanıcı bilgisayarındaki görüntüsü ..... 14
Şekil 3.1	: Java sınıf hiyerarşisi ..... 21
Şekil 3.2	: Applet'in çalışma evreleri ve metodları ..... 23
Şekil 3.3	: Kartezyen-Coğrafi koordinat dönüşümü programının arayüzü ... 26
Şekil 3.4	: Index.html dosyasının görünümü ..... 28
Şekil 4.1	: İnci Temel ödev çözümü yapan uygulama ..... 32
Şekil 4.2	: İnci Temel ödev çözümü yapan uygulama ..... 35
Şekil 4.3	: Kartezyen-Coğrafi koordinatlar dönüşümü yapan program .....37
Şekil 4.4	: Coğrafik – UTM Koordinatlar Dönüşümü yapan program ..... 40
Şekil 4.5	: 3 Derecelik sistemde dilimden dilime dönüşüm yapan program.. 44
Şekil 4.6	: 6 Derecelik sistemde dilimden dilime dönüşüm yapan program.. 45
Şekil 4.7	: Datum dönüşüm programı ..... 46
Şekil 4.8	: Julian Günü ve GPS Haftası hesabını yapan applet .....48

## SEMBOL LİSTESİ

$\phi$	: Enlem
$\lambda$	: Boylam
$h$	: Elipsoit yüksekliği
$\alpha_{12}$	: 1. Noktadan 2. noktaya azimut
$\alpha_{21}$	: 2. Noktadan 1. noktaya azimut
$a$	: Elipsoidin büyük yarı eksen
$b$	: Elipsoidin küçük yarı eksen
$s$	: Noktalar arasındaki elipsoidal mesafe
$f$	: Elipsoidin basıklığı
$e$	: Elipsoidin birinci dış merkezliliği
$X, Y, Z$	: Noktanın kartezyen koordinat bileşenleri
$E$	: Sağa değer
$N$	: Yukarı değer
$\gamma$	: Meridyen yakınsama açısı
$K$	: Haritalama ölçeği
$K_0$	: Küçültme faktörü
$\varepsilon_x$	: X eksenine etrafındaki dönüklük
$\varepsilon_y$	: Y eksenine etrafındaki dönüklük
$\varepsilon_z$	: Z eksenine etrafındaki dönüklük
$t_x$	: X eksenine yönündeki öteleme
$t_y$	: Y eksenine yönündeki öteleme
$t_z$	: Z eksenine yönündeki öteleme
$k$	: Ölçek

## ÖZET

Bu çalışmada, haritacılıkta kullanılan temel jeodezik programların internet ortamında kullanılmasına ilişkin bir uygulama yapılmıştır. Öncelikle hangi uygulamaların programlarının yapılacağı tespit edilmiştir. Daha sonra bu uygulamayı internet ortamında en iyi temsil edebilecek bir programlama dili araştırılmıştır.

Nesnel programlama yapısı, görsel programcılığı desteklemesi, platformdan bağımsız çalışabilme ve internet programcılığına uygunluğu sebebiyle Java programlama dilinin Applet teknolojisi bu proje için ideal görülmüştür.

Java Applet'lerinin çalıştığı ortam, kullanıcının internet tarayıcısıdır. Programların bulunduğu siteye kurulan bağlantı ile Applet'in kaynak kodu kullanıcının bilgisayarına aktarılır. Yükleme tamamlandıktan sonra uygulama çalışmaya hazırdır. Tarayıcının içinde bulunan Java sanal makinesi bu kodu yorumlar ve sonucu ekrana taşır.

Applet'in çalışma hızı kullanıcının bilgisayarının hızı ile aynıdır. İnternet sunucusunda çalışan programlar ise ilgili uygulamaya olan talep ile bağlantılı hızda çalışırlar. Örneğin, ilgili uygulamaya talep o anda ne kadar fazlaysa ve internet bağlantı hızı ne kadar yavaşsa, bekleme süresi o kadar fazla olacaktır. Halbuki bir Java Applet'i, aktarım tamamlandıktan sonra, internet bağlantısı kesilse bile çalışmaya devam edecektir.

Temel uygulama programları olarak; julian günü hesabı, kartezyen-coğrafi koordinatlar dönüşümü, coğrafi-düzlem koordinatlar dönüşümü, 3 ve 6 derecelik dilim genişliği sistemlerinde dilimden dilime dönüş hesabı, İnci ve İnci temel ödev çözümleri ve datum dönüşüm hesabı programları hazırlanmıştır. Programların kullanımı tamamen grafik arayüzlü olup kullanıcıya kolaylık sağlamaktadır.

Bu çalışmanın sadece bu programlar ile sınırlı kalmayıp, kullanıcıların önerileri ve katkılarıyla geliştirilip bir program kütüphanesine dönüşmesi hedeflenmektedir.

## **SUMMARY**

In this study, an application about using the basic geodetic programs via internet is developed. First of all, geodetic applications that would be programmed were determined. After that, a research was performed about which programming language would be suitable for this project.

The Applet technology of Java programming language was found ideal since its OOP (Object Oriented Programming) and GUI (Graphical User Interface) based structure, and platform independent feature.

The Applets work just within the user's internet browser. After the connection to the server where the Applets reside, the bytecode is downloaded to the user's computer. After the download process, the Applet is ready to run. The JVM (Java Virtual Machine) that is embedded in the browser interprets this code and carries out the results to the screen.

The speed of Applet is directly proportional to the speed of the user's computer. The programs which use the server to run are slow as the amount of the request to the server. However a Java Applet will continue to run even you disconnect from the internet.

The applications that were programmed are; finding the julian day, cartesian-geographic coordinates transformation, geographic plane coordinates transformation, zone to zone transformations, direct and inverse problem, datum transformation. All programs are GUI based and user friendly.

It is aimed that this study develop with the proposals and the contributions of the users and become a geodetic program library.

# 1. GİRİŞ

## 1.1 Giriş ve Çalışmanın Amacı

Teknolojinin ilerlemesine paralel olarak bilgiye erişim hızının önemi gün geçtikçe artmaktadır. İnternet teknolojisinin gelişmesiyle birlikte kaynak ve uygulama araştırma çalışmaları evden veya ofisten, bilgisayarın karşısında saniyeler içerisinde yapılabilmektedir.

Haritacılıkta kullanılan temel jeodezik hesaplamalar, bu işle uğraşanların her zaman ihtiyaç duydukları bir konudur. Gerek lisans eğitimi sırasında, gerek arazi çalışmalarında ve gerekse projelerde kullanılmak üzere bu programlara sık sık ihtiyaç duyulmaktadır. Kişiler bu ihtiyacı karşılamak için ya kendileri bir programlama dilini kullanmakta ya da başkalarının yazdıkları hazır programları bulmaya çalışmaktadırlar. Genellikle görsellikten uzak ve kullanıcı kolaylığı olmayan bu programların dağılımı hem sınırlı kalmakta hem de el değiştirdikçe kullanım zorlukları başlamaktadır.

Yine bu tür programlar için en büyük dezavantaj, belli bir platforma bağlı çalışmak zorunda olmalarıdır. Burada mantık, yazılıma uygun bir sistem aramak yerine, her sistemde çalışmaya müsait bir programın seçilmesi olmalıdır.

Bu çalışmada, kullanıcıların ihtiyaç duyabilecekleri temel jeodezik uygulamaları bir internet sayfasında yayınlarak genel bir program arşivi oluşturmanın ilk adımları atılmıştır.

Başlangıçta programları yapılmak istenen uygulamalar tespit edilmiştir. Daha sonra internette sunulacak bu programların hazırlanacağı uygun bir dil araştırılmıştır. Öncelikle görsel bir dil olan Delphi ile çalışmalara başlanmıştır. Programcılık için kolay olan bu dilden, internet ortamına çok uygun olmadığı ve Windows işletim sistemine bağımlı yapısı nedeniyle vazgeçilmiştir. Java diline bakıldığı zaman ise bu dezavantajların ortadan kalktığı ve bu çalışma için oldukça uygun bir tabana sahip olduğu görülmüştür.

Java programlama dili temel olarak Application ve Applet olarak ikiye ayrılmaktadır. Applet teknolojisi Java'nın internetle ilgili olan kısmıdır. Bu çalışmada temel jeodezik hesaplamalar Applet ile hazırlanmış ve internet ortamına aktararak kullanıcıların istifadesine sunulmuştur. Hazırlama aşamasında bu programlar, internet ortamı olarak ücretsiz web sitesi veren [turkiye2000](http://utils.turkiye2000.com) sitesinin <http://utils.turkiye2000.com> ve <http://atlas.cc.itu.edu.tr/~celikn/geoinfo> adreslerine yüklenmiştir. Ayrıca hatırlaması kolay olması bakımından, bu sayfaya <http://geoinfo.cjb.net> adresinden de ulaşılabilir. Daha sonra olabilecek değişikliklerde, bu adreslere eklenecek bir bağlantıyla kullanıcıların yönlendirilmesi sağlanacaktır.

İkinci bölümde interaktif web uygulamalarının günümüzdeki kullanım alanları açıklanmıştır. Yine bu bölümde, bu çalışma için seçilen Java dilinin tanıtımı yapılarak, Java'nın çalışma şekli ve özellikleri verilmiştir. Ayrıca, Java'nın diğer dillerle olan karşılaştırılması yapılmış ve neden bu dilin seçildiği hakkında fikir verilmiştir.

Üçüncü bölümde, bu uygulamalardan faydalanabilecek hedef kitle tespit edilmeye çalışılmıştır. Applet teknolojisinin çalışma yapısı hakkında bilgi verilmiş ve hazırlanan uygulamaların geçtiği aşamalardan bahsedilmiştir. Uygulamalardan bir örnek seçilerek internet ortamındaki yapısı anlatılmıştır.

Dördüncü bölümde ise yapılan programlar, arayüzleri, kullanma şekilleri ve formülasyonları verilmiştir.

Son bölümde, bu çalışmadan çıkarılan sonuçlar ve öneriler sunulmuştur.

## 2. İNTERAKTİF WEB UYGULAMALARI

Günümüzde interaktif web hizmetleri, hızla geleneksel statik web sayfalarının yerini almaktadır. İnternet teknolojisinin ilk zamanlarında web sayfaları sadece statik bir görünüm taşımaktaydı. Tıpkı bir duyuru panosu şeklinde, kullanıcı sadece metin ve şekillerden oluşan sayfalar arasında gezinmekte, etkileşimli bir işlem yapamamaktaydı.

CGI(Common Gateway Interface), ASP(Active Server Pages) ve Java gibi interaktif işlemleri destekleyen dillerin ortaya çıkmasıyla beraber, internet sayfalarından etkileşimli hizmet veren sitelerin sayısı oldukça artmıştır. İnteraktif servis sağlayan yukarıdaki programlama dillerinin temel görevi kullanıcı ile sunucu arasındaki iletişimi sağlamaktır. Bağlanılan sayfadan girilen istekler aracı dil tarafından sunucuya iletilir. Sunucuda bu istek doğrultusunda yapılan işlemlerin sonucu yine bu dil tarafından kullanıcıya aktarılır.

Bu konuda internet ortamındaki en temel örnek arama motorlarıdır. Bu hizmeti veren herhangi bir siteye bağlantı kurulduğunda, ilgili bölüme aranması istenilen bilgi ile ilgili referans kelimeler girilip sunucuya gönderilir. Sunucu, veritabanını sorgular ve cevap verebildiği oranda bilgiyi kullanıcıya geri gönderir. Tüm bu işlem adımlarının veri taşıma ve interaktif işlem aşamalarında, yukarıda sayılan programlama dilleri hizmet verir.

İnternetin güvenlik konularındaki gelişmeye bağlı olarak, interaktif ticari web uygulamalarında önemli ilerlemeler kaydedilmiştir. İnternet ortamında yapılacak bir alışveriş kredi kartıyla yapılabildiği için, güvenlik ile ilgili problemlerin halledilmesi gerekmektedir. Günümüzde bu sorunlar büyük ölçüde çözülmüştür. İnteraktif işlemler sayesinde, saniyeler içinde bilgi kaynaklarına ulaşmak, bankaya gitmeden fatura ödemelerini yapmak, süper marketten bilgisayar başında alışveriş yapmak ve borsa işlem talimatları vermek gibi hizmetler gittikçe artan bir ivmeyle yaygınlaşmaktadır.

## 2.1 Genel Uygulamalar

İnteraktif web uygulamaları genel olarak, ticari, enformatik, eğlence ve bilimsel uygulamalar olarak dört bölüm altında incelenebilir. Ticari işlemlerini internet ortamına taşımak isteyen kuruluşlar, web sayfalarında bu işlemlere yer vermektedirler. Bunun artmasına etken olan en büyük gelişme, kredi kartıyla işlem yapabilme seviyesinde güvenlik sistemlerinin gelişmesidir.

Ticari uygulamalar olarak interaktif bankacılık işlemleri, borsa işlemleri, günlük alışverişler, kitap ve dergi satın alma gibi işlemler sayılabilir. Hatta bazı otomobil firmaları özel modellerinin satışını sadece internet üzerinden yapmaktadırlar.

Enformatik alanındaki uygulamalar, bilgiye ulaşımı sorgulamalı bir halde kullanıcıya sunan sitelerdir. Bu konuda örnek olabilecek siteler arasında haber portalları başta gelmektedir. Ulusal ve uluslararası yayın yapan tüm gazete ve televizyonlar, haber arşivlerini interaktif bir şekilde kullanıcılara sunmaktadırlar. Ayrıca bilimsel kuruluşların veya kişilerin çalışmalarını ilgili web sitelerinden interaktif sorgulamalar sayesinde takip etmek olanaklıdır.

Eğlence alanındaki uygulamalar her kesime hitap edecek şekilde yaygınlaşmaktadır. Kişisel bilgisayarlarda oynanmak için üretilen eğlence amaçlı uygulamalar, interaktif programlama dilleri sayesinde web ortamına taşınarak farklı coğrafyalardaki kişileri biraraya getirebilmektedir. Özellikle sohbet hizmeti veren sitelerin çoğalmasıyla beraber yeni arkadaşlıkların oluşması yaygınlaşmıştır.

İnteraktif web uygulamalarının kullanıldığı alanlardan birisi de bilimsel ve eğitim amaçlı platformlardır. Bu uygulamalar ile birlikte kullanıcılar artık internet üzerinden diploma alma şansına bile sahiptirler. Bu çalışmalara örnek olarak; yabancı dil eğitimleri, temel eğitim kursları, üniversite hazırlık kursları, çeşitli bilim dallarında hazırlanan programlar gösterilebilir. Kullanıcı, interaktif olarak hazırlanan sitelerden, herhangi bir konu ile ilgili forumlara katılabilir veya karşılaştığı sorunlarla ilgili mesaj bırakarak daha önce bu problemleri yaşamış veya çözüm bulmuş kişilerin yazılarına okuyup sorunlarını hızlı bir şekilde halledebilmektedir. Konusuyla ilgili bazı temel bilgisayar programlarına ihtiyaç duyan bir kimse, daha önceden hazırlanmış programların bulunduğu siteyi kullanarak sonuç alabilmektedir.

## 2.2 Jeodezik Amaçlı Web Uygulamaları

Jeodezik amaçlı web uygulamaları, özellikle hesaplama ve formülasyon yükünün fazla olduğu bu alanda oldukça yaygın bir şekilde kullanılmaktadır. Bu uygulamaları kendi arasında GPS uygulamaları, CBS uygulamaları, astronomi uygulamaları ve temel jeodezik uygulamalar olarak dört bölüme ayırmak mümkündür.

GPS uygulamaları ile ilgili interaktif olarak hazırlanmış en kapsamlı sayfalardan birisi, Amerikan NGS (National Geodetic Survey) dairesinin hazırlamış olduğu OPUS (Online Positioning User Service) yazılımıdır. Şekil 2.1'de görüldüğü gibi kullanıcı <http://ngs.noaa.gov/OPUS/> adresine bağlanarak sayfada yer alan formu doldurur ve GPS gözlem kütüğünü kendi bilgisayarından sunucuya gönderir. Sunucu, bu noktaya en yakın üç sabit istasyonun verilerini de hesaba katarak çözümü yapar ve sonucu yine interaktif bir şekilde kullanıcıya ulaştırır.

OPUS Home Page - Microsoft Internet Explorer

OPUS Online Positioning User Service

[Dec 12, 2001]  
1 addition to the list of antenna types:  
TOP LEGANT3 UHF - Topcon Legant 3 with UHF antenna

**What is OPUS**

**OPUS Guidelines**

**GPS Height Measurements**

**Antenna Types**

**Output Description**

**Discussion**

**Expected Precisions**

**Latest Precise Orbit**

**FAQs**

**Questions - Comments**

**OPUS File Upload**

1.   
Enter your email address

2.

Enter your dual frequency RINEX file in the standard 8.3 naming convention for example: gain0940.01a or home1110.99a  
The extension must be 95a, 96a, 97a, 98a, 99a, 00a, 01a or 02a

3. NONE NONE  
Select the antenna type

4. 0.0 meters 5. 0 none 6.

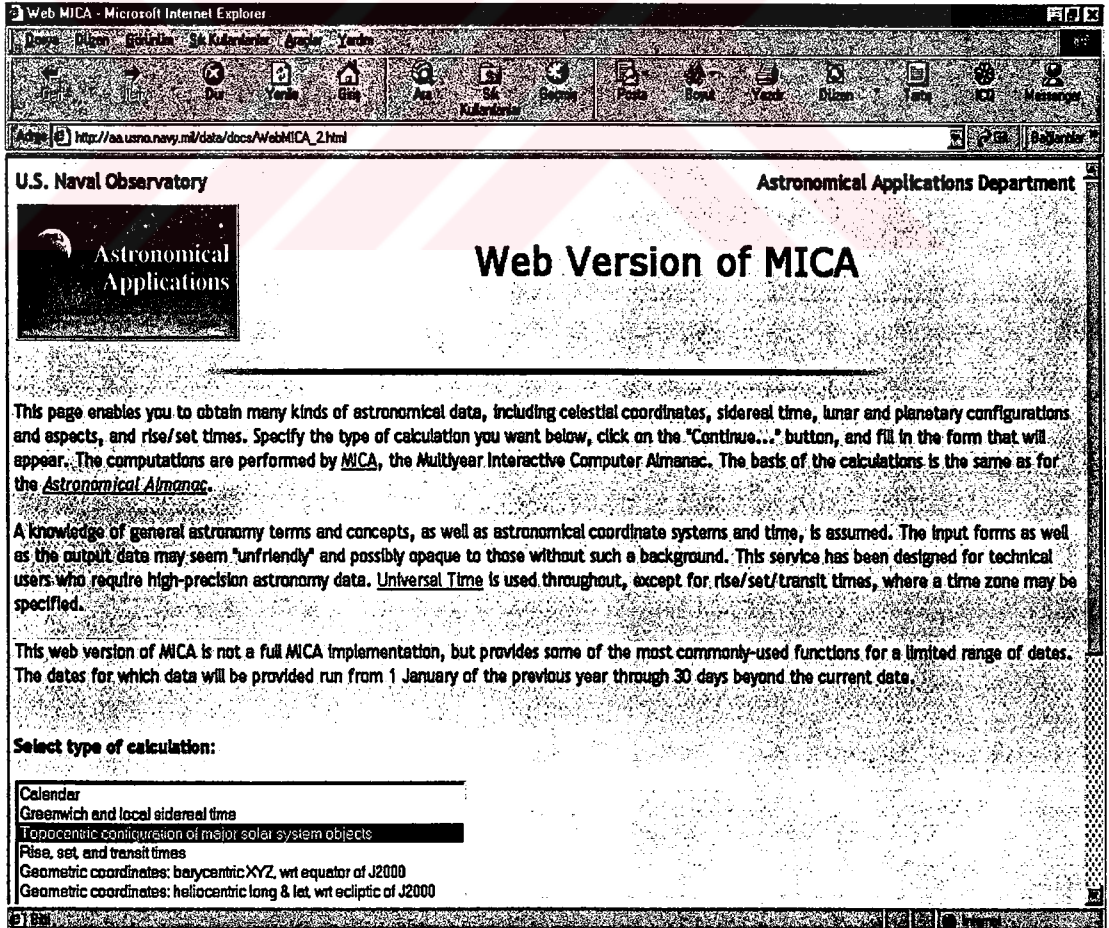
Enter the antenna height Optional: State Plane Coordinates

Your data must have a minimum of 2 hours of observations!

Şekil 2.1: İnteraktif GPS veri değerlendirme uygulamasına örnek sayfa

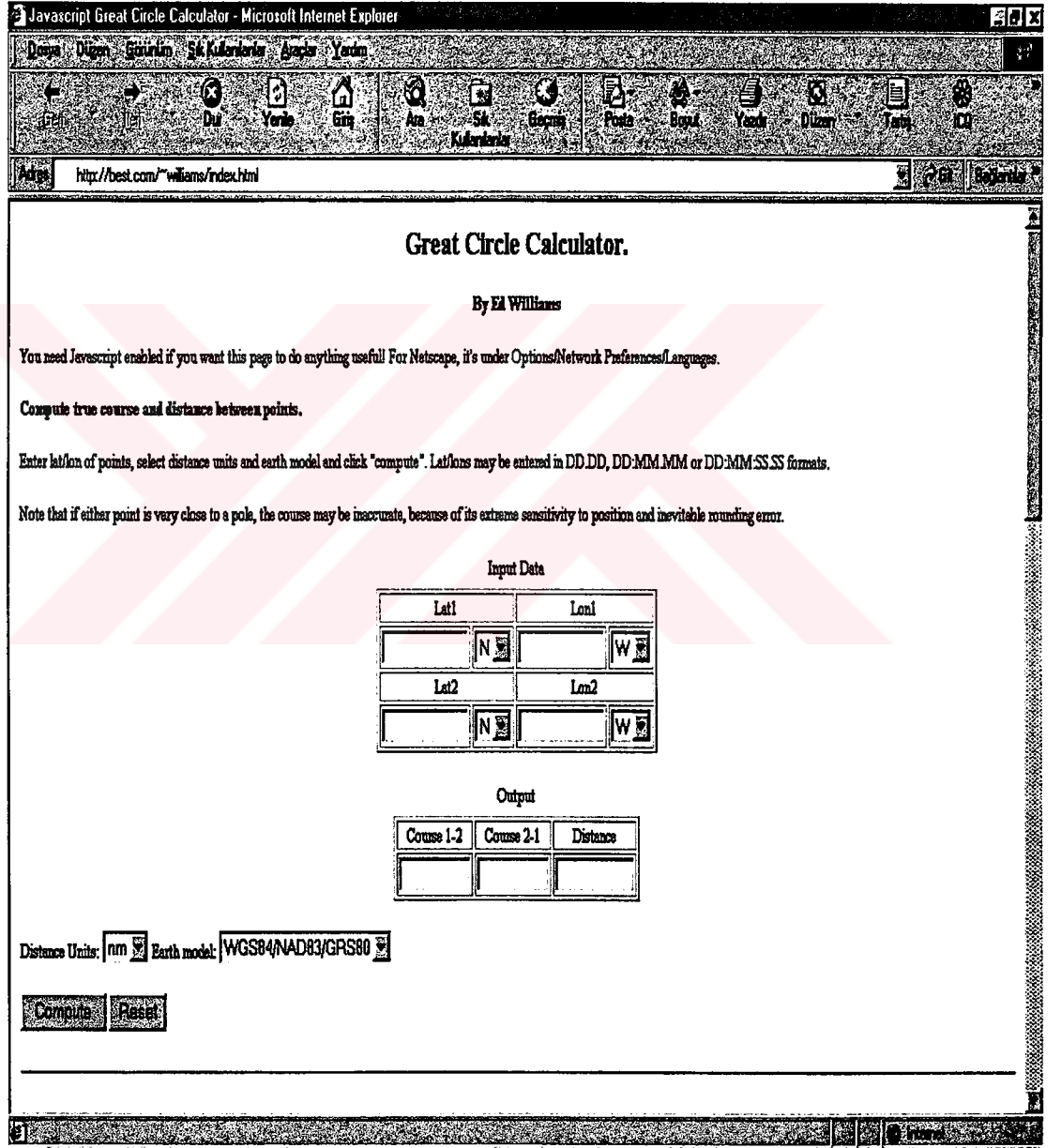
Coğrafi bilgi sistemlerine ilişkin yapılan uygulamalar zaten başlı başına interaktif özelliklere sahiptir. Kuruluşlar, bu alanda yapılmış uygulamaları internet ortamına taşıyarak kullanıcılara sürekli güncellenen bir hizmet vermektedirler. Örneğin kent bilgi sistemi web ortamına aktarılmış olan Ankara planı üzerinde, çeşitli sorgulamalar vasıtasıyla güzergah seçimi yapılabilir. Bu sistemin sadece bir paket program şeklinde dağıtıldığı düşünülürse, kullanıcının güncel değişikliklerden mahrum kalacağı açıktır.

Astronomik uygulamalar jeodezinin temel fakat bir o kadar da hesap yükünün ağır olduğu alanlardan birisidir. Bu alanda meydana getirilmiş oldukça fazla yazılım mevcuttur. Bunlardan birisi de Amerika Astronomik Uygulamalar Departmanının hazırladığı MICA(Multiyear Interactive Computer Almanac) yazılımıdır. Bu yazılım paket program olarak pazarlandığı gibi, aynı zamanda internet versiyonu da mevcuttur. [http://aa.usno.navy.mil/data/docs/webMICA\\_2.html](http://aa.usno.navy.mil/data/docs/webMICA_2.html) adresinden yayınlanan bu yazılımın ana sayfası şekil 2.2’de görülmektedir.



Şekil 2.2: MICA yazılımı ile interaktif astronomik hesaplamaların internet sayfası

Temel jeodezik uygulamalarla ilgili interaktif web sayfaları genellikle toplu bir halde bulmak mümkün olmamaktadır. Jeodezi ve fotogrametri bölümüne sahip bazı üniversitelerin internet sayfalarından veya bu işle ilgilenen bazı kimselerin web sayfalarından ancak yoğun bir araştırmayla ulaşılabilmektedir. Bu sayfalardan örnek olarak 1. ve 2. temel ödev çözümü yapan bir sayfa şekil 2.3’de verilmiştir.



Şekil 2.3: 1. Temel ödev çözümü yapan interaktif web sayfası örneği

## 2.3 İnteraktif Web Tasarımı için Programlama Dili Seçimi

### 2.3.1 Java programlama dili

Java, Sun Microsystems tarafından geliştirilen, nesne-yönetimli bir programlama dilidir. Java ile ilgili çalışmaların başlangıçtaki amacı, platformdan bağımsız bir programlama dili ve basit ev aletleri için bir işletim sistemi ortaya çıkarmaktı. İlk düşünce, C++ programlama dilini kullanmaktı. Fakat çalışmalar ilerledikçe, C++'ı geliştirmek yerine yeni bir programlama dilinin oluşturulmasının amaca daha çok hizmet edeceği düşünüldü ve Java, programlama dilleri arasındaki yerini aldı [1].

Günümüzde Java'nın internet uygulamalarında en yoğun kullanılan dil olmasına rağmen, başlangıçtaki amacı internete yönelik değildi. Bu sebeple Java, genel amaçlı programlamacılığa da oldukça uygun bir dildir.

Kaynak kodlarını makine diline çeviren geleneksel derleyici programlama dillerinin tersine, Java derleyicisi kaynak kodunu kendisinin yorumlayabileceği komutlara dönüştürür. Yani, bu dilin temelini oluşturan C ve C++'ın tersine, Java yorumlayıcı bir dildir. Geleneksel olarak, bir programlama dilinin derleyici programı eğer kendi orjinal diliyle yazılmışsa , o dil oluşumunu tamamlamış demektir [1]. Bu geleneğe göre Java oluşumunu tamamlamış bir dildir.

### 2.3.2 Neden Java

Bu çalışmada Java'nın kullanılmasının en temel nedenleri, nesne yönetimli yapıya ve platformdan bağımsız çalışma yeteneklerine sahip olmasındandır.

Nesnel programlamaya uygun bir dilin bu çalışmadaki en büyük avantajı ise, hazırlanan programların daha sonraki çalışmalara adaptasyonunun kolay olmasıdır. Bu şekilde, yeni uygulamaları tasarlama ve algoritmalarını oluşturma aşamalarında oldukça fazla zaman kazanılacağı düşünülmektedir.

Programcılıkta karşılaşılan platforma bağlı kalma problemi Java'nın sunduğu en önemli özelliklerden birisidir. Kullanıcıların sistem olanaklarını değiştirmeden yararlanabilecekleri uygulamalar, optimizasyon ve zaman tasarrufu açısından oldukça önemlidir.

### 2.3.3 Java'nın Özellikleri

- Java basit bir programlama dilidir

Java'nın nesne-tabanlı yapısı ve kod yazılım özelliklerinin büyük bir çoğunluğu C ve C++ dillerinden uyarlanmıştır. Ortak özellikleri sebebiyle, C ve C++ programcıları Java'yı öğrenmek için daha az zaman harcamaktadırlar.

C++'ın gözardı edilen elemanlarından dolayı, Java oluşturulurken, "C++ -" olarak tanımlanmıştır. C++'ın "Pointer aritmetiği" ve hafıza yönetimi gibi karmaşık kısımları Java'dan çıkarılmıştır [1,2]. Bu elemanların kullanımı oldukça karmaşıktır. Örneğin, çok uzun yazılmış bir program kodunda bir "pointer" hatasını bulmak çok güçtür. Hafıza yönetimi Java'da otomatik olarak sunulan bir özelliktir. Dolayısıyla kullanıcılar hafızayı temizleme veya yer açma gibi sorunlardan kurtulmuş olmaktadır.

Java'yı basit kılan diğer bir özelliği de temel veri tipleri ve nesnelidir. Bu dil değişkenlerle ilgili oldukça katı kurallar uygulamaktadır. Hemen hemen tüm durumlarda değişkenler, tanımlandıkları veri tipiyle kullanılmalıdır. Bu düzenleme hataları henüz program derlenirken düzeltmeye olanak sağlar.

- Java nesne yönetimli bir dildir

Nesne tabanlı programcılık, bir yazılımı düzenlemede ve geliştirmede oldukça etkili bir yöntemdir. Nesne yönetimli programcılık kısaca, nesne olarak adlandırılan bileşenlerin organize edilmesidir. Bu nesnelere birbirlerinden bağımsız olarak bulunurlar. Diğer nesnelere iletişim kurmak için ve bu nesnelere yapacakları işleri bildirmek için kuralları vardır.

Java, nesne yönetim konseptlerini C++'dan almıştır. Bir programlama dilinin nesne yönetimli olması bazılarının göre faydalı bir özellik gibi görünmeyebilir. Hatta diğer programlama dillerinde tecrübesi olsa bile nesne tabanlı programcılık bazılarının ürkütücü gelebilir. Halbuki nesne tabanlı yazılmış programların anlaşılması ve hata ayıklaması kolay ve aynı zamanda diğer projelerle adaptasyonu oldukça rahattır.

Java dili, temel değişken tipleri, sisteme giriş/çıkış yetenekleri ve diğer fonksiyonları sağlayan bir takım nesne kütüphanelerine sahiptir. Aynı zamanda, ağ desteği, internet

protokolleri ve grafik kullanıcı arayüzü fonksiyonlarını sağlayan kütüphaneleri de vardır. [1,2]

- Java güvenli bir dildir

Java'nın başarısındaki diğer bir etken, onun güvenli bir dil oluşudur. Herhangi bir internet sayfasından çalışabilen Java uygulamasına "applet" denir. Diğer tüm Java programları "application" olarak adlandırılır. Bir internet sayfasında bir applet ile karşılaşıldığında, applet uygulaması sayfadaki yazı ve görüntü uygulamaları ile birlikte kullanıcının bilgisayarına yüklenir. Bundan sonra applet uygulaması kullanıcının bilgisayarında çalışmaya başlar. Bu noktadan itibaren tehlike başlamaktadır. Çünkü programlar çalışmaya başladığında birçok zararlı şey meydana gelebilir: virüsler, truva atları vb.

Java, farklı seviyelerde güvenlik seçenekleri sağlamaktadır. İlk olarak bu dil, zarar verici kodların çalışmasını engelleyici bir dil olarak dizayn edilmiştir. Bu bağlamda pointer aritmetiğinin bu dilden çıkarılmış olması büyük bir adımdır. Pointer oldukça güçlü bir özelliktir fakat, bir programın erişimine izin verilmeyen kısımlarına veya bilgisayarın hafızasındaki değişime izin verilmeyen bölümlerine ulaşmak için zorlanabilirler. Pointer özelliğinin, sınırlı bir kısmı dışında bu dilden çıkarılmasıyla Java daha güvenli bir dil haline gelmiştir.

Diğer bir güvenlik seviyesi ise kod onaylayıcısıdır. Bir java programı çalışmadan önce, kod onaylayıcısı her bir kodu şüpheli herhangi bir şey olup olmadığına karşı kontrol etmektedir.

Yukarıda sayılan önlemlere ilaveten Java, applet'lere uyguladığı bazı özel güvenlik kiplerine sahiptir. Kullanıcının disk sürücülerine zarar vermesini önlemek amacıyla, bir applet kullanıcının sisteminde dosya açma, okuma ve yazma gibi işlemleri yapamaz.

Çoklu güvenlik seviyeleri ve bu seviyeleri sürekli geliştirmek için harcanan çabalar sayesinde Java, internet ortamında bir program yapmak için en güvenli dildir [1,2].

- Java platformdan bağımsız bir dildir

Java programları işletim sisteminden bağımsız çalışırlar. Yazılımların çoğu sadece bir işletim sistemi için geliştirilir. Java ise tamamen platformdan bağımsız bir dildir.

Java'nın değişken tipleri, tüm Java geliştirme platformlarında eşit büyüklüğe sahiptirler [1,2]. Herhangibir yerde oluşturulmuş bir Java nesnesi hiçbir değişikliğe uğramadan tüm işletim sistemlerinde çalışabilir. Bu durum tak ve kullan özelliğine sahip bilgisayar parçalarına benzemektedir. Genellikle monitörlerde karşılaşılan bu durum, herhangi bir yazılıma ihtiyaç duymadan, cihazı istenilen sistemde hemen kullanmayı sağlar. Sisteme bağlı çalışan yazılımlar, öncelikle farklı sistemlerde farklı derleyicilere ihtiyaç duyarlar. Yazılım kodunu öncelikle kullanacağınız sistemin gereksinimlerini karşılayacak şekilde uyarlamamız ve bu sistemdeki derleyicisiyle derlemek gerekir.

- Java Çok Kanallı (multi-threaded) Bir Dildir

Çok kanallılık özelliği, bir yazılımın aynı anda birden fazla görevi yürütebilmesidir. Birçok işletim sistemi, çok kanallıdır. Örneğin, windows işletim sisteminde bir yandan yazı yazarken, bir yandan da müzik dinleyebilirsiniz. Çok kanallı bir programlama dili bu özelliği programlara yansıtabilir. İşte Java, çok kanallı programlar yazmaya ve bu programları güvenli bir şekilde çalıştırmayı sağlayacak bileşenlere sahiptir [1,2].

#### **2.3.4 Java'nın Çalışma Biçimi**

Java, bir programlama dili, bir çalışma zamanı sistemi, bir yazılım geliştirme elemanları topluluğu ve uygulama arayüzü programlarından oluşan bir yazılımdır. Bu sayılan elemanlar ve aralarındaki ilişki şekil 2.4'de görülmektedir [3].

Şekil 2.4'de görüldüğü gibi, bir program yazılımcısı, daha önceden tanımlanmış uygulama arayüz programlarını kullanarak, kaynak program kodlarını oluşturur. Java derleyicisi vasıtasıyla programı derler. Bu, derlenmiş kaynak kodunun oluşumuyla sonuçlanır. Derlenmiş kod, java çalışma sisteminin merkezini oluşturan Java sanal makinesinin çalıştırabileceği bir formattadır. Java sanal makinesi, işletim sisteminin ve bilgisayar donanımının olanaklarını kullanarak çalışan ve yazılımın

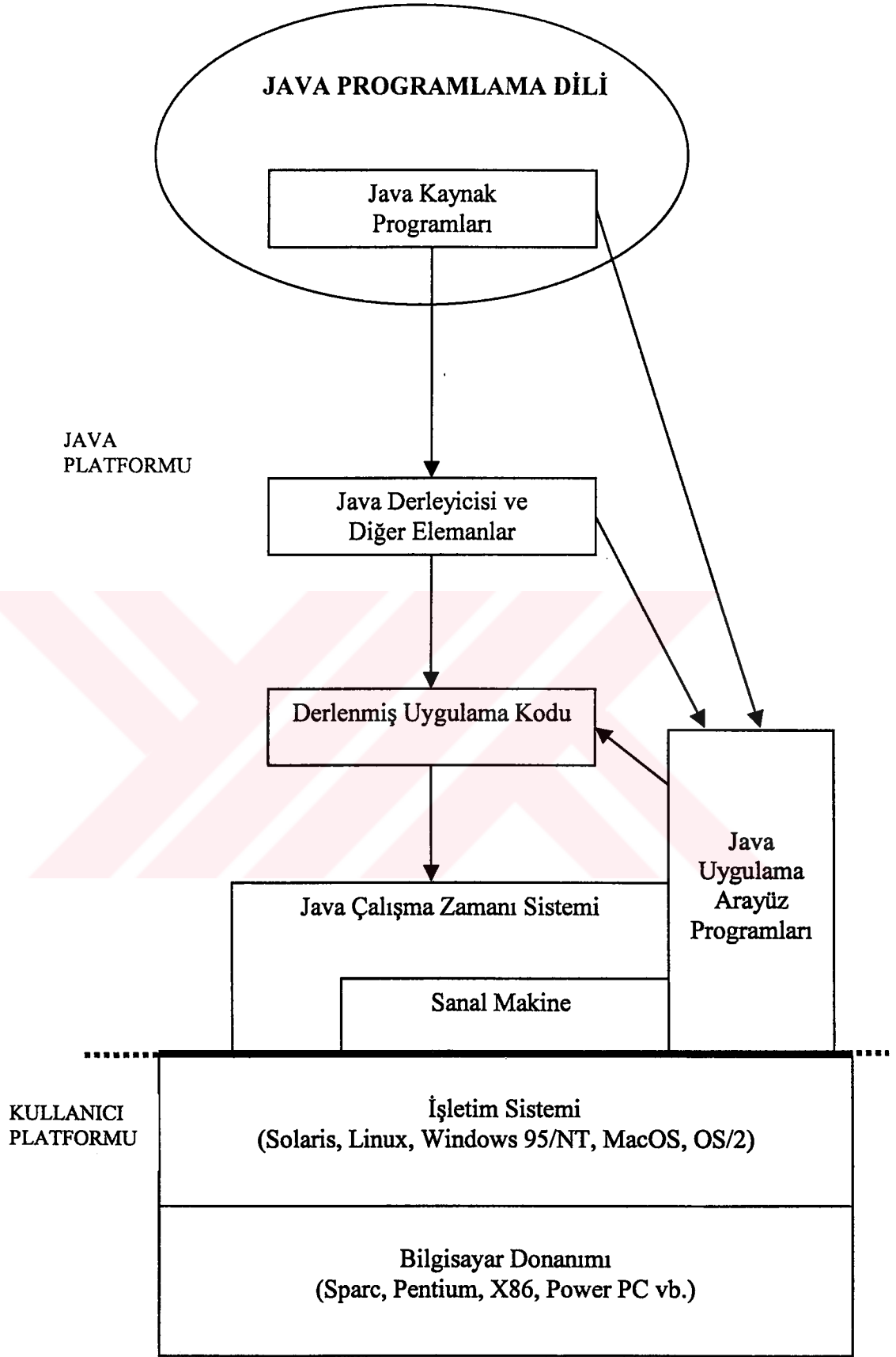
içine yerleştirilmiş bir mikroişlemci gibi düşünülebilir. Halbuki Java sanal makinesi gerçek bir mikroişlemci olmadığı için, derlenmiş java kodu kullanıcı bilgisayarının makine komutlarıyla direkt çalışmaktan ziyade yorumlanarak işleme konmaktadır. Java çalışma anı sistemi sanal makine ile birlikte kullanıcı arayüzü programlarını ilgili işletim sistemi ve bilgisayarda çalıştırmaya yarayacak dinamik bağlantı kütüphaneleri gibi ilave yazılımlardan oluşur [3].

Java uygulama arayüz programları, kullanıcı işletim sisteminin yerel pencereleme ve ağ işletim yeteneklerini artırıcı sayısız bağlantılar kuran önceden tanımlanmış yazılım paketlerini ihtiva eder. Java uygulama arayüz programları, java'nın bulunduğu tüm işletim sistemleri için geçerlidir.

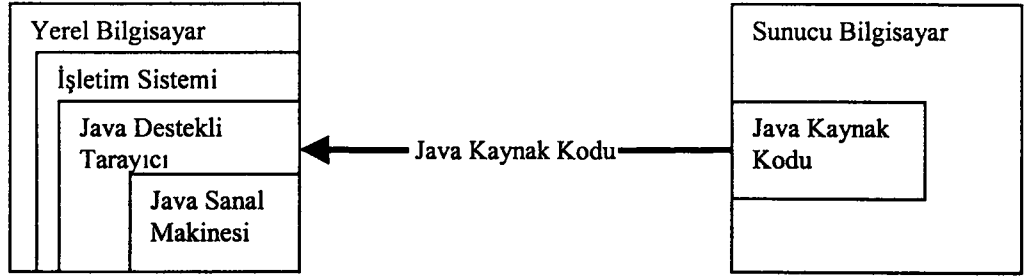
Java'nın taşınabilirliğinin anahtarları, onun çalışma anı sistemi ve uygulama arayüz programlarıdır. Çalışma anı sistemi, Sun Microsystems'in tüketici elektroniği ile ilgili bir yazılım platformu kurmak için yaptığı ilk çalışmalardan doğan oldukça sıkı yapılı bir sistemdir. Bu platform, varolan herhangi bir mikroişlemci etrafında dizayn edilmemiştir. Basit ve etkili olabilmesi için en baştan tasarlanmıştır ve herhangi bir donanım yapısına bağlı kalmadan oluşturulmuştur. Çalışma anı sisteminin basit, etkili, bütünleşik ve bağımsız yapılı tabiatı, onun değişik sistemlere uyarlanabilir ve yüksek performanslı olmasını sağlamıştır.

Java uygulama arayüz programlarına dahil edilen pencereleme ve ağ işletim özellikleri programcıların hem etkili hem de platformdan bağımsız yazılım geliştirmelerine olanak sağlar. Örneğin, "Ada" programlama dili yüksek standartlı ve çoğu işletim sistemlerince desteklenen bir dildir. Fakat Ada uygulamaları henüz çok değişkenli değildir. Çünkü Ada dili henüz tüm platformlarda çalışabilen genel bir uygulama arayüz programına sahip değildir. İşte Java'yı Ada'dan ve diğer tüm dillerden ayıran en önemli özelliği bütün işletim sistemleri için tek bir evrensel fakat güçlü uygulama arayüz programlarının varlığıdır [3].

Şekil 2.5, Java kaynak kodunun nasıl transfer edildiğini göstermektedir. Şekilde kaynak kodunu saklayan bir bilgisayar görülmektedir. Yerel bilgisayardaki bir kullanıcı Java destekli bir tarayıcı vasıtasıyla interneti kullanıp karşıdaki bilgisayara bağlandığında, kaynak kodu kendi bilgisayarına transfer edilmektedir.

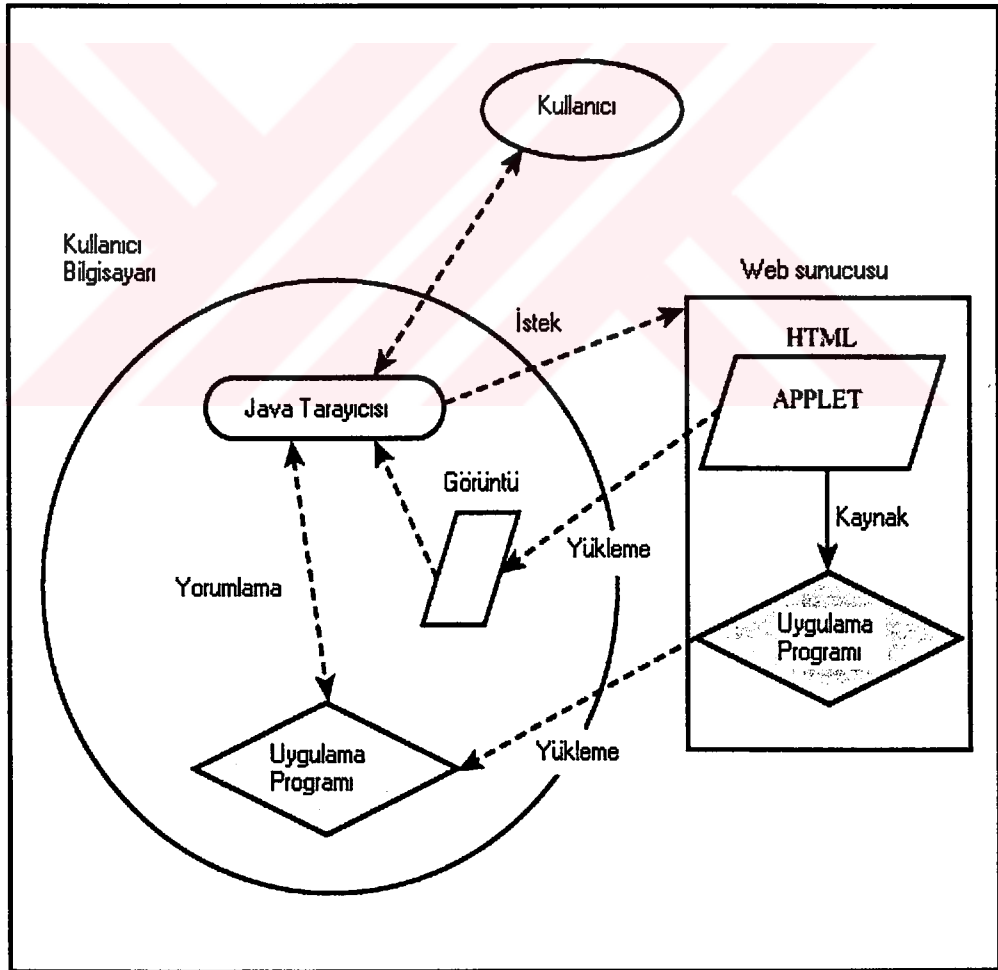


Şekil 2.4: Java işletim sistemi elemanları ve aralarındaki ilişkileri gösteren diagram.



Şekil 2.5 : Java kaynak kodunun internet üzerinden transferi

Java destekli tarayıcının içinde yer alan sanal makine, kaynak kodlarını yorumlar ve istenilen program komutlarına dönüştürür [2].



Şekil 2.6: Java uygulamasının kullanıcı bilgisayarında nasıl görüntülendiğini gösteren diagram.

Java interneti tamamen deęiřtirmektedir; çünkü önceki web yazılım sistemlerini kullanarak ulařılamayacak interaktiflięe ve bilgi eriřimine zenginlik getirmektedir. Java, programcılara internet üzerinden daęıtabilecekleri ve çok çeřitli bilgisayarlar da çalıştırabilecekleri bir yazılım geliřtirmelerine olanak saęlar. Sonuç programın içerięi, baęlanılan siteyi web sunucusundan kullanıcının ekranına tařır. Őekil 2.6, Java'nın interaktif özellięinin zenginlięini ve sunucudan çok kullanıcı merkezlilik özellięini göstermektedir [1].

### 2.3.5 Java'nın Dięer Programlama Dilleriyle Karřılařtırılması

Java řu anda hiębir programlama dilinin kendisine yaklařamayacaęı kadar iyi bir konumdadır [4]. Fakat yine de bu, Java'nın herřeyi yapabileceęi anlamına gelmez. Bir programlama dili olarak Java ve dięer genel amaęlı programlama dilleri, avantajları ve internet ortamında nasıl kullanıldıkları aęısından karřılařtırılabilir.

- **Java ve Perl**

Perl programlama dili, uzun zaman CGI (Common Gateway Interface) programcılıęının tercih edilen dili olmuřtur. Çok az sayıda dil sunucu tabanlı programlar için Perl ile saęlanan kolaylıkları verebilmiřtir. Perl'in gerçek gücü çok hızlı metin iřlemlerinde ve mükemmel cümle iřlemleri özelliklerinde yatmaktadır. Sadece metin iřlemleri söz konusu olduęunda, Perl genellikle Java'dan daha hızlıdır. Fakat yine de Perl Java'nın güçlü grafik yeteneklerine sahip deęildir.

Java daęılımcı bir dildir. Yani Perl gibi sadece sunucu üzerinde çalışmak zorunda olan bir dil deęildir. Sunucunun iřlem gücünün korunmasına ilaveten Java applet'leri bir siteden dięerine daha kolay bir řekilde tařınabilmektedir. Java'yı kullanıcı olarak çalıştırmak demek, kaynak kodlarda veya uygulama programlarında hiębir deęiřiklik yapmadan her çeřit sunucuda saklayabilmek ve çalıştırabilmek demektir. Perl dosyaları da siteden siteye tařınabilmektedir fakat her tařımada sonra bazı uyarlamalara gerek duymaktadırlar.

Perl ile karřılařtırıldıęında Java applet'lerinin en büyük kısıtlaması Java'nın sunucuya bilgi yazamamasıdır [4]. Yine de bu Java'nın programlama gücü eksiklięi deęil, sadece bir güvenlik kısıtlamasıdır.

## • Java ve Diğer Programlama Dilleri

Perl'in yanısıra Java diğer programlama dilleri ile de karşılaştırılabilir. Programlar iyi bilinen ve popüler avantajları olan UNIX shell, C veya C++ dillerinde yazılabilir. Bu diller Java'nın vermediği bazı düşük seviyede programları sağlayabilirler. Programcıya uygulamayı spesifik bir işletim sistemi veya donanım için özelleştirmeyi sağlarlar. Aynı zamanda uzun bir zamanda kullanımda oldukları için herbirisi ile yapılmış oldukça fazla uygulama bulunmaktadır. Dolayısıyla şu an için Java'nın her alanda bu dillerin yerini almasını beklememek gerekir.

Java sağlam ve güçlü dağılımı olan bir dildir. Aynı zamanda Java'nın güvenlik anlayışı sözkonusu dillerin oldukça ilerisindedir. Fakat yine de bu güvenlik anlayışı tüm uygulamalar için gerekli değildir. Java kaynak kodunun yeniden kullanılmasını destekler, fakat kodlamanın büyük bir kısmı diğer bir dilde yapılmışsa, belki de o dilin kullanımı daha uygun olacaktır. Java'nın sistemden bağımsız olarak kullanılabilen yapısı önemli bir avantaj olarak görülse bile, örneğin çok büyük oranlarda matematiksel işleme gerek duyulan alanlarda C'nin kullanımı çok daha hızlı olabilir.

Amaca göre programlama dilini seçmek en doğrusu olacaktır. Eğer proje sunucuya birşeyler yazmayı gerektiriyorsa Perl veya C++ en uygun tercih olabilir. Fakat internet üzerinden dağıtılabilen uygulamaların yapılması planlanıyorsa, bu konuda Java'nın rakibi yoktur [4]. Amacımızın uygun dili seçmesine izin vermemiz gerekir.

## • Java ve Paket Programlar

Genel amaçlı yazılımlar arasında Java en güvenli olmasına rağmen, güvenlik özellikleri oluşumunu henüz tamamlamamıştır. Bir dilin kullanım alanı ne kadar artarsa güvenlik engelini aşmak için yapılan girişimler de o derecede artacaktır. Programlama dili ne kadar gelişirse güvenlik konuları da o derecede gelişecektir. Yine de Java'nın yaygın kullanımı birçok programcının bu konuları gözardı etmesine sebep olmaktadır.

Java uygulamalarının transfer hızı konusundaki sorunlar, eğer ISDN veya kablo modemler internet ortamında bir standart haline gelirse, büyük ölçüde çözülmüş

olacaktır. ISDN bağlantısı kullanarak internet erişim hızı 64 Kbps, 128 Kbps veya daha üzerine çıkabilmektedir. Kablo modemlerin kullanımı ise hızı 1 Mbps'ye kadar çıkarmaktadır. Yine de günümüzde internet bağlantı hızının sabit diskimizle aynı hıza ulaşması biraz imkansız görünmektedir.

Java'nın internet programcılığının yanısıra paket programcılık konusunda da rekabet halindedir. Java'nın tek başına çalışabilen programlarının yanısıra diğer dillerle entegre olmuş programları destekleyen uygulamaları da bulunmaktadır [4].

- **Java ve HTML**

Uzun yıllar boyunca HTML web'in tek dili olmuştur. Java'nın onun yerini alması söz konusu değildi. Tüm web tarayıcıları bir web uygulamasını yayınlamak için HTML'in içindeki kalıpları kullanmaktadırlar. Java bu yeteneğin genişletilmiş bir bölümü olup onun yerine geçmemektedir.

Bazı programcılar genişleyen HTML kalıplarının yerini Java'nın alacağını düşünmektedirler. Bir web dökümanını HTML ile düzenlemek yerine bu düzenlemeyi Java'nın yapabileceğini düşünmektedirler. Her ne kadar Java uygulamalarının transfer ve çalışma hızı yavaşsa da, yakın gelecekte bu düşünce hayata geçebilir. HTML dosyaları düz metin biçiminde oldukları için yüklenme süreleri oldukça hızlıdır. Çoğu siteler için de kullanıcılar bunu dikkate almaktadırlar.

HTML dosyaları resim ve ses öğeleri içerdikleri zaman transfer zamanı artmaktadır. Eğer bu dosyalar statik bir durumda ise ve HTML kullanarak başarılı bir şekilde görülebiliyorsa, Java'yı kullanmak için bir sebep olmayabilir. Eğer bu resimlerin dinamik bir görüntü sergilemesi isteniyorsa bununla ilgili bir applet yapmak daha avantajlı olabilir. Eğer ses klibini kullanıcının başlatıp bitirmesi isteniyorsa Java applet'i sayfayı bu konuda daha çok zenginleştirecektir.

Java en iyi HTML ile birlikte çalışabilmektedir; onun yerini almaya çalışmamaktadır. Java kullanıcının interaktif işlemleri için daha iyi yeteneklere sahip olmakla birlikte, sadece zaman konusunda biraz problem olabilmektedir. Java CGI programlarına benzer bir şekilde HTML formlarına veri girişi yapıp sonuç almayı sağlar. Applet uygulamaları, içinde birden fazla web adresi saklayan CGI programlarının

alternatifidir. Hem CGI hem de Java görüntü ve internet uygulamaları için HTML'i kullanırlar.

Gelecekte internet bağlantı hızındaki gelişmeler birden fazla Applet'in kullanılmasında daha çok zaman kazandırıcı bir etken olacaktır. Bu, Applet teknolojisinin günümüzdekinden çok daha yaygın kullanımına olanak sağlayacaktır [4].



### 3. TEMEL JEODEZİK HESAPLAMALAR İÇİN WEB TASARIMI

#### 3.1 Hedef Kitle ve Gereksinimlerin Tespiti

Haritacılıkta kullanılan temel jeodezik hesaplamalar, bu işle uğraşanların her zaman ihtiyaç duydukları bir konudur. Gerek lisans eğitimi sırasında, gerek arazi çalışmalarında ve gerekse projelerde kullanılmak üzere bu programlara sık sık ihtiyaç duyulmaktadır.

Lisans eğitimi sırasında yapılan hesaplamalarda sonuçların kontrolü oldukça önemlidir. Çünkü bu hesaplamaların formülasyonları oldukça fazladır ve en küçük bir işaret hatası bile sonuçları önemli ölçüde etkilemektedir. Öğrencinin bulunduğu sonuçları kontrol edebilmesi veya doğrudan bu programları kullanarak problemi çözmesi açısından bu çalışmanın faydalı olacağı düşünülmektedir. Temel jeodezik hesapların internet ortamından interaktif bir şekilde yayınlanması sayesinde herkesin bu uygulamalara rahatça erişebilmesi hedeflenmektedir.

Arazi ve proje çalışmalarında, ihtiyaç duyulabilecek en basit bir uygulamaya hemen ulaşamamak, bu hesapların kullanıcının kendisinin yapmasına ve vakit kaybetmesine neden olmaktadır. Dünyanın her yerinden ve saniyeler içinde ulaşabileceği bir web sayfasından bu işlemlerini görsel bir şekilde yapabilmesi yine bu çalışmanın hedeflerinden birisidir.

Hedef kitle olarak sadece haritacıların değil, arazide basit GPS ölçmeleri yapan veya elindeki koordinatları başka bir sisteme çevirmek için program arayan herkesin bu çalışmadan faydalanabileceği değerlendirilmektedir. Bu konuyu bir örnekle açıklayacak olursak; basit bir el GPS'i ile arazide fay hattının koordinatlarını belirleyen bir jeolog veya deprem bilimci aletten okuyacağı WGS-84 koordinatlarını elindeki ED-50 datumundaki harita üzerine atmaya kalkarsa, yaklaşık 100-150 m civarında bir konum hatası yapmış olacaktır. Halbuki internet sayfasına girip datum dönüşümü programını kullanırsa, yaklaşık bir metre doğrulukta bir dönüşüm yapmış olacaktır. Bu programı kullanmak için herhangi bir haritacıdan yardım almasına bile gerek yoktur, çünkü sayfanın grafik arayüzü onu yönlendirecek ve kolayca

hesaplama yapmasını sağlayacaktır. Hatta, elde ettiği coğrafi koordinatları yine aynı sayfada bulabileceği düzlem koordinat elde etme programı ile çevirerek sonuçlarını bu formatta sunma imkanı bile bulabilecektir.

Kısacası bu uygulama, haritacılıkla uğraşanların dışında sayfayı inceleyen herkes eğer işine yarayacak bir hesaplama uygulaması bulursa, bunu kolay bir şekilde kullanabilmesi amacına da uyacak şekilde tasarlanmıştır.

### 3.2 Temel Jeodezik Hesapların Web’de Ele Alınma Biçimi

Java programları temelde ikiye ayrılmaktadır: Applet ve Application. Application’lar Java’nın tek başına çalışabilen uygulamalarıdır. Applet teknolojisi ise Java’nın internet ile ilgili kısmıdır. Applet’ler tek başına çalışamazlar; mutlaka bir tarayıcının (browser) içinden çalıştırmak gerekmektedir. Applet teknolojisi, Java’nın nesne yönetimli programcılığına en güzel örnektir. Fonksiyonellik açısından Applet, küçük ve yoğun bir Java Application’ıdır. Fakat programcılık açısından ise Applet bir nesne sınıfıdır [5].

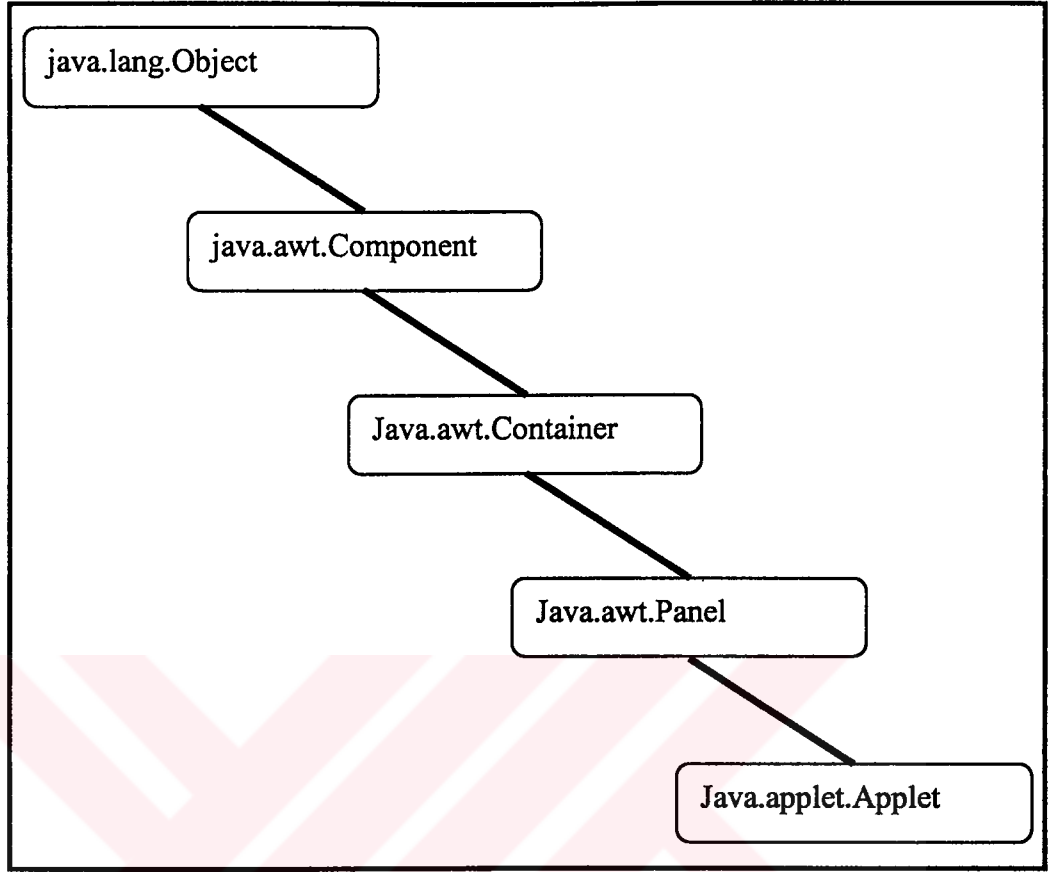
Appletler’in nesne olmalarının arkasındaki düşünce oldukça basittir. Bir nesne olarak Applet, bağlı olduğu üst nesnesinin özelliklerini kullanabilmektedir. Bu sayede nesne yönetimli programcılık teoriden kurtulup pratik bir uygulamaya dönüşmektedir. Şekil 3.1 Java için nesne hiyerarşisini göstermektedir.

Bir Applet programının ilk satırına göz atacak olursak ;

```
public class Deneme extends java.applet.Applet {
```

bu satır “Deneme” Applet’ini kuran satırdır. “Deneme” kelimesinden hemen sonra gelen “extends java.applet.Applet” cümlesi, bu Deneme Applet’inin orjinal Applet sınıfına dayandığını ve bu sınıfın fonksiyonlarını kullanacağını göstermektedir.

Java dilinin nesnelere dayalı bir yapısı ve dizaynı vardır. Bu dizayn stratejisine bağlı olarak, benzer fonksiyonlara sahip metodlar aynı paketlerin içinde gruplandırılmıştır. Örneğin AWT (Abstract Window Toolkit) paketi içerisinde Windows ile çalışan ve kullanıcı arayüzleri oluşturarak ekrana görüntü nesnelere çizdirmeye yarayacak metodlar vardır [5]. Applet paketi ise özellikle appletler üzerine çalışmak için dizayn edilmiştir.



Şekil 3.1 : Java Sınıf Hiyerarşisi

Applet paketi appletler'in oluşturulmasını sağlayacak metodları ihtiva eder. Örneğin appletler sunucudan birtakım görüntü veya ses öğelerinin yüklenmesine ihtiyaç duyar; bu ihtiyacı sağlayacak olan "getImage()" ve "getAudioClip" metodları Applet paketinin birer parçasıdır. Bu metodlar gibi daha birçok yararlı metod Applet paketinin içinde yer alır. Bu paketi inceleyerek hangi metodu kullanacağımıza karar verebiliriz.

Applet paketinin bazı özel fonksiyonlara sahip metodları vardır. Bu metodlar çalışma evresi metodları olarak adlandırılan ve applet'in çalışması esnasında nasıl davranacağını belirleyen metodlardır.

Applet içeren bir web sayfasını yüklediğiniz zaman, applet ekranda görünene kadar bazı aşamalardan geçer. Tüm bu aşamalar süresince applet, çoğu kullanıcıya görünmeyen çok çeşitli görevler gerçekleştirir. Bu aşamalar applet'in başlatılması, çalıştırılması ve sona ermesidir [5].

Başlangıç aşamasında Applet, görüntüleri, ses kliplerini ve çalışmasını sağlayacak diğer bileşenleri kullanıcının bilgisayarına yükler. Hatta bazen “Loading Images...” gibi mesajlar vererek kullanıcıya ekranın arkasında neler olduğu hakkında bilgi verir. Applet’in çalışmasını sağlayacak bütün nesnelere yüklendiği zaman başlangıç aşaması sona ermiştir ve Applet çalışmaya hazırdır.

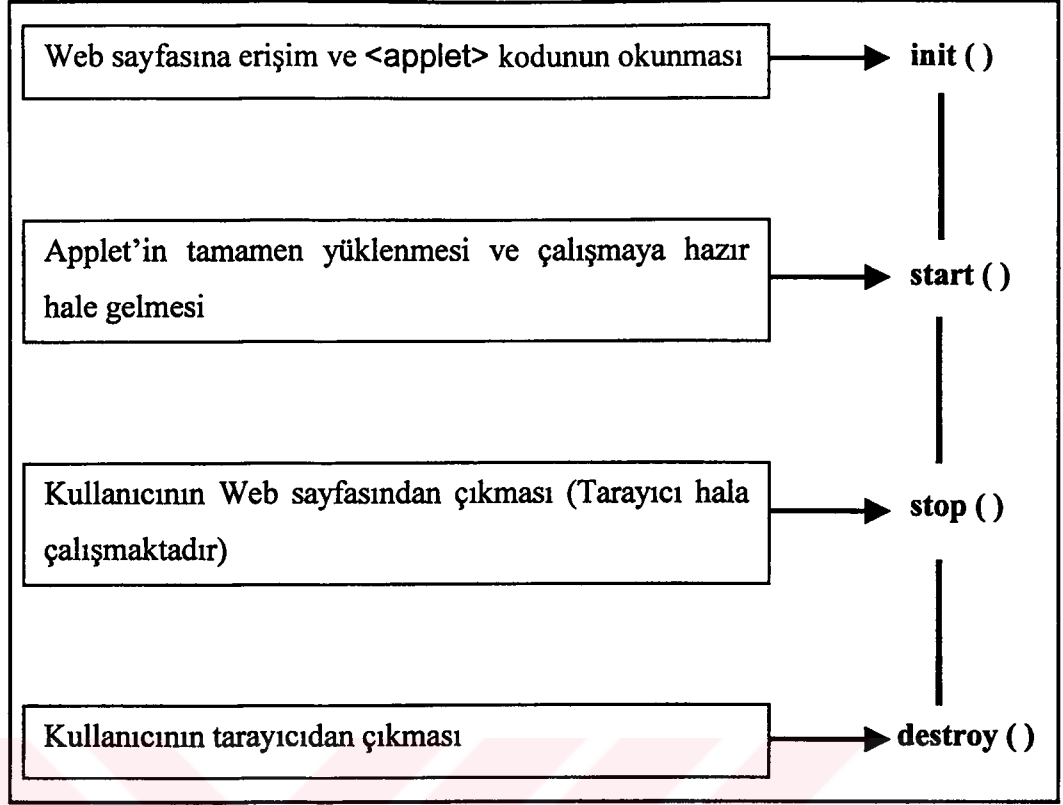
Applet çalışmaya başladığı zaman, gerçekleştirmesi için dizayn edilen görevleri uygulamaya başlar. Buna zıt olarak bir Applet çalışmadığı zaman ise kullanıcı web sayfasına tekrar girene kadar hiçbir iş yapmadan bekler. Applet’in başlama ve bitişini kullanıcının web sayfasına giriş ve çıkışı ile olduğu için çalışma durumu başlama ve bitiş ile kesin bir şekilde ayrılmıştır. Başlama ve bitiş aşamalarının kendine özgü metodları vardır. Applet’in başlama ve bitişini ayrı ayrı kontrol edilebilir.

Appletler bilgisayarın hafızasına yüklendikleri ve CPU zamanını kullandıkları için web tarayıcısından çıkıldığı zaman applet’in hafızada kalması istenmez. Çıkış aşamasında Java Sanal Makinesi applet’in kullandığı tüm kaynakları hafızadan tamamen temizler [5].

Appletler’in çeşitli aşamalardan oluşmasının bazı avantajları vardır. Örneğin bir animasyon programı yapılmaya çalışılıyorsa applet’in çalışmaya başlamasından önce programın kullandığı çok sayıda görüntü nesnesinin yüklenip yüklenmediğinin kontrol edilmesi gerekir. Eğer eksik birtakım yüklemeler oluşursa, yapılan bu program başarısız olur veya bazı çerçeveleri görünmeyebilir. Applet’in oluşum evrelerinin kontrolü oldukça faydalı avantajlar sağlar ve bunun gerçekleşebilmesi için Applet paketinin içerdiği metodlar bulunmaktadır.

Applet Paketi dört çalışma evresi metoduna sahiptir. Bu metodların herbiri ilgili evreyle direkt bağlantılıdır. Şekil 3.2 bu evrelerin birbirleriyle ve applet ile olan ilişkilerini göstermektedir.

Bu metodların her biri applet’in yüklenmesi, çalışması ve sona ermesi aşamaları sırasında otomatik olarak çalışmaktadır. Örneğin bir sayfadan çıkarken ses klibinin bitirilmesi gibi özel durumlarda spesifik olarak birkaç tanesi kullanılabilir. Genellikle de bu metodların hepsi değil, bir veya iki tanesi kullanıcı tarafından kullanılır. Bu metodların hangisine ihtiyaç duyulacağını değerlendirilmesi, applet’in oluşturulması esnasındaki planlamanın bir parçasıdır.



Şekil 3.2 : Applet'in çalışma evreleri ve metodları

- init () metodu

Applet yüklenirken çağırılan ilk metod init () metodudur. Bu metodun gerçekleştirdiği önemli işlerden bazıları görüntü nesnelerinin yüklenmesi veya kullanıcı arayüzünün hazırlanmasıdır. Örnek olarak düğmeye basıldığında ses klibi çalan bir applet düşünelim. Böyle bir uygulamada applet'in çalışmaya başlamasından önce ses klibinin ve düğmenin yüklenmesi gerekir.

- start () metodu

Applet'in tarayıcıya yüklenmesinden ve başlamaya hazır olmasından sonra, start () metodu otomatik olarak çağırılır. Start () metodu genel olarak applet'in yapacağı temel görevleri içerir. Yine ses klibi örneğinde, bu metodun kullanılmasıyla beraber applet çalışmaya başlar.

- stop () metodu

Bu metod start () metoduna çok benzemektedir. Web sayfasını terk ettiğinizde applet'in çalışmayı durdurabilmesi için bu metod otomatik olarak çağırılır. Eğer start() metodu ile başlatılan bazı fonksiyonların kullanıcı müdahalesi olmadan

durdurulması isteniyorsa, stop () metodu kullanılmalıdır. Örneğin ses klibi uygulamasında kullanıcının sayfayı terk ettiği zaman sesin kesilmesi beklenir. Bir döngü içinde çalışan ses klibinin bitirilmesi için bu metod kullanılabilir.

Tüm appletlerde stop () metodunun tekrar tanımlanmasına gerek yoktur. Eğer kullanılan applet basit bir yapıda ise bu metodun çalışan herhangi bir metodu otomatik olarak bitirmesi istenebilir.

- destroy ()

Destroy () metodu deyim yerindeyse, applet'in ölmesi demektir. Tarayıcıdan tamamen çıkıldığında gerekli olabilecek tüm temizlik işlemlerinin yapılabilmesi için bu metod otomatik olarak çağrılır. Adından da anlaşılacağı gibi destroy () metodu applet'in bütün izlerini temizler. Applet'in işgal ettiği tüm hafızayı boşaltır ve çalışan tüm metodları durdurur. Genel olarak, bu metodun kullanımı için özel bir çaba sarf etmeye gerek yoktur; temel bir destroy () metodu otomatik olarak önceden tanımlıdır ve bu metod gerekli tüm işlemleri kendisi yapar.

### **3.3 Temel Jeodezik Hesapların Web'deki Yapısı**

Programların hazırlanmasında Java dilinin Applet teknolojisi kullanılmıştır. Öncelikle internet üzerinden sunumu yapılacak olan jeodezi programlarının hangileri olacağı planlanmıştır. Bu programlar belirlenirken kullanıcıların en çok ihtiyaç duyduğu temel uygulamalara öncelik verilmiştir. Uygulamalar sırasıyla;

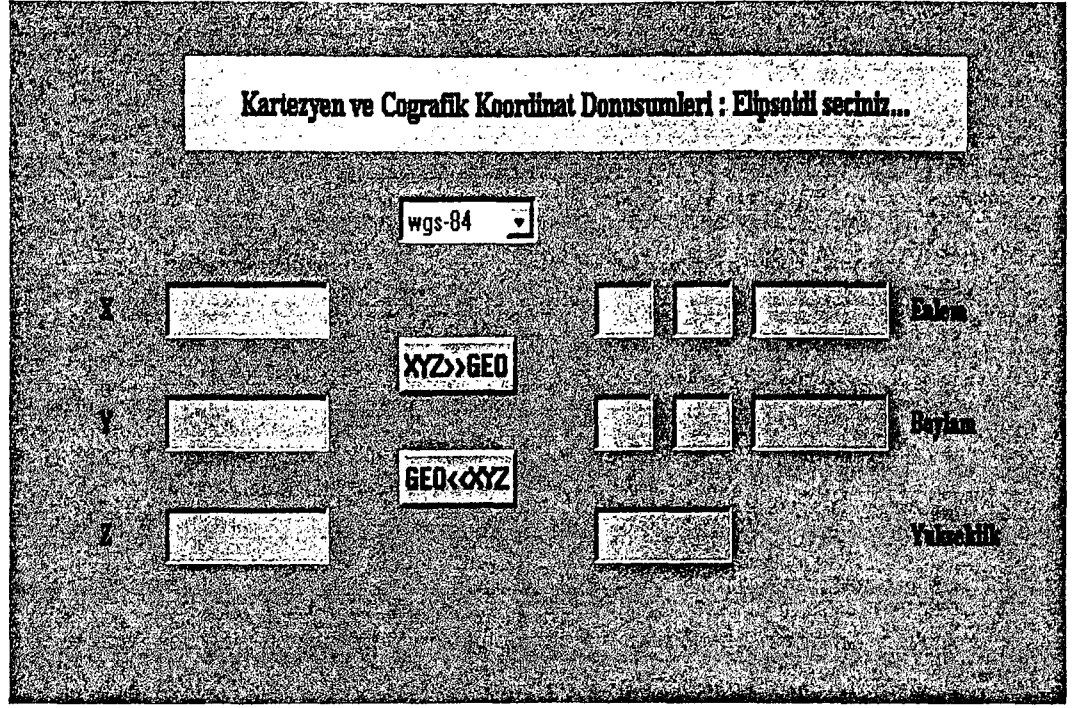
- Julian günü, gps haftası hesabı ,
- Kartezyen-Cografik koordinatlar dönüşümü,
- Cografik-UTM koordinatlar dönüşümü,
- 3 Derecelik sistemde dilimden dilime dönüşüm,
- 6 Derecelik sistemde dilimden dilime dönüşüm,
- İnci temel ödev çözümü,
- İİnci temel ödev çözümü,
- Datum dönüşüm programı

olarak seçilmiştir.

Daha sonraki aşamada seçilen programların formülasyonları ve algoritmaları hazırlanmıştır. Algoritmalar hazırlanırken birden fazla programın kullanabileceği bazı alt programlar ayrıca hazırlanmıştır. Örneğin çoğu uygulamada kullanılan elipsoid seçimi ve derece-radyan dönüşümü gibi programlar bir kere hazırlanıp class dosyası oluşturulduktan sonra ihtiyaç duyulan tüm programların alt programı olarak dahil edilmiştir.

Ana uygulamaların kodları yazıldıktan sonra JDK (Java Development Kit) yazılımı kullanılarak class dosyaları oluşturulmuştur. Bu işlemlerin tamamlanmasıyla ana appletlerin hazırlanmasına geçilmiştir. Appletlerin hazırlanmasında Visual Cafe Standart Edition 4.1 yazılımı kullanılmıştır. Bu yazılımda, her bir uygulama için bir proje oluşturulmuştur. Daha önce hazırlanan uygulamaların class dosyaları projeye dahil edildikten sonra applet'in grafik arayüzü, yazılımın hazır nesnelere kullanılarak tasarlanmıştır. Visual Cafe yazılımını kullanarak oluşturulan arayüzler için kod yazım işlemi tamamen otomatik olarak yazılımın kendisi tarafından yapılmaktadır.

Kullanıcıya düşen görev, hazır bileşenler ile kendisinin hazırladığı programlar arasındaki ilişkiyi düzenleyecek kodları yazmaktır. Bu ilişkiyi daha iyi açıklamak için kartezyen ve coğrafik koordinatlar arasında dönüşüm yapan uygulamayı ele alalım. Şekil 3.3'de bu uygulamayı gerçekleştiren programın arayüzü verilmektedir. Şekilde istenilen elipsoid seçilip ilgili alanlara X, Y ve Z değerleri girilip "XYZ>>GEO" düğmesine basıldığında, öncelikle seçilen elipsoidin parametrelerini almak üzere elipsoid seçim programı çağrılır. Daha sonra X, Y, Z değerleri ve ilgili elipsoidin parametreleri girdi olarak ana programa gönderilir. Ana programda hesaplama yapıldıktan sonra sonuç değerler applet'e gönderilir. Sonuçlar derece-radyan dönüşüm programına girdi olarak gönderilir. Buradan gelen sonuçlar enlem, boylam ve yükseklik hanelerinde görüntülenir. Aynı aşamalar diğer düğme (GEO<<XYZ) için de geçerlidir.



Şekil 3.3 : Kartezyen-Coğrafi Koordinat dönüşümü yapan programın arayüzü

Applet'in arayüzü hazırlanıp diğer programlarla birlikte derlendikten sonra proje dizininde programa ilişkin bütün class dosyaları oluşmuş demektir. Örnek olarak yine yukardaki programın ilgili dosyaları şöyledir :

- ellsec.class
- xyzgeoapplet.class
- xyzgeo.class
- xyzgeoapplet\$SymAction.class
- dms.class
- xyzgeoapplet\$SymFocus.class

Daha önceki bölümlerde belirtildiği gibi Java dilinin Applet uygulaması ancak bir tarayıcı içinde çalışabilmektedir. Bir tarayıcının öncelikle bir dosyayı görüntüleyebilmesi için HTML (Hyper Text Markup Language) dosyasının kodlarının yazılması ve kaydedilmesi gerekir. HTML dosyasından ve dolayısıyla tarayıcıdan applet'in çalıştırılması için gerekli komut <APPLET> komutudur. Yine

aynı örnekten yola çıkarak bu uygulama için oluşturulmuş HTML dosyası şu şekilde kodlanmıştır :

```
<HTML>
<HEAD>
<TITLE>Autogenerated HTML</TITLE>
</HEAD>
<center>
<BODY>
<APPLET CODE="xyzgeoapplet.class" WIDTH=650 HEIGHT=300></APPLET>
</BODY>
</HTML>
```

Burada koyu renli olarak yazılı kısım applet'in çağrıldığı komut satırıdır. Parametre olarak applet'in tarayıcıda çalışacağı pencerenin boyutları da bu satırda belirtilmelidir. Dosya olarak ana applet'in class dosyası olan xyzgeoapplet.class yazılı olmakla beraber aynı dizinde bulunan ve yukarıda verilen tüm class dosyaları çalışma esnasında kullanılmaktadır.

Sonuç olarak elimizde bir HTML dosyası ve derlenmiş program dosyaları bulunmaktadır. HTML dosyası tarayıcıda çalıştırıldığında applet otomatik olarak yüklenir ve çalışmaya hazır halde bekler. Fazla sayıda class dosyasının diğer programların dosyaları ile aynı dizinde bulunması karışıklığa sebep olacağından, java'nın "jar" komutu kullanılarak bir arşiv dosyasına uygulamanın tüm dosyaları yerleştirilmiştir. Yukardaki HTML koduna yapılan

```
<APPLET CODE="xyzgeoapplet.class" archive="xyzgeo.jar" WIDTH=650
HEIGHT=300></APPLET>
```

eklemeyle birlikte, sonuç olarak elimizde her uygulama için bir HTML dosyası ve bir de java arşiv dosyası bulunmaktadır.

Son olarak, yapılan uygulamaların düzenli bir şekilde sunulması için bir "index.html" dosyası hazırlanmış ve kullanıcının istediği programa erişmesi için buradan seçim yapması istenmiştir. Hazırlanan index dosyası Şekil 3.4'de verilmektedir.



Şekil 3.4 : Index.html dosyasının görünümü

### 3.4 Web sayfasının Kullanım Klavuzu

Bu çalışmanın temel amaçlarından biri de uygulamaların kullanımının kolay olmasıdır. Bu sebeple programlar grafik arayüzlü olarak hazırlanmıştır. Bu sayede kullanıcı, her zaman yararlandığı diğer interaktif web hizmetlerinden farklı olmayan bir sayfa ile karşılaşmaktadır. İlk defa uygulamaları kullanan birisi bile, bir iki denemeden sonra uygulamalara kolayca adapte olabilmektedir. Fakat yine de bazı ipuçları vererek hesaplamaların daha kullanışlı bir hale geleceği değerlendirilmektedir.

Genel olarak tüm uygulamalarda açık mavi arka plana sahip olan yerler veri girişi yapılabilecek alanlardır. Beyaz olan kısımlar ise, program çalıştığında yan çıktılarının sunulduğu kısımlardır.

#### - Julian günü hesabı

Bu uygulamada yıl, ay ve gün değişkenleri tamsayı olarak girilmelidir. Eğer ay kısmına 12'den ve gün kısmına 31'den büyük bir değer girilirse program hata verecektir. Zaman kısmına (ut) ise saat ondalıklı veya tamsayı olarak girilebilir. Örneğin ut zamanı 06:30 için bu değeri 6.5 olarak girmek gerekir. Bu hane boş bırakılırsa program  $ut=0.0$  kabul eder ve çalışır. Sol taraftaki diğer hanelerin hiçbirini boş bırakmamak gerekir. Program aynı zamanda GPS haftası ve saniyesi, yılın günü ve haftanın günü değerlerini yan çıktı olarak hesaplamaktadır. Elimizdeki julian gününden yıl, ay ve günü hesaplamak için sağdaki beyaz haneleri doldurmaya gerek yoktur. Sadece julian gününü girip ilgili düğmeye basmak yeterli olacaktır.

#### - Kartezyen-coğrafi koordinatlar dönüşümü hesabı

Kartezyen koordinatlardan coğrafi koordinatlara dönüşüm yapmak isteniyorsa, X, Y ve Z hanelerinin hepsi doldurulup elipsoid seçilmelidir. Elipsoid seçimi yapılmazsa program tercih kutusunda görünen elipsoidi alır ve bu elipsoidin parametrelerine göre çalışır. Eğer coğrafi koordinatlardan kartezyen koordinatlar elde edilmek isteniyorsa, enlem ve boylam değerlerinin dakika ve saniye değerleri boş bırakılabilir. Program bu değerleri sıfır kabul eder.

#### - Coğrafi-UTM koordinatlar dönüşümü hesabı

Coğrafi koordinatlardan düzlem koordinatlara dönüşüm yapmak için, öncelikle ilgili elipsoid seçilir. Daha sonra kaç derecelik sistemde ve hangi dilimde koordinat elde

etmek isteniyorsa, sayfanın sağ üst bölümünden bu parametreler belirlenir. Enlem ve boylam değerlerinin dakika ve saniye haneleri boş bırakılırsa program bu değerleri sıfır kabul eder. Fakat derece haneleri muhakkak tamsayı olarak doldurulmalıdır. Bu hesap neticesinde elde edilen düzlem koordinatlardan sağa değerine 500.000 eklenmiş durumdadır. Aynı zamanda beyaz hanelerde görülen meridyen yakınsama açısı ve haritalama ölçeği, bu programın yan çıktısıdır. Bu hanelere herhangi bir değer girmeye gerek yoktur. Zaten program, bu hanelere herhangi bir veri girişine müsaade etmemektedir.

#### - Dilim dönüşümleri

Bu hesaplamaları yapmak için, koordinatların ait olduğu dilim orta meridyenini soldaki kutudan seçerek sağa ve yukarı değerlerini soldaki hanelere girmek gerekir. Hesaplanmasını istediğimiz komşu dilim meridyeni sağdaki seçenek kutusundan seçilir. Hesaplama butonuna basıldığında sonuçlar ekrana gelir. 3 ve 6 derecelik sistemlerde dilimden dilime dönüşüm uygulamalarının her ikisi için de yukardaki açıklamalar geçerlidir.

#### - I. Temel ödev çözümü

Bu uygulamada, Vincenty formüllerinden yararlanılmıştır. Elipsoid seçiminden sonra, soldaki tüm haneler doldurulmalıdır. Yalnız, enlem, boylam ve azimut değerlerinin dakika ve saniye haneleri seçime bağlı olarak, eğer sıfır girilmek isteniyorsa boş bırakılabilir. Hesaplama düğmesine basıldığında sağdaki hanelerde sonuç değerleri görülür. Bu hanelerin en altında bulunan elipsoidal mesafe kısmı fazlalık olarak görüntülenmektedir. Bunun sebebi, ikinci nokta hesaplandıktan sonra üçüncü bir nokta hesaplanmak isterse, tekrar soldaki haneleri doldurmak yerine, sağdaki hanelerde değişiklikleri yaparak kullanıcıya kolaylık sağlamaktır.

#### - II. Temel ödev çözümü

Bu hesaplamalar yapılırken de yukarıda olduğu gibi Vincenty formüllerinden yararlanılmıştır. Soldaki mavi hanelere birinci ve ikinci noktaların enlem ve boylam değerleri girilir. Dakika ve saniye kutularına değer girilmezse program sıfır kabul eder. Sağdaki beyaz hanelerde sadece sonuçlar sergilenir. Herhangi bir değer girilmesi gerekmemektedir.

- Datum dönüşümü hesabı

Bu programda, WGS-84 elipsoidal koordinatları ile ED-50 datumundaki koordinatlar arasında üç boyutlu dönüşüm yapılmaktadır. Programda kullanılan dönüşüm parametreleri Harita Genel Komutanlığı tarafından Türkiye için hesaplanmış parametrelerdir [10]. Hesaplamanın yapılabilmesi için ilgili datumdaki koordinatlara ilişkin tüm bilgilerin girilmesi gerekir. Diğer uygulamalarda olduğu gibi dakika ve saniye haneleri seçime bağlı olup boş bırakıldığında bu değerler sıfır olarak kabul edilir. Yükseklik değerlerinin birimi metredir.



## 4. TEMEL JEODEZİK HESAPLAR

### 4.1 I. Temel Ödev Çözümü

Bu uygulamada, istenilen bir elipsoid üzerinde, coğrafi koordinatları mevcut bir noktadan azimut ve elipsoidal mesafe girilerek diğer bir noktanın koordinatlarını ve ters azimutunu hesaplayan bir program yapılmıştır.

İnci temel ödev olarak da bilinen bu uygulama için çok çeşitli metodlar bulunmaktadır. Vincenty yöntemi de bu metodlardan birisidir ve 1-2 cm.'den 20.000 km.'ye kadar değişen bir mesafe aralığında milimetre doğruluğunda sonuç verdiği için bu uygulamada Vincenty yöntemi kullanılmıştır.

I. Temel Ödev Çözümü (Vincenty Yöntemi)

Elipsoidi Seçiniz...

Elipsoid

wgs-84

	İnci Nokta				2.nci Nokta			
Doğum	39	53	18.0549		42	47	55.433662	Doğum
Boylam	32	45	31.7110	1 >> 2	60	46	50.856024	Boylam
Azım (1-2)	72	56	45.17825	1 << 2	271	40	22.259488	Azım (2-1)
Elipsoidal Mesafe (s)	2356456.25				2356456.25			Elipsoidal Mesafe (s)

Şekil 4.1 : I. Temel ödev çözümü yapan uygulama

Şekil 4.1’de görüldüğü gibi öncelikle ilgili elipsoid seçimi yapılır. Daha sonra, birinci noktaya ait enlem ve boylam değerleri ile birlikte, azimut ve elipsoidal mesafe girilir.  $1 \gg 2$  düğmesine basıldığı zaman ikinci noktaya ait enlem, boylam ve ters azimut değerleri elde edilir.

Programda kullanılan Vincenty yöntemine ait formülasyon ve sembollere ilişkin açıklamalar aşağıdaki gibidir [6]:

$$\tan U_1 = (1-f) \tan \phi_1 \quad (4.1)$$

$$\tan \sigma_1 = \tan U_1 / \cos \alpha_{12} \quad (4.2)$$

$$\sin \alpha = \cos U_1 \sin \alpha_{12} \quad (4.3)$$

$$u^2 = \cos^2 \alpha (a^2 - b^2) / b^2 \quad (4.4)$$

$$A = 1 + (u^2/16384) \{4096 + u^2[-768 + u^2(320 - 175u^2)]\} \quad (4.5)$$

$$B = (u^2/1024) \{256 + u^2[-128 + u^2(74 - 47u^2)]\} \quad (4.6)$$

$$\sigma = (s/bA) \quad (4.7)$$

yaklaşımıyla;

aşağıdaki 3 eşitlik,  $\sigma$  değerinin değişimindeki fark önemsiz bir hale gelene kadar iterasyona sokulur.

$$2\sigma_m = 2\sigma_1 + \sigma \Delta\sigma = B \sin \sigma \{ \cos 2\sigma_m + (B/4) [\cos \sigma (-1 + 2\cos^2 2\sigma_m) - (B/6) \cos 2\sigma_m (-3 + 4\sin^2 \sigma) (-3 + 4\cos^2 2\sigma_m)] \} \sigma = (s/bA) + \Delta\sigma \quad (4.8)$$

Daha sonra;

$$\tan \phi_2 = (\sin U_1 \cos \sigma + \cos U_1 \sin \sigma \cos \alpha_{12}) / \{ (1-f) [\sin^2 \alpha + (\sin U_1 \sin \sigma - \cos U_1 \cos \sigma \cos \alpha_{12})^2]^{1/2} \} \quad (4.9)$$

$$\tan \lambda = (\sin \sigma \sin \alpha_{12}) / (\cos U_1 \cos \sigma - \sin U_1 \sin \sigma \cos \alpha_{12}) \quad (4.10)$$

$$C = (f/16) \cos^2 \alpha [4 + f(4 - 3\cos^2 \alpha)] \quad (4.11)$$

$$\omega = \lambda - (1-C) f \sin \alpha \{ \sigma + C \sin \sigma [\cos 2\sigma_m + C \cos \sigma (-1 + 2\cos^2 2\sigma_m)] \} \quad (4.12)$$

$$\lambda_2 = \lambda_1 + \omega \quad (4.13)$$

$$\tan \alpha_{21} = (\sin \alpha) / (-\sin U_1 \sin \sigma + \cos U_1 \cos \sigma \cos \alpha_{12}) \quad (4.14)$$

eşitlikleri ile sonuç elde edilir.

- $\phi_1$  : 1. Noktanın enlemi  
 $\lambda_1$  : 1. Noktanın boylamı  
 $\alpha_{12}$  : 1. Noktadan 2. noktaya azimut  
s : Noktalar arasındaki elipsoidal mesafe  
f : Elipsoidin basıklığı  
a : Elipsoidin büyük yarı ekseni  
b : Elipsoidin küçük yarı ekseni  
 $\phi_2$  : 2. Noktanın enlemi  
 $\lambda_2$  : 2. Noktanın boylamı  
 $\alpha_{21}$  : 2. Noktadan 1. noktaya azimut

#### 4.2 II. Temel Ödev Çözümü

Bu uygulamada, istenilen bir elipsoid üzerinde, coğrafi koordinatları mevcut iki noktanın koordinatlarından hareketle noktaların birbirine göre olan azimutları ve aralarındaki elipsoidal mesafeyi hesaplayan bir program yapılmıştır.

İlinci temel ödev olarak da bilinen bu uygulama için çok çeşitli metodlar bulunmaktadır. Vincenty yöntemi de bu metodlardan birisidir ve 1-2 cm.'den 20.000 km.'ye kadar değişen bir mesafe aralığında milimetre doğruluğunda sonuç verdiği için bu uygulamada Vincenty yöntemi kullanılmıştır.

Şekil 4.2'de görüldüğü gibi öncelikle ilgili elipsoid seçimi yapılır. Daha sonra, birinci ve ikinci noktanın coğrafi koordinatları girilir. HESAPLA düğmesine basıldığı zaman her iki noktanın birbirine göre olan azimut değerleri ve aralarındaki elipsoidal mesafe elde edilir.



$$C = (f/16) \cos^2\alpha [4+f(4-3\cos^2\alpha)] \quad (4.23)$$

$$\lambda = \omega + (1-C)f \sin\alpha \{ \sigma + C \sin\sigma [\cos 2\sigma_m + C \cos\sigma (-1 + 2\cos^2 2\sigma_m)] \} \quad (4.24)$$

Daha sonra;

$$u^2 = \cos^2\alpha (a^2 - b^2) / b^2 \quad (4.25)$$

$$A = 1 + (u^2/16384) \{ 4096 + u^2[-768 + u^2(320-175u^2)] \} \quad (4.26a)$$

$$B = (u^2/1024) \{ 256 + u^2[-128 + u^2(74-47u^2)] \} \quad (4.26b)$$

$$\Delta\sigma = B \sin\sigma \{ \cos 2\sigma_m + (B/4) [\cos\sigma (-1 + 2\cos^2 2\sigma_m) - (B/6)\cos 2\sigma_m (-3 + 4\sin^2\sigma) (-3 + 4\cos^2 2\sigma_m)] \} \quad (4.27)$$

$$s = bA(\sigma - \Delta\sigma) \quad (4.28)$$

$$\tan\alpha_{12} = (\cos U_2 \sin\lambda) / (\cos U_1 \sin U_2 - \sin U_1 \cos U_2 \cos\lambda) \quad (4.29)$$

$$\tan\alpha_{21} = (\cos U_1 \sin\lambda) / (-\sin U_1 \cos U_2 + \cos U_1 \sin U_2 \cos\lambda) \quad (4.30)$$

elde edilir.

$\phi_1$  : 1. Noktanın enlemi

$\phi_2$  : 2. Noktanın enlemi

$\lambda_1$  : 1. Noktanın boylamı

$\lambda_2$  : 2. Noktanın boylamı

$f$  : Elipsoidin basıklığı

$a$  : Elipsoidin büyük yarı eksen

$b$  : Elipsoidin küçük yarı eksen

$\alpha_{12}$  : 1. Noktadan 2. noktaya azimut

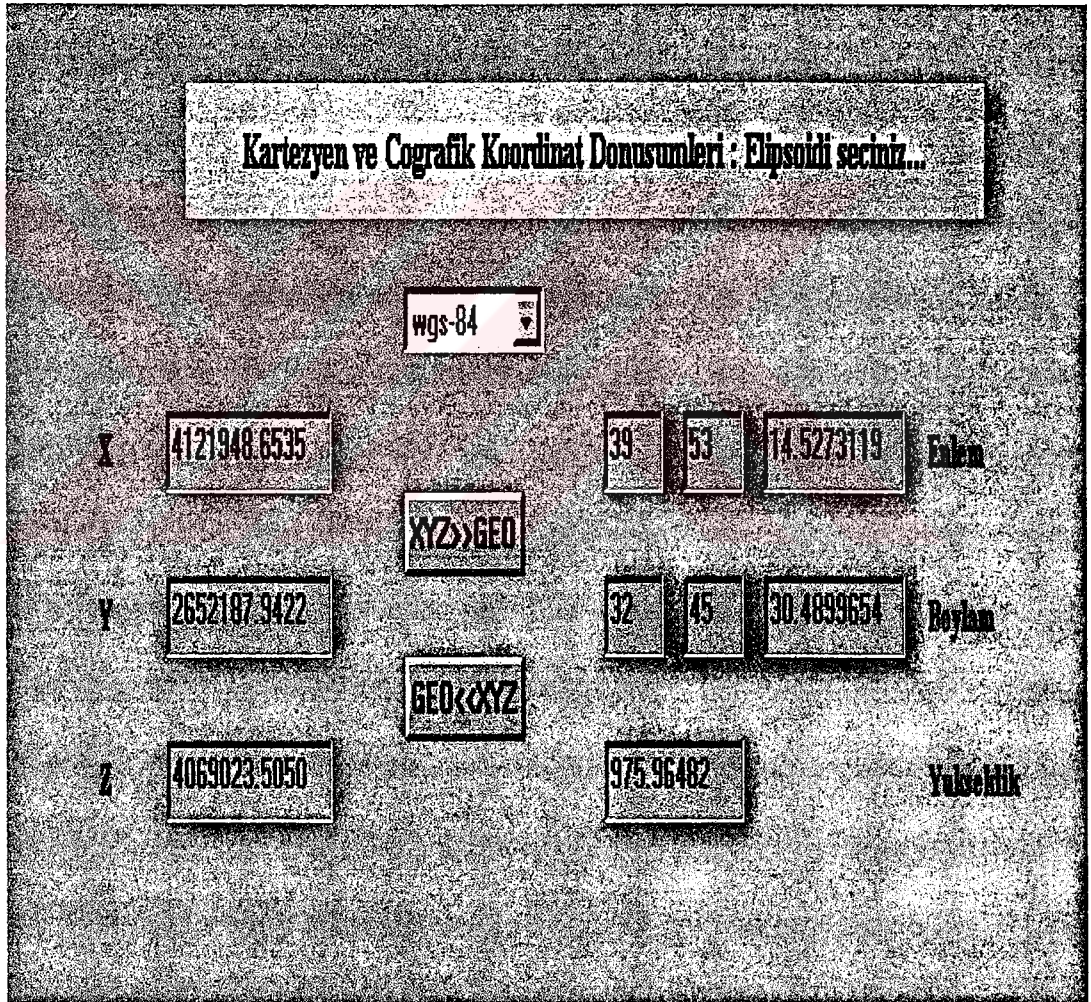
$\alpha_{21}$  : 2. Noktadan 1. noktaya azimut

$s$  : Noktalar arasındaki elipsoidal mesafe

### 4.3 Kartezyen-Coğrafik Koordinatlar Dönüşümü ve Tersİ

Bu uygulamada, herhangi bir elipsoid üzerinde, verilen kartezyen koordinatlardan coğrafi koordinatları ve tersi olarak da verilen coğrafi koordinatlardan kartezyen koordinatları hesaplayan bir program yapılmıştır.

Şekil 4.3’de görüldüğü gibi, öncelikle elipsoidin belirlenmesi gerekmektedir. Daha sonra ilgili hanelere X, Y, Z dik koordinatları girilerek XYZ>>GEO düğmesine basılarak noktaya ait enlem, boylam ve yükseklik değerleri elde edilebilir. Aynı şekilde enlem, boylam ve yükseklik haneleri doldurulup GEO<<XYZ düğmesine basılırsa bu sefer de noktanın jeodezik dik koordinatları elde edilir.



Şekil 4.3 : Kartezyen-Coğrafi koordinatlar dönüşümü yapan program

Kartezyen koordinatlardan coğrafi koordinatların hesabı  $\phi$  ve  $h$ 'nin iterasyonunu gerektirir [7]. Boylam değeri ;

$$\lambda = \arctan ( Y / X) \quad (4.31a)$$

eşitliğinden kolayca bulunur.  $h \ll N$  olup , ilk aşama için  $h = 0$  alınırsa;

$$N = \frac{a}{\sqrt{1-f(2-f)\sin^2\phi}} \quad (4.31b)$$

$$\phi = \arctan\left(\frac{Z}{\sqrt{X^2 + Y^2}}\left(1 - \frac{(2-f)fN}{N+h}\right)^{-1}\right) \quad (4.32)$$

$$h = \frac{\sqrt{X^2 + Y^2}}{\cos\phi} - N \quad (4.33)$$

$\phi$  ve  $h$ 'nin iterasyonu ile coğrafi koordinatlar elde edilir.

Coğrafi koordinatlardan kartezyen koordinatların hesabı :

$$X = N \cos\phi \cos\lambda \quad (4.34)$$

$$Y = N \cos\phi \sin\lambda \quad (4.35)$$

$$Z = (1 - f)^2 N \sin\phi \quad (4.36)$$

Burada  $N$  değeri yine yukardaki gibi ;

$$N = \frac{a}{\sqrt{1-f(2-f)\sin^2\phi}} \quad (4.37)$$

ile hesaplanabilir. Sonuç formülasyon :

$$X = (N+h) \cos\phi \cos\lambda \quad (4.38)$$

$$Y = (N+h) \cos\phi \sin\lambda \quad (4.39)$$

$$Z = ((1 - f)^2 N + h) \sin\phi \quad (4.40)$$

- a : Elipsoidin büyük yarı eksen  
f : Elipsoidin basıklığı  
N : Meridyen normalinin eğrilik yarıçapı  
X, Y, Z: Noktanın kartezyen koordinat bileşenleri  
 $\phi$  : Noktanın enlemi  
 $\lambda$  : Noktanın boylamı  
h : Noktanın elipsoit yüksekliği

#### 4.4 Coğrafi – UTM Koordinatlar Dönüşümü ve Ters

Bu uygulamada, herhangi bir elipsoid üzerinde, verilen coğrafi koordinatlardan düzlem koordinatları ve tersi olarak da verilen düzlem koordinatlardan coğrafi koordinatları hesaplayan bir program yapılmıştır.

Şekil 4.4’de görüldüğü gibi, öncelikle elipsoidin, dilim genişliğinin ve dilim orta meridyeninin belirlenmesi gerekmektedir. Daha sonra ilgili hanelere enlem boylam girilerek GEO>>UTM düğmesine basılarak noktaya ait düzlem koordinatları elde edilebilir. Aynı şekilde sağa ve yukarı haneleri doldurulup GEO<<UTM düğmesine basılırsa bu sefer de noktanın coğrafi koordinatları elde edilir. Program yan çıktı olarak meridyen yakınsama açısı ve haritalama ölçeğini de hesaplamaktadır.

Bu uygulamada aynı zamanda 3 derecelik ve 6 derecelik sistemler arasında dönüşüm de yapılabilmektedir. Örneğin elimizde 3 derecelik sistemde sağa yukarı koordinatları olan bir nokta olsun. İlk yapmamız gereken bu değerleri sağa yukarı hanelerine yazıp dilim genişliği ve dilim orta meridyenini seçtikten sonra düğmeye basarak coğrafi koordinatlarını hesaplamaktır. Daha sonra, bu koordinatlara hiç dokunmadan 6 derecelik dilim sistemini işaretleyip dilim orta meridyenini seçerek düğmeye basarsak, aynı noktanın 6 derecelik sistemde düzlem koordinatlarını elde etmiş oluruz.

**COGRAFI-UTM Koordinatlar arasında donusum:**  
**Elipsoid, Dilim Genisligi ve Dilim Orta Meridyenini seciniz...**

Elipsoid  
ed-50

3 Der. D.O.M 30  
6 Der. D.O.M 33

Enlem 39 53 18.0549 GEO>>UTM 479377.5593 Sağ

Boylam 32 45 31.7118 GEO<<UTM 4415468.5055 Yukarı

Meridyen Yakınsama Açısı 0 9 17.0495

Haritalama Ölçeği 0.999605235

Şekil 4.4 : Coğrafik – UTM Koordinatlar dönüşümü yapan program

Coğrafi ve grid dönüşümlerini yapmak için gerekli formülasyonlar aşağıda verilmiştir [6,8].

Öncelikli hesaplamalar:

Meridyen mesafesi hesabı

$$m = a(1-e^2) \int_{\phi_1}^{\phi_2} [1-(e^2 \sin^2 \phi)]^{-3/2} d\phi \quad (4.41)$$

formülüyle yapılabileceği gibi, seri açılım kullanmak daha etkili olacaktır.

$$m = a\{A_0\phi - A_2 \sin 2\phi + A_4 \sin 4\phi - A_6 \sin 6\phi\} \quad (4.42)$$

$$A_0 = 1 - (e^2/4) - (3e^4/64) - (5e^6/256) \quad (4.43)$$

$$A_2 = (3/8)(e^2 + e^4/4 + 15e^6/128) \quad (4.44)$$

$$A_4 = (15/256)(e^4 + 3e^6/4) \quad (4.45)$$

$$A_6 = 35e^6/3072 \quad (4.46)$$

Ayak noktası enlemi ( $\phi'$ ):

$$n = (a-b)/(a+b) = f/(2-f) \quad (4.47)$$

$$G = a(1-n)(1-n^2)(1+(9/4)n^2+(225/64)n^4)(\pi/180) \quad (4.48)$$

$$\sigma = (m\pi)/(180G) \quad (4.49)$$

$$\begin{aligned} \phi' = & \sigma + ((3n/2) - (27n^3/32))\text{Sin}2\sigma + ((21n^2/16) - (55n^4/32))\text{Sin}4\sigma + (151n^3/96)\text{Sin}6\sigma \\ & + (1097n^4/512)\text{Sin}8\sigma \end{aligned} \quad (4.50)$$

Eğrilik yarıçapı :

$$\rho = a(1-e^2) / (1-e^2\text{Sin}^2\phi)^{3/2} \quad (4.51)$$

$$v = a / (1-e^2\text{Sin}^2\phi)^{1/2} \quad (4.52)$$

$$\psi = v / \rho \quad (4.53)$$

Coğrafi koordinatlardan düzlem koordinatlarının hesabı :

$$t = \text{Tan } \phi \quad (4.54)$$

$$\omega = \lambda - \lambda_0 \quad (4.55)$$

$$E' = (K_0 v \omega \text{Cos} \phi) \{ 1 + \text{Term1} + \text{Term2} + \text{Term3} \} \quad (4.56)$$

$$\text{Term1} = (\omega^2/6)\text{Cos}^2\phi (\psi - t^2) \quad (4.57)$$

$$\text{Term2} = (\omega^4/120)\text{Cos}^4\phi [4\psi^3(1-6t^2) + \psi^2(1+8t^2) - \psi 2t^2 + t^4] \quad (4.58)$$

$$\text{Term3} = (\omega^6/5040)\text{Cos}^6\phi (61-479t^2+179t^4-t^6) \quad (4.59)$$

$$E = E' + 500\,000 \quad (4.60)$$

$$N' = K_0 \{ m + \text{Term1} + \text{Term2} + \text{Term3} + \text{Term4} \} \quad (4.61)$$

$$\text{Term1} = (\omega^2/2) v \text{Sin} \phi \text{Cos} \phi \quad (4.62)$$

$$\text{Term2} = (\omega^4/24) v \text{Sin} \phi \text{Cos}^3\phi (4\psi^2 + \psi - t^2) \quad (4.63)$$

$$\begin{aligned} \text{Term3} = & (\omega^6/720) v \text{Sin} \phi \text{Cos}^5\phi [8\psi^4(11-24t^2) - 28\psi^3(1-6t^2) \\ & + \psi^2(1-32t^2) - \psi(2t^2) + t^4] \end{aligned} \quad (4.64)$$

$$\text{Term4} = (\omega^8/40320) v \text{Sin} \phi \text{Cos}^7\phi (1385-3111t^2+543t^4-t^6) \quad (4.65)$$

$$N = N' + 0 \quad (4.66)$$

Meridyen yakınsama açısı :

$$\gamma = \text{Term1} + \text{Term2} + \text{Term3} + \text{Term4} \quad (4.67)$$

$$\text{Term1} = -\omega \sin \phi \quad (4.68)$$

$$\text{Term2} = -(\omega^3/3) \sin \phi \cos^2 \phi (2 \psi^2 - \psi) \quad (4.69)$$

$$\text{Term3} = -(\omega^5/15) \sin \phi \cos^4 \phi [\psi^4(11-24t^2) - \psi^3(11-36t^2) + 2\psi^2(1-7t^2) + \psi t^2] \quad (4.70)$$

$$\text{Term4} = -(\omega^7/315) \sin \phi \cos^6 \phi (17-26t^2+2t^4) \quad (4.71)$$

Haritalama ölçeği :

$$k = K_0 + K_0 \text{Term1} + K_0 \text{Term2} + K_0 \text{Term3} \quad (4.72)$$

$$\text{Term1} = (\omega^2/2) \psi \cos^2 \phi \quad (4.73)$$

$$\text{Term2} = (\omega^4/24) \cos^4 \phi [4\psi^3(1-6t^2) + \psi^2(1+24t^2) - 4\psi t^2] \quad (4.74)$$

$$\text{Term3} = (\omega^6/720) \cos^6 \phi (61-148t^2+16t^4) \quad (4.75)$$

Düzlem koordinatlardan coğrafi koordinatların hesabı :

$$E' = E - 500\,000 \quad (4.76)$$

$$x = E' / (K_0 \nu') \quad (4.77)$$

$$\phi = \phi' - \text{Term1} + \text{Term2} - \text{Term3} + \text{Term4} \quad (4.78)$$

$$\text{Term1} = (t' / (K_0 \rho')) (xE' / 2) \quad (4.79)$$

$$\text{Term2} = (t' / (K_0 \rho')) (E'x^3 / 24) [-4\psi'^2 + 9\psi' (1-t'^2) + 12t'^2] \quad (4.80)$$

$$\text{Term3} = (t' / (K_0 \rho')) (E'x^5 / 720) [8\psi'^4 (11-24t'^2) - 12\psi'^3 (21-71t'^2) + 15\psi'^2 (15-98t'^2 + 15t'^4) + 180\psi' (5t'^2 - 3t'^4) + 360t'^4] \quad (4.81)$$

$$\text{Term4} = (t' / (K_0 \rho')) (E'x^7 / 40320) (1385 + 3633t'^2 + 4095t'^4 + 1575t'^6) \quad (4.82)$$

$$\omega = \text{Term1} - \text{Term2} + \text{Term3} - \text{Term4} \quad (4.83)$$

$$\text{Term1} = x \sec \phi' \quad (4.84)$$

$$\text{Term2} = (x^3/6) \sec \phi' (\psi' + 2t'^2) \quad (4.85)$$

$$\text{Term3} = (x^5/120) \sec \phi' [-4\psi'^3(1-6t'^2) + \psi'^2(9-68t'^2) + 72\psi't'^2 + 24t'^4] \quad (4.86)$$

$$\text{Term4} = (x^7/5040) \sec \phi' (61 + 662t'^2 + 1320t'^4 + 720t'^6) \quad (4.87)$$

$$\lambda = \lambda_0 + \omega \quad (4.88)$$

Meridyen yakınsama açısı :

$$x = E'/K_0 v' \quad (4.89)$$

$$t' = \text{Tan}\phi' \quad (4.90)$$

$$\gamma = \text{Term1} + \text{Term2} + \text{Term3} + \text{Term4} \quad (4.91)$$

$$\text{Term1} = -t' x \quad (4.92)$$

$$\text{Term2} = (t' x^3/3) (-2\psi'^2 + 3\psi' + t'^2) \quad (4.93)$$

$$\text{Term3} = (-t' x^5/15) [\psi'^4 (11 - 24t'^2) - 3\psi'^3 (8 - 23t'^2) + 5\psi'^2 (3 - 14t'^2) + 30\psi't'^2 + 3t'^4] \quad (4.94)$$

$$\text{Term4} = (t' x^7/315) (17 + 77t'^2 + 105t'^4 + 45t'^6) \quad (4.95)$$

$\phi$  : Enlem

$\lambda$  : Boylam

a : Elipsoidin büyük yarı ekseni

b : Elipsoidin küçük yarı ekseni

f : Elipsoidin basıklığı

e : Elipsoidin birinci dış merkezliliği

E : Sağa değer

N : Yukarı değer

$\gamma$  : Meridyen yakınsama açısı

k : Haritalama ölçeği

$K_0$  : Küçültme faktörü

## 4.5 Dilimden Dilime Dönüşümler

### 4.5.1 Üç Derecelik Sistemde Dilimden Dilime Dönüşüm

Bu uygulamada, herhangi bir elipsoid üzerinde, 3 derecelik sistemde düzlem koordinatları olan bir noktanın komşu dilimdeki koordinatlarını hesaplayan bir program yapılmıştır.

Şekil 4.5’de görüldüğü gibi öncelikle ilgili elipsoid ve düzlem koordinatların ait olduğu dilimin orta meridyeni seçilir. Sağ tarafta bulunan 2. DOM seçeneğinden komşu dilimin orta meridyeni belirlenir. Daha sonra, noktanın sağa ve yukarı değerleri soldaki hanelere yazılıp HESAPLA düğmesine basılırsa, noktanın komşu dilimdeki düzlem koordinatları elde edilir.

3 Derecelik Sistemde Dilimden Dilime Donusum Hesabi  
Elipsoid. 1nci ve 2nci Sistemlerin D. O. M.'ni seciniz...

1. DOM  
33

Elipsoid  
wgs-84

2. DOM  
36

saga 578965.56

yukari 4256568.63

HESAPLA

saga 317035.086

yukari 4258261.7288

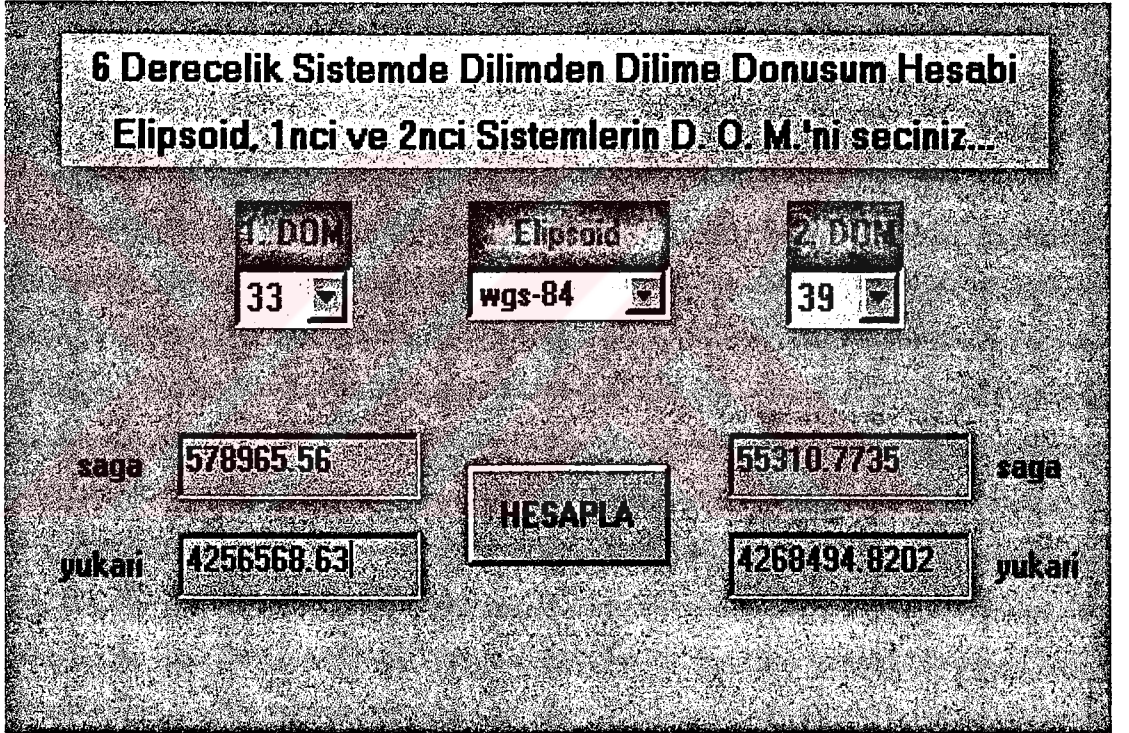
Şekil 4.5 : 3 Derecelik sistemde dilimden dilime dönüşüm yapan program

Bu programla ilgili olarak özel bir formül kodlaması yapılmamıştır. Daha önce elde mevcut olan düzlem koordinatlar ve coğrafi koordinatlar arasında dönüşüm yapan programın class dosyaları kullanılmıştır. Aslında bu program, öncelikle noktanın düzlem koordinatlarından, tek anlamlı bir değer olan coğrafi koordinatlarını elde etmektedir. Daha sonra bu değeri istenen dilimdeki sağa ve yukarı değerlerine dönüştürmektedir.

#### 4.5.2 Altı Derecelik Sistemde Dilimden Dilime Dönüşüm

Bu uygulamada, herhangi bir elipsoid üzerinde, 6 derecelik sistemde düzlem koordinatları olan bir noktanın komşu dilimdeki koordinatlarını hesaplayan bir program yapılmıştır.

Şekil 4.6'da görüldüğü gibi öncelikle ilgili elipsoid ve düzlem koordinatların ait olduğu dilimin orta meridyeni seçilir. Sağ tarafta bulunan 2. DOM seçeneğinden komşu dilimin orta meridyeni belirlenir. Daha sonra, noktanın sağa ve yukarı değerleri soldaki hanelere yazılıp HESAPLA düğmesine basılırsa, noktanın komşu dilimdeki düzlem koordinatları elde edilir.



Şekil 4.6 : 6 Derecelik sistemde dilimden dilime dönüşüm yapan program

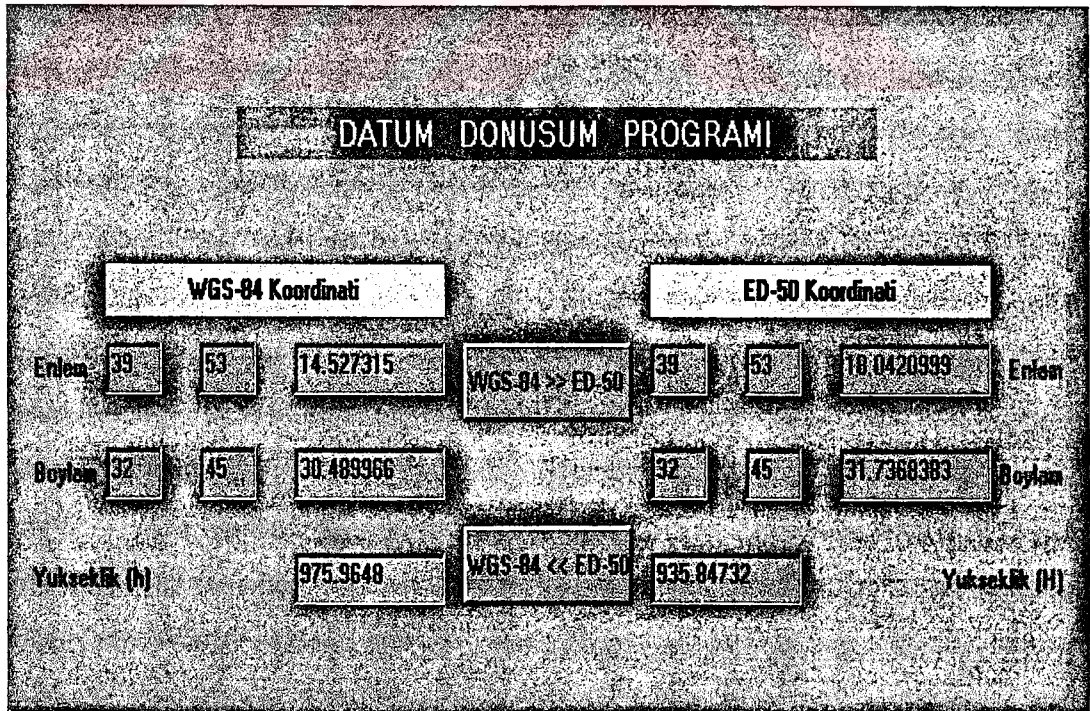
Üç derecelik sistemde dilimden dilime dönüşüm yapan programda olduğu gibi, bu program için de özel bir formül kodlaması yapılmamıştır. Daha önce elde mevcut olan düzlem koordinatlar ve coğrafi koordinatlar arasında dönüşüm yapan programın class dosyaları kullanılmıştır. Aslında bu program, öncelikle noktanın düzlem koordinatlarından, tek anlamlı bir değer olan coğrafi koordinatlarını elde etmektedir. Daha sonra bu değeri istenen dilimdeki sağa ve yukarı değerlerine dönüştürmektedir.

#### 4.6 Datum Dönüşümü Hesabı

Bu uygulamada, ED-50 ve WGS-84 datumları arasında üç boyutlu dönüşüm yapan bir program geliştirilmiştir. Dönüşüm parametreleri olarak Harita Genel Komutanlığı'nın belirlediği parametreler kullanılmıştır [10]. Bu parametrelerle yaklaşık 1 metre doğruluğunda sonuç alınabilmektedir.

Şekil 4.7'de görüldüğü gibi, eğer elimizde WGS-84 koordinatı bulunan bir nokta varsa, bu koordinatları WGS-84 koordinatları bölümüne yazıp WGS84>>ED50 düğmesine basarak noktanın ED-50 datumunda koordinatlarını elde ederiz. ED-50 datumunda elde ettiğimiz yükseklik ortometrik yüksekliktir. Aynı şekilde, ED-50 koordinatı ve ortometrik yüksekliği olan bir noktanın WGS-84 koordinatlarını elde etmek için WGS-84<<ED-50 düğmesine basmak yeterli olacaktır.

Bu programda, dönüşümün yapılabilmesi için kartezyen koordinata ihtiyaç duyulduğu için, daha önceden hazırlanmış olan kartezyen-coğrafi koordinat dönüşümü programının class dosyaları kullanılarak girilen coğrafi koordinat önce kartezyene çevrilmektedir. Hesaplama yapıldıktan sonra sonuç kartezyen koordinat coğrafi koordinata çevrilerek sergilenmektedir.



Şekil 4.7 : Datum dönüşüm programı

Datum dönüşümü ile ilgili temel formülasyon aşağıdaki gibidir [7]:

$$\begin{bmatrix} X_{ED50} \\ Y_{ED50} \\ Z_{ED50} \end{bmatrix} = \begin{bmatrix} 1 & \varepsilon_Z & -\varepsilon_Y \\ -\varepsilon_Z & 1 & \varepsilon_X \\ \varepsilon_Y & -\varepsilon_X & 1 \end{bmatrix} \begin{bmatrix} X_{WGS84} \\ Y_{WGS84} \\ Z_{WGS84} \end{bmatrix} + k \begin{bmatrix} X_{WGS84} \\ Y_{WGS84} \\ Z_{WGS84} \end{bmatrix} + \begin{bmatrix} t_X \\ t_Y \\ t_Z \end{bmatrix} \quad (4.96)$$

Yukardaki formülde, Harita Genel Komutanlığı tarafından belirlenen dönüşüm parametreleri aşağıdaki gibidir [10]:

WGS-84'den ED-50'ye dönüşüm parametreleri :

X eksenini etrafındaki dönüklük	$\varepsilon_X = 0.0183''$
Y eksenini etrafındaki dönüklük	$\varepsilon_Y = -0.0003''$
Z eksenini etrafındaki dönüklük	$\varepsilon_Z = 0.4738''$
X eksenini yönündeki öteleme	$t_X = 84.003 \text{ m}$
Y eksenini yönündeki öteleme	$t_Y = 102.315 \text{ m}$
Z eksenini yönündeki öteleme	$t_Z = 129.879 \text{ m}$
Ölçek	$k = -1.0347$

ED-50'den WGS-84'e dönüşüm parametreleri :

X eksenini etrafındaki dönüklük	$\varepsilon_X = -0.0183''$
Y eksenini etrafındaki dönüklük	$\varepsilon_Y = 0.0003''$
Z eksenini etrafındaki dönüklük	$\varepsilon_Z = -0.4738''$
X eksenini yönündeki öteleme	$t_X = -84.003 \text{ m}$
Y eksenini yönündeki öteleme	$t_Y = -102.315 \text{ m}$
Z eksenini yönündeki öteleme	$t_Z = -129.879 \text{ m}$
Ölçek	$k = 1.0347$

#### 4.7 Julian Günü, GPS Haftası Hesabı

Bu uygulamada, istenilen bir tarihe tekabül eden julian gününün hesaplanması ve yine girilen bir julian günden ilgili tarihi hesaplayan bir program yapılmıştır. Aynı zamanda program yan çıktı olarak ilgili tarihe rastlayan GPS haftasını, yılın gününü ve haftanın gününü hesaplamaktadır.

Şekil 4.8’de görüldüğü gibi, julian gününü elde etmek için en az yıl, ay ve gün hanelerinin doldurulması gerekir. UT alanı seçime bağlı olup, boş bırakıldığında UT=0.0 zamanını almaktadır.

Julian günü ve GPS haftasının hesabı ve julian günden yıl, ay ve günün bulunması					
Yıl	2001		2452254.0	Julian Day	
Ay	12	ymd>>jd	344	Yılın Günü	
Gün	10	ymd<<jd	1144	GPS Haftası	
ut	12.0		129600.0	GPS Saniyesi	
			1	Haftanın günü	

\* ut değeri decimal olarak giriniz.  
\* Örn: 12:15:18 >>> 12.255

Şekil 4.8: Julian Günü ve GPS Haftası Hesabını yapan applet

Julian gününün (JD) hesabı için gerekli formülasyon [9];

yıl için Y, ay için M, gün için D ve saat için UT kullanılırsa

$$JD = \text{INT} [365.25 * y] + \text{INT} [30.6001 * (m+1)] + D + UT/24 + 1720981.5 \quad (4.97)$$

$$y = Y - 1 \quad \text{ve} \quad m = M + 12 \quad \text{eğer} \quad M \leq 2 \quad (4.98)$$

$$y = Y \quad \text{ve} \quad m = M \quad \text{eğer} \quad M > 2 \quad (4.99)$$

formülasyonu ile hesaplanabilir. Julian gününden tarihi hesaplamak için ise aşağıdaki adımlar kullanılabilir :

$$a = \text{INT} [\text{JD} + 0.5] \quad (4.100)$$

$$b = a + 1537 \quad (4.101)$$

$$c = \text{INT} [(b-122.1)/365.25] \quad (4.102)$$

$$d = \text{INT} [365.25 * c] \quad (4.103)$$

$$e = \text{INT} [(b - d) / 30.6001] \quad (4.104)$$

$$D = b - d - \text{INT} [30.6001 * e] + \text{FRAC} [\text{JD} + 0.5] \quad (4.105)$$

$$M = e - 1 - 12 * \text{INT} [e/14] \quad (4.106)$$

$$Y = c - 4715 - \text{INT} [(7 + M)/10] \quad (4.107)$$

Haftanın gününü hesaplamak için mod 7' ye göre aşağıdaki formül kullanılabilir :

$$N = \text{modulo} \{ \text{INT} [\text{JD} + 0.5] , 7 \} \quad (4.108)$$

GPS haftası ise aşağıdaki bağıntı ile elde edilebilir :

$$\text{HAFTA} = \text{INT} [(\text{JD} - 2444244.5)/7] \quad (4.109)$$

## 5. SONUÇLAR VE ÖNERİLER

Bu çalışmada, haritacılıkla ilgilenen herkesin ihtiyaç duyacağı temel jeodezik hesapların programları yapılmış ve internet ortamında kullanıcıların istifadesine sunulmuştur.

Programlama dili olarak, nesnel programlama tabanına sahip bir dil olan Java'nın Applet teknolojisi kullanılmıştır. Bundaki temel amaç, programların platformdan bağımsız bir şekilde çalışmalarının istenmesidir. Kullanıcının tek sahip olması gereken şey, Java'yı destekleyen bir internet tarayıcısıdır. Böyle bir tarayıcıyı kullanmıyorsanız, internet üzerinde interaktif bankacılık işlemlerinden tutun da daha birçok gelişmeden mahrum kalıyorsunuz demektir. Günümüzde popüler olarak kullanılan tüm tarayıcılar (Internet Explorer, Netscape vb.) zaten Java'yı desteklemektedir. İnternet programcılığının gelişmesinde Java'nın etkisi oldukça fazladır. Sonuç olarak kullanıcının müdahalesine gerek kalmadan sistem otomatik olarak kurulu bir şekilde gelmektedir.

Java'nın platformdan bağımsız çalışabilme özelliği sayesinde, ilerde ihtiyaç duyulabilecek sorunlar ortadan kalkmaktadır. Bu sayede, programların yerleştirileceği sunucunun önemi ortadan kalkmaktadır. Örneğin, Windows işletim sistemi tabanlı bir yazılım olan ASP (Active Server Pages) ile hazırlanan bir program sadece Windows tabanlı bir sunucuyla çalışabilmektedir. UNIX tabanlı bir sunucuda çalışması mümkün değildir. Fakat Java ile hazırlanan bir program yerleştirildiği sunucuda sadece bir misafir gibi davranır ve internet bağlantısı ile kullanıcının bilgisayarına yüklendikten sonra, sanki o bilgisayarın kendisinde saklıymış gibi hızlı çalışır.

Bu uygulamalar sadece internet üzerinde değil, kullanıcının bilgisayarında, tıpkı paket yazılımlarda olduğu gibi, herhangi bir bağlantıya gerek duymadan da çalışabilmektedirler. Üstelik hazır yazılımların ihtiyaç duyacağı bazı özel sistem gereksinimlerine de ihtiyaç duymazlar. Yukarıda belirtildiği gibi, sadece basit bir tarayıcının içinde, kullanıcı kolaylıklı bir şekilde çalışmaktadırlar.

Bu çalışma, genel bir program arşivine başlangıç teşkil etmektedir. Sayfadaki programları kullanan kişilerin önerileri doğrultusunda bu çalışmayı ilerletmek mümkündür. Aynı zamanda bu konuya ilgi duyanların katkılarıyla da sayfanın zenginleşeceği düşünülmektedir.



## KAYNAKLAR

- [1] **Morrison, M.**, 1996. Java 1.1 Unleashed (CD Book), Chapter 1, pp. 1-12 , Macmillan Computer Publishing, USA.
- [2] **Cohn, M.**, 1996. Java Developer's Reference (CD Book), pp. 23-29, Sams Publishing, Indianapolis.
- [3] **Jaworski, J.**, 1996. Java Developer's Guide (CD Book), pp. 38-48, Sams Publishing, Indianapolis.
- [4] **Norton, P. and Stanek, W.**, 1996. Peter Norton's Guide to Java Programming (CD Book), pp. 78-79, Sams Publishing, Indianapolis.
- [5] **Gulbransen, D. and Rawlings, K.**, 1996. Creating Web Applets with Java (CD Book), Chapter 4, pp. 112,123, Sams Publishing, Indianapolis.
- [6] **Ayhan, M.E.**, 1995. Jeodezi Ders Notları.
- [7] **Strang, G. and Borre, K.**, 1997. Linear Algebra, Geodesy and GPS, pp. 469-480, Wellesley-Cambridge Press, Wellesley.
- [8] **Nakiboğlu, M.**, 1994. Jeodezi Ders Notları.
- [9] **Hofmann-Wellenhof, B., Lichtenegger, H. and Collins, J.**, 1994. Global Positioning System, Theory and Practice, pp. 41-42, Springer-Verlag, New York.
- [10] **Ayhan, M.E., Kılıçoğlu, A.**, 1995. Türkiye için Avrupa Datumu 1950 (ED-50) ile Dünya Jeodezik Sistemi 1984 (WGS-84) Arasındaki Dönüşüm Parametrelerinin Hesaplanması, Türk Haritacılığının 100. yılı TUJJB ve TUFUAB Kongreleri Bildiri Kitabı, Cilt II, 1-5 Mayıs, Ankara, Harita Genel Komutanlığı, sayfa 348-357.
- [11] **Hopson, K.C.**, 1996. Developing Professional Java Applets (CD Book), Sams Publishing, Indianapolis.
- [12] **Valnum, C.**, 1996. Java by Example (CD Book), Que Corporation Publishing, Indianapolis.
- [13] **Weber, J.**, 1996. Special Edition Using Java (CD Book), Que Corporation Publishing, Indianapolis.
- [14] **Girdley, M. and Jones, K.A.**, 1996. Web Programming with Java (CD Book), Sams Publishing, Indianapolis.

- [15] **Ablan, J.**, 1996. Developing Intranet Applications with Java (CD Book), Sams Publishing, Indianapolis.
- [16] **Wutka, M.**, 1997. Java Expert Solutions (CD Book), Que Corporation Publishing, Indianapolis.
- [17] **Cadenhead, R.**, 1996. Teach Yourself Java Programming in 24 Hours (CD Book), Sams Publishing, Indianapolis.
- [18] **Lemay, L., Perkins, C.L. and Morrison, M.**, 1996. Teach Yourself Java Programming in 21 Days Professional Reference Edition (CD Book), Sams Publishing, Indianapolis.
- [19] **Joshi, D., Perkins, C.L. and Lemay, L.**, 1996. Teach Yourself Java in Cafe in 21 Days (CD Book), Sams Publishing, Indianapolis.



**TC. PUSAT KOGNITIF KURDI:  
KORDUNANTASYON MERKIZI**

## ÖZGEÇMİŞ

Ali İhsan KURT, 1974 yılında Akşehir’de doğdu. İlk öğrenimini Çivril İsmail Özcan İlkokulunda, orta öğrenimini de Akşehir Atatürk ortaokulunda 1988 yılında tamamladı. Aynı yıl Kuleli Askeri Lisesi’ne başladı ve 1992 yılında buradan mezun olarak Kara Harp Okulu’nda öğrenimine devam etti. 1996 yılında Kara Harp Okulu’ndan Harita Teğmen olarak mezun olarak M.S.B. Harita Genel Komutanlığı Harita Yüksek Teknik Okulu’nda iki yıl süreyle Jeodezi ve Fotogrametri eğitimi aldı.

1998 yılında Harita Yüksek teknik Okulu’ndan Harita Mühendisi diploması alarak aynı yıl Harita Genel Komutanlığı Jeodezi Dairesi Uzay Teknikleri Şubesinde göreve başladı ve halen aynı dairede GPS Hesap Subayı olarak görevine devam etmektedir.