**Kaya Han Taş**

**20183405003**

**UBT**

***YÖRÜNGE MEKANİĞİ ARASINAV ÖDEVİ***

***SORU 1:*** *Öğrenci numaranızın son iki hanesini asteroidin numarası olarak alın.*

*Bu numarayı ssd.jpl.nasa.gov/horizons.cgi sitesinde kullanarak doğum tarihinizde cismin ekliptikel konum ve hız vektörlerini elde edin.*

***a)*** *Cismin o tarihteki konum ve hız vektörlerini kullanarak o tarihteki ve enberiden geçiş zamanını () hesaplayın.*

***b)*** *1 yıl sonra cismin 'sini (yani Güneş'ten uzaklığını), yörünge hızını, , ve açılarını hesaplayın.*

***ÇÖZÜM:***

Öncelikle Horizon sitesinden aldığımız değerleri ve incelediğimiz asteroid ile alakalı bilgileri yazıyoruz.

***İncelediğimiz cisim (Target Body):*** *03 Juno*

***Koordinatın merkezi (Coordinate Origin):*** *Güneş (cisim merkezi)*

***İncelediğimiz tarih:*** *30.11.2000*

***Konum Vektörleri:***

***rx:*** *2.239438437258419 AB,* ***ry:*** *-0.4730590511961436 AB,* ***rz:*** *0.01927016483610738 AB*

***Hız Vektörleri:***

***Vx:*** *-0.0001863873512393852 AB/Gün,* ***Vy:*** *0.01184904856109035 AB/Gün,* ***Vz:*** *-0.002681282244695262 AB/Gün*

***k2 (GM Değeri):*** *0.00029591220929559093 AB3 Gün-2 MGüneş-1*

Son olarak başlamadan önce soru çözümünde lazım olacak olan ve ’yi tanımlayıp, normlarını (büyüklüklerini) bulup, açısal momentum (angular momentum) değerini bulmak işimizi kolaylaştıracaktır. Öncelikle ve ’yi yani konum ve hız vektörlerini tanımlıyoruz.

Şimdi bu vektörlerin normlarını (büyüklüklerini) aşağıdaki gibi buluyoruz.

Bu değerleri bulduktan sonra açısal momentumu (angular momentum) aşağıdaki formül ile bulabiliriz.

Yukarıda gösterildiği gibi determinant alıyoruz ve elde edeceğimiz sonuç bize vektörünü yani () değerlerini verecektir. Bu değerleri işlem sonucunda x,y,z değil i,j,k olarak bulmuş oluyoruz fakat zaten aynı koordinat sistemini bize bildirmekte olduğundan dolayı direkt x,y,z olarak da alabiliriz. Buna göre işlemimizi yaparsak:

Buradan vektörünü aşağıdaki gibi vektör formunda yazabiliriz.

Son olarak vektörünün büyüklüğünü buluyoruz.

Artık sorumuzun ***a)*** kısmı ile başlayabiliriz. Soruda olan parametre sıralamasına bağlı kalmayarak, farklı bir sıralama ile parametreler bulunacaktır. Ayrıca Horizon verileri sorunun sonunda verilecektir.

***a)***

***1. Inclination ():***

İlk olarak eğikliği (inclination) yani değerini bularak başlıyoruz. Bunun için aşağıdaki formülü kullanıyoruz.

Burada ifadesi direkt olarak vektörünün z bileşenidir. ise vektörünün normudur (büyüklüğüdür). Buna göre değerlerimizi yerine yazarak değerini buluyoruz. Burada önemli olan noktalardan biri arccos fonksiyonunu radyan değil derece olarak hesaplamaktır. Şimdi hesabımızı yaparsak;

Bu değeri Horizon (artık JPL olarak bahsedilecek) sayfasından aldığımız sonuç (IN) ile karşılaştırıyoruz.

***Bulduğumuz (inclination) değeri:***

***JPL’den alınan verideki inclination değeri:***

***2. Çıkış düğümü boylamı (Longitude of Ascending Node) (Ω):***

vektörünü bulduktan sonra çıkış düğümü boylamını bulmamız daha kolay olacaktır. Bunun için önce ***Ω*** formülünü yazıyoruz.

Burada vektörünü yani çıkış düğümü vektörümü tanımlamamız gerekmektedir. vektörü aşağıdaki şekilde tanımlanabilir.

Yani direkt olarak vektörünün bileşenlerini kullanarak hem çıkış düğümü vektörünü hem de çıkış düğümü boylamını bulabilmekteyiz. Buna göre çıkış düğümü vektörü ve normunu buluyoruz.

Artık değerini bulabiliriz. Yine arccos fonksiyonun içi derece olarak alınacaktır.

Bu değeri JPL sayfasından aldığımız sonuç (OM) ile karşılaştırıyoruz.

***Bulduğumuz (Longitude of Ascending Node) değeri:***

***JPL’den alınan verideki Longitude of Ascending Node değeri:***

***3. Yörüngenin basıklığı (Orbital eccentricity) ():***

Önce yörüngenin basıklığını yani değerini bulmak için kullanacağımız formülü yazıyoruz.

Burada ; vektörünün büyüklüğünün karesi, ; vektörünün büyüklüğü ve ; sorunun başında bahsettiğimiz değeri yani *0.00029591220929559093* değerine eşittir. Ayrıca belirtmemiz gerekir ki noktasal bir çarpımdır. Önce bu çarpımı yapmak işimizi kolaylaştırır. Sonrasında ve vektörlerinin katsayılarını direkt bulmamız daha uygundur. Çünkü simgesinden görüleceği üzere sonuç bir vektör olacaktır. Yani örneğin bileşenini bulmamız için katsayıları ve değerleri ile çarpmamız gerekecektir. Bunu x,y,z bileşenleri için teker teker yapmamız gerekmektedir.

O halde bahsettiğimiz gibi önce noktasal çarpımını buluyoruz. Sonucu tabii ki de skaler çıkacaktır.

*ve* vektörlerinin büyüklüklerini ilk başta bulmuştuk. Buna göre katsayıdaki işlemlerde kullanılacak üzere vektörünün büyüklüğünü ve vektörünün büyüklüğünün karesini yazıyoruz. *(Bu noktadan itibaren vektör normları normal harf olarak da gösterilecektir. Yani vektör işareti olanlar vektör, olmayanlar normları gösterecektir.)*

Artık formülündeki ve vektörlerinin katsayılarını bulabiliriz. olmak üzere katsayıları buluyoruz.

*vektörünün katsayısı:*

*vektörünün katsayısı:*

Şimdi teker teker vektörünün bileşenlerini buluyoruz.

Son olarak vektörünün büyüklüğünü buluyoruz.

Bu değeri JPL sayfasından aldığımız sonuç (EC) ile karşılaştırıyoruz.

***Bulduğumuz (Eccentricity) değeri:***

***JPL’den alınan verideki Eccentricity değeri:***

***4. Yarı-büyük eksen uzunluğu (Semi-major axis) ():***

Şimdi yörüngenin yarı-büyük eksen uzunluğunu yani değerini bulmak için kullanacağımız formülü yazıyoruz.

Elimizde bu formülde gerekli olan vektörünün büyüklüğü, vektörünün büyüklüğü, GM değeri bulunmakta. Buradan yarı-büyük eksen uzunluğu ’yı çekerek sonucu bulmuş oluruz. Önce ve vektörlerinin büyüklüklerinin karesini alıyoruz.

Şimdi yazdığımız formülü ’ya göre düzenliyoruz.

Artık değerlerimizi yerine yazarak değerini bulabiliriz. olmak üzere:

Bu değeri JPL sayfasından aldığımız sonuç (A) ile karşılaştırıyoruz.

***Bulduğumuz (Semi-major axis) değeri:***

***JPL’den alınan verideki Semi-major axis değeri:***

***5. Enberinin argümanı (Argument of Perifocus) ():***

Enberinin argümanının () değerini bulmak için kullanılacak formülü yazıyoruz.

Anlaşılacağı üzere ve vektörleri noktasal olarak çarpılacaktır. Bununla beraber büyüklüklerinin çarpılması gerekmektedir. Bu iki işlemi yapmadan önce ve vektörlerini hatırlamamızda fayda var.

Şimdi noktasal çarpımını yapıyoruz.

ve değerlerini zaten önceden bulmuştuk. Onları da hatırlatmakta fayda var.

Şimdi yapmamız gereken bu iki değeri çarpmaktır.

Artık bu değerleri yerine yazarak değerini bulabiliriz. Yine arccos’un içi derece olarak alınmalıdır.

Burada özel bir durum bulunmaktadır. Az önce bulduğumuz değeri düzeltilmelidir! vektörüne bakarsak bileşeninin sıfırdan küçük yani negatif bir değer olduğu görülmektedir. Böyle bir durumda bulunan değerinin düzeltilmesi gerekmektedir çünkü olmaktadır. Bu durumda düzeltme olarak kullanacağımız eşitlik aşağıda gösterilmiştir.

Buna göre düzeltilmiş değerimizi bulabiliriz.

Bu değeri JPL sayfasından aldığımız sonuç (W) ile karşılaştırıyoruz.

***Bulduğumuz (Argument of Perifocus) değeri:***

***JPL’den alınan verideki Argument of Perifocus değeri:***

***6. Ortalama Anomali (Mean anomaly) ():***

Ortalama Anomali () değerini bulmak için kullanacağımız formül aşağıdaki gibidir.

yani eccentricity (eğiklik) değerini biliyoruz. Bulmamız gereken değer yani dış ayrıklıktır. Buradan ortalama anomaliye geçiş yapabiliriz. Önce değerini bulmak için formülü yazıyoruz.

Bu denklemde ve değerleri bilinmekte fakat değeri bilinmemektedir. Bu nedenle değerini bulmak için gerekli olan formülü yazmalıyız. konum vektörünün yörünge düzlemi üzerindeki bileşenlerinden biridir. Buna göre formülü yazarsak:

Burada ve değerleri bilinmektedir fakat bu seferde yani öz kiriş değerini bilmiyoruz. Bu değeri bulmak için iki tane formülü kullanabilmekteyiz. Bu iki formül aşağıda verilmiştir.

’nin eşit olduğu iki ifadeye de değerleri yazarak çifte kontrol etmiş olacağımızdan değerleri her iki ifadeye de yazıyoruz.

Artık yani öz kiriş değerinin olduğunu biliyoruz. Bunu götürüp diğer değerler ile beraber   
 denklemine yazıyoruz. ve değerlerini önceki kısımlarda bulmuştuk.

değerini de bildiğimize göre artık açısını bulabiliriz. Formülde direkt olarak değerleri yerine yazıyoruz. Burada arccos fonksiyonunun içi *radyan* olarak alınmalıdır.

Artık sonunda Ortalama Anomali yani değerini bulabiliriz. Bunun için en başta yazdığımız formülde bulduğumuz değerini ve eccentricity (eğiklik) değerini yerine yazıyoruz. Burada E değerini yine radyan cinsinden yerine yazmamız gerekmektedir.

Şimdi bulunan bu değerini düzeltmek için 360 dereceden çıkarmamız gerekmektedir.

Bu değeri JPL sayfasından aldığımız sonuç (MA) ile karşılaştırıyoruz.

***Bulduğumuz (Mean anomaly) değeri:***

***JPL’den alınan verideki Mean anomaly değeri:***

***7. Enberiden geçiş zamanı (Periapsis time) ():***

Enberiden geçiş zamanını () bulmak için kullanacağımız formülü yazıyoruz.

Şimdi denklemi ’ya göre yeniden düzenliyoruz.

Burada ortalama açısal hız (derece/gün cinsinden), ortalama anomali ve t ise doğum tarihinin Julian’s Date cinsinden değeridir. Burada bilinmeyen tek değer yani ortalama açısal hızdır. Bu değeri bulmak için kullanacağımız formül aşağıdaki gibidir.

Burada periyot olup değerini aşağıdaki formül ile bulabilmekteyiz.

Bu formülde parametresi değerini hesapladığımız semi-major axis (yarı-büyük eksen uzunluğu) olup buradan direkt olarak periyotu bulabiliriz.

Buradan ortalama açısal hızı hesaplamadan önce az önce bahsettiğimiz gibi ortalama açısal hız derece/gün cinsinden bulunmalıdır. Bu nedenle bulunan değeri gün’e çevirmeliyiz. 365.2564 gün 1 yıl olmak üzere;

Artık ortalama açısal hızını hesaplayabiliriz. Formülde bulduğumuz değerini yazıyoruz.

Son olarak ’yu hesaplamadan önce bulmamız gereken doğum tarihinin Julian’s Date cinsinden değeridir. Bunun için direkt olarak JPL’de verilen (A.D.) değerini kullanıyoruz. Bir önceki çözümden değerini zaten olarak bulmuştuk. Buna göre değerini hesaplayabiliriz.

Bu değeri JPL sayfasından aldığımız sonuç (Tp) ile karşılaştırıyoruz.

***Bulduğumuz (Time of periapsis) değeri:***

***JPL’den alınan verideki Time of periapsis değeri:***

**b)**

Bize bir yıl sonra cismin bazı parametreleri sorulmakta. Bunun için JPL’den 30.11.2001 yılı için veriler; işlemler ve kontrol için kullanılacaktır. Bazı parametrelerde ufak değişiklikler olduğundan dolayı hassas işlem yapmak için 2001 yılı verilerini kullanmak daha uygun olacaktır. Yine çözümlerimizde 2000 ve 2001 yılında bu parametreler karşılaştırmalı olarak verilecektir.

***1. Ortalama Anomali (Mean anomaly) ():***

İlk olarak periyotu yani ’yi hesaplayabiliriz. Yarı-büyük eksen uzunluğunda 2000 ve 2001 yılları arasında küçük bir değişiklik bulunmaktadır. Bunu gösterecek olursak:

*değeri 2000:*

*değeri 2001:*

Burada hassasiyet bakımından 2001 yılının yarı-büyük eksen uzunluğunu JPL sitesinden alarak kullanıyoruz. Kendimiz de hesaplayabiliriz fakat önceki şıklarda detaylı olarak bunu gerçekleştirmiştik.

Şimdi periyotu a şıkkında kullandığımız formülü ve 2001 yılı değerini kullanarak bulabiliriz.

Buradan açısal hız yine a şıkkında hesapladığımız gibi hesaplanacak olup sonucumuzda ufak bir değişiklik olacaktır.

Burada Enberiden geçiş zamanını yani ’yu JPL sitesinden alabiliriz. 2000 ile 2001 yılları arasında ufak bir değişim bulunmaktadır. 2000 yılı değerini kullanmamız durumunda fazla bir fark olmayacak olup hassasiyet bakımından biz JPL’den alınan 2001 değerini işlemlerimizde kullanıyoruz.

*değeri 2000:*

*değeri 2001:*

30.11.2001 yılının denk geldiği Julian’s date değeri olup bu değeri ve yukarıdaki değerleri kullanarak değerini buluyoruz. Bunun için kullanacağımız formülle beraber işlemler aşağıda gösterilmiştir.

Bu değeri JPL sayfasından aldığımız sonuç (MA) ile karşılaştırıyoruz.

***Bulduğumuz (Mean anomaly) değeri:***

***JPL’den alınan verideki Mean anomaly değeri:***

***2. Gerçel Anomali (True anomaly) ():***

Gerçel anomali değerini bulmak için kullanacağımız formül aşağıdaki gibidir.

Burada eccentricity yani değerini biliyoruz. Bulmamız gereken *E* değeri olup aşağıdaki denklemden bulabiliriz.

Burada bu denklemin bir analitik çözümü bulunmadığından dolayı değerine yaklaşım yani iterasyon yapılmalıdır. Bu iterasyon metodunda değeri olarak değerini kullanacak olup köke yaklaşım yapılacaktır. Bunu formülsel olarak gösterecek olursak, önce denklemi E için düzenliyoruz:

Sonrasında iterasyonu belirtecek şekilde denklemimizi yazıyoruz. sin fonksiyonun içine yazılacak değer radyan cinsinden olmalıdır. Buna göre:

Son olarak yine hassasiyet açısından değerini JPL’den 2001 yılı için alıyoruz. 2000 yılı ile 2001 yılı arasında pek bir fark bulunmamaktadır fakat yine de hassasiyet açısından 2001 değerini kullanıyoruz. İki değer arası fark aşağıdaki gibidir.

*değeri 2000:*

*değeri 2001:*

Bahsettiğimiz gibi ilk iterasyonda olacak şekilde, değeri olarak değerini kullanıyoruz ve bu şekilde iterasyona başlıyoruz. İlk iterasyonumuz aşağıdaki gibidir.

Bu iterasyonu istediğimiz hassasiyete kadar devam ettiriyoruz. Örneğin hassasiyet olarak 0.000001 değerini aldığımızı varsayarsak buna bir program yazarak iterasyonu bu hassasiyete kadar gerçekleştirebiliriz. Bu hassasiyet değerine ulaşmamız için yapacağımız 9 iterasyon sonrasında bulacağımız E değeri (kodu soru sonunda verilmiştir):

Artık bunu aşağıdaki formülde yerine yazabiliriz.

, olmak üzere:

Bu değeri JPL sayfasından aldığımız sonuç (TA) ile karşılaştırıyoruz.

***Bulduğumuz (True anomaly) değeri:***

***JPL’den alınan verideki True anomaly değeri:***

**(Sonuç hatalı, Newton-raphson metodu için ayrıca kod yazıldı (soru sonunda kodlar bulunmakta) fakat sonuç bulunamadı)**

***3. Güneşe olan uzaklık ():***

Direkt olarak JPL’den alacağımız konum vektörlerini kullanarak bu vektörün büyüklüğünü bulup sonuca ulaşabiliriz. Bu işlemi zaten önceki şıkta yaptığımızdan dolayı kısaca r değerimiz aşağıdaki gibidir.

Bir önceki kısımda true anomaly değerimiz hatalı çıktığından dolayı JPL’den alınan değer kullanılacaktır. Juno’nun Güneşe olan uzaklığını bulmamız için kullanacağımız formül aşağıdaki gibidir.

değerini bulabilmek için formülümüz aşağıdaki gibidir.

Bu değerlerin hepsi bizde bulunduğundan dolayı değerini hesaplayabiliriz. olmak üzere:

Artık cismin güneşe olan uzaklığını bulabiliriz.

Bu iki değer birbirine yakın olduklarından dolayı sonucun doğruya yakın bir değer olduğu söylenebilir.

***4. Yörünge hızı ():***

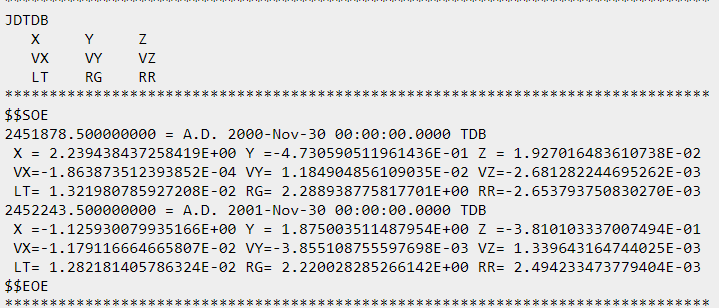
Bunun için yine bir önceki işlemde yaptığımız gibi Direkt olarak JPL’den alacağımız hız vektörlerini kullanarak bu vektörün büyüklüğünü bulup sonuca ulaşabiliriz. Bu işlemi zaten önceki şıkta yaptığımızdan dolayı kısaca değerimiz aşağıdaki gibidir.

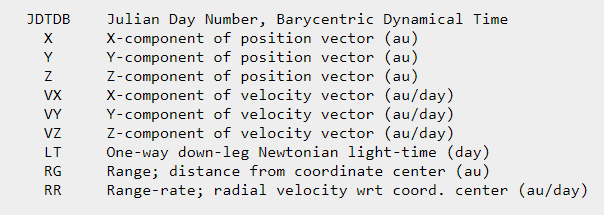
Bu hızı bulabilmek için kullanabileceğimiz yöntem aşağıdaki gibidir.

Bu formüldeki bütün değerleri bildiğimizden direkt olarak yerine yazabiliriz.

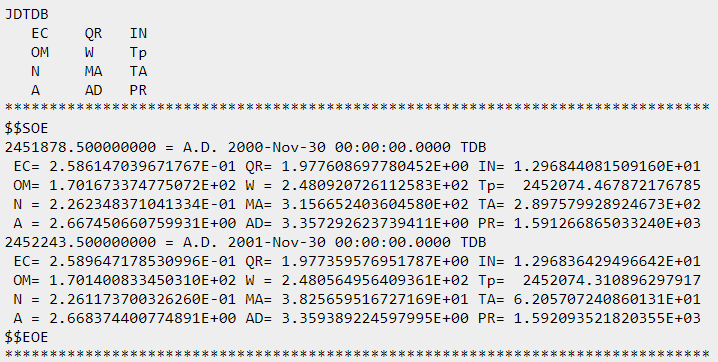
Bu iki değer birbirine yakın olduklarından dolayı sonucun doğruya yakın bir değer olduğu söylenebilir.

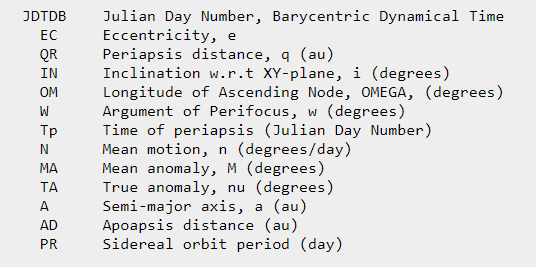
***JPL VERİLERİ***





***(HIZ VE KONUM VEKTÖRLERİ)***





***(ORBITAL ELEMENTS)***

***İTERASYON KODLARI (E Değeri için):***

***1. YOL:***

*import numpy as np*

*M=38.256647361500534*

*#Mean Anomaly (in degrees)*

*e=0.2589647178530996*

*#eccentricity*

*Eo=38.256647361500534*

*#E\_old*

*En=0*

*#E\_new*

*D=M+((180/np.pi)\*e\*np.sin(np.deg2rad(Eo)))*

*print("1. iterasyon: ", D)*

*En=D*

*i=1*

*while True:*

*if np.abs(En-Eo)<0.000001:*

*break*

*else:*

*D=M+((180/np.pi)\*e\*np.sin(np.deg2rad(En)))*

*i=i+1*

*print(i,". iterasyon: ", D)*

*Eo=En*

*En=D*

***(Sonuç: 49.54716421277749 derece)***

***2. YOL (Newton-Raphson):***

import numpy as np

e=0.2589647178530996

#eccentricity

M=np.deg2rad(38.256647361500534)

#mean anomaly (in radians)

Eo=(0.6677044572325856)

#E\_old (in radians)

En=0

#E\_new

fe=Eo-(e\*np.sin(Eo))-M

derivativefe=1-(e\*np.cos(Eo))

D=Eo-(fe/derivativefe)

print("1 . iterasyon: ",D, "rad")

En=D

i=1

while np.abs(En-Eo)>=0.000001:

fe=En-(e\*np.sin(En))-M

derivativefe=1-(e\*np.cos(En))

D=En-(fe/derivativefe)

i=i+1

print(i,". iterasyon: ",D, "rad")

Eo=En

En=D

***(Sonuç: 0.8647611512146033 rad)***

***3. YOL (Newton-Raphson Derece Düzeltmesi ile):***

#Newton-Raphson Method

import numpy as np

M=38.256647361500534

#mean anomaly (in degrees)

e=0.2589647178530996

#eccentricity

Eo=38.256647361500534

#E\_old (in degrees)

En=0

#E\_new

fE=Eo-(e\*(np.sin(Eo)))-M

#Radyan cinsinden yazıldığından derece için düzeltme ile beraber denklem

En=Eo+((M-Eo+(180\*np.pi)\*e\*np.sin(Eo))/(1-(e\*(180\*np.pi)\*np.cos(Eo))))

print("1. iterasyon: ", En)

i=1

while True:

if np.abs(En-Eo)<0.0001:

break

else:

D=En+((M-En+(180\*np.pi)\*e\*np.sin(En))/(1-(e\*(180\*np.pi)\*np.cos(En))))

i=i+1

print(i,". iterasyon: ",En)

Eo=En

En=D

***(Sonuç: 0.6579067209295193 rad)***

***DOĞRU SONUÇ: 1.0831002376786207 rad***

***SORU 2:*** *Eliptik bir yörüngede dolanan cismin enberi ve enöte noktalarındaki hızları ve , enberi ve enöte mesafeleri ve ise olmak üzere:*

olduğunu gösterin.

***ÇÖZÜM:***

Öncelikle bu çıkarımda/ispatta kullanılacak açılımları yazarsak işimiz kolaylaşacaktır.

Şimdi işlemlerimize başlayabiliriz. Amacımız ’nın bu iki denkleme eşit olduğunu göstermek. Elimizde ve ’nin açılımları bulunmakta. Fakat ’nin bir açılımı bulunmamakta, bu yüzden ilk olarak ’yi eşit olduğu denklemden çıkarmaya çalışıyoruz.

Önce ifadesini karşıya atıp kök içine karelerini alıp yerleştirmek uygundur.

Burada görüleceği üzere ’nın açılımı ters bir şekilde denklemimizde bulunmakta. Gerekli sadeleştirmeler yapılırsa ve açılımı yerine yazılırsa elde edilecek denklem:

Şimdi ve açılımlarını da kullanıp gerekli sadeleştirmeleri yaparak ’nin son halini yazıyoruz.

Artık denklemini üç ifadeyi de açtığımızdan dolayı rahatlıkla yazabiliriz.

Şimdi kök içerisinde bulunan ifadesinin pay ve paydasını değeri ile çarpıyoruz.

Sadeleştirme yaparken kolaylık olması açısından yerine ve yerine yazarsak elde edilen denklem aşağıdaki gibi olmaktadır.

Kökün dışarısında bulunan ifadesini karesini alarak kökün içine alıyoruz.

Gerekli sadeleştirmeleri yapıyoruz.

Son olarak ifadesini olarak açıyoruz.

Tam olarak soruda verdiği hale dönüştürmek için çarpım işaretini vs. kaldırırsak elde edeceğimiz son ifade:

Bu çıkarımdan anlaşılacağı üzere soruda verilen ifadeler birbirine eşittir.