Emrehan HALICI

ÇÖZÜLEMEYEN PROBLEMLER

Uğur HALICI*

Sayılarla ilgili çözülemeyen problemlerden en ilginci Palindrom Problemidir. En az iki haneli bir sayı yazın ve bunun altına sayının tersini yazarak ilk sayı ile toplayın. Bundan sonra toplam sayının tersini yazarak tekrar toplayın. Bu işleme bir palindrom, yani tersten ve düzden aynı okunan bir sayı elde edinceye kadar devam edin. Örneğin 1988 sayısı ile başlayalım. Sadece yedi adımda 2322232 palindromu ortaya çıkar:

1988	10879	108680	195481	380072	650155	1201211
+ 8891	+ 97801	+ 086801	+ 184591	+ 270083	+ 551056	+ 1121021
10879	108680	195481	380072	650155	1201211	2322232

Yüzyıllar boyunca, her başlangıç sayısının bir palindrom ile sonuçlanacağına inanıldı. Charles Trigg adlı emekli bir matematikçi bundan emin değildi. Sayıları sırayla denemeye başladı ve 89'a gelinceye kadar olan sayıların altı veya daha az adımda bir palindroma ulaştığını gördü. Ancak 89 sayısından bir palindrom elde edebilmek için 24 adım gerekiyordu. Şüphesiz 98 sayısı da 24 adımda bir palindroma ulaşıyordu. Bundan sonra 196 sayısına ulaşıldığında, problemdeki zorluk ortaya çıktı. Trigg, 100 adım ilerlediği halde 196 sayısından bir palindrom elde edememişti ve 1000'e kadar olan sayılar arasında, palindrom üretilemeyecek gibi görünen tam 149 tane sayı buldu.

1975 yılında, Harry Saal adlı bir bilgisayarcı, bilgisayar yardımıyla 196 sayısını 237310 adım ilerletti. Toplam 98305 haneye ulaştığı halde, henüz bir palindrom elde edememişti. Gariptir ki, ikili sayı sistemlerinde palindroma ulaşamayan sayıların varlığı ispat edilmiş olmasına rağmen, bugüne kadar kimse diğer sayı sistemlerinde palindroma ulaşamayan bir sayının varlığını ispat edemedi.

Aynı şekilde, $a^n + b^n = c^n$ denkleminin tamsayılarla çözümü, üzerinde en çok uğraşılan ve çözülemeyen problemlerden biridir. Burada $n = 2$ olduğu zaman denklemin sonsuz çözümü vardır. Çözümlerden en basiti $3^2 + 4^2 = 5^2$'dir. Bu çözümler Pisagor üçlüsü diye adlandırılırlar, çünkü $n = 2$ olduğu zaman a, b ve c bir dik üçgenin kenarlarına karşılık gelmektedir. Eğer n sayısı ikiden büyük alınırsa, bu denkleme bir çözüm olup olmadığı matematikçileri uzun yıllar uğraştırmıştır. 17. yüzyılda, Fransız matematikçisi Pierre Fermat, bir kitabın kenarına, problemin cevabının olumsuz olduğunu, yani bir çözüm bulunamayacağını gösteren mucize bir ispat yaptığını elyazısı ile yazmış ve oradaki yerin ispatı yazmaya yetmeyeceğini de eklemişti. Ancak, Fermat ispatın nasıl olduğunu hiç bir zaman açıklamadı ve birçok matematikçi, onun daha sonra ispatının yanlış olduğunu anladığına inandılar.

Bugüne kadar hiç kimse, ne Fermat'ın son teoreminin doğruluğunu ispatlayabildi, ne de teoremin yanlış olduğunu gösterecek bir örnek bulabildi. Bilgisayar kullanılarak, n sayısını 125.000'e ve a,b,c sayılarını milyonlarca haneye çıkaracak kadar çok deneme yapılmasına rağmen, yine de bir sonuca ulaşmak mümkün olmadı.

Çözülemeyen matematik problemlerinden bazıları o kadar önemlidir ki, bir çözüm, matematik dünyasında yankılar uyandırabilir. 1900 yılında David Hilbert, matematiğin yeterlilik ve tutarlılığını tartışmaya açan bir dizi problem sormuştu. Bunlardan birisi, örneğin $4 \times y^2 + 2 \times y^2 z^3 - 11 \times {}^3 y^2 z^2 — 1164$ biçimindeki bir Diophantine denklemine tam sayı x, y, z çözümlerinin olup olmadığına karar verecek genel bir algoritmanın bulunup bulunamayacağını soruyordu. Daha sonra Hilbert'in (H.10.P) olarak anılacak olan 10. Problemi, 1971 yılında buna karar verecek bir algoritmanın olamayacağı ispatlanıncaya kadar çözümsüz kaldı.

1930'lu yıllarda, içlerinde Alan Turing'in de bulunduğu bir grup mantıkçı, matematiksel önermelerin doğruluğuna karar verecek algoritmaların varlığıyla ilgilenmeye başlamışlardı. Turing ve diğerleri ilk defa, bazı matematiksel problemlerin karar verilemez olduğunu, yani bu tür problemlerin her örnek durumunu çözecek yetenekte algoritmaların var olamayacağını ispatladılar. Karar verilemez problemlere ilk örnek bilgisayar programlarının "durmasını" inceleyen problemdi. Bu problemi anlayabilmek için önce bir bilgisayar programının nasıl çalıştığını inceleyelim.

Bir odada bir sehpa ve bir masa olsun. Odaya bir tepsi ile giriyorsunuz. Amacınız tepsidekileri masaya koymak veya masadakileri ya da sehpadakileri tepsiye koyarak odadan çıkarmak. Tepsi, masa ve sehpanın boş ya da dolu olması durumuna göre yapacağınız işler aşağıdaki gibi programlanmış olsun:

1. Odaya gir.

* *ODTÜ Elektrik ve Elektronik Müh. Bl. Öğretim Görevlisi.*

2. Eğer masa boş ise tepsidekileri masaya koy ve 4. adıma git, değilse devam et.

3. Eğer sehpa boş ise masadakileri sehpaya koy ve 2. adıma git, değilse devam et;

4. Eğer tepsi boş ise, sehpadakileri tepsiye koy ve 5. adıma git, değilse 2. adıma git;

5. Odadan çık.

Masa, sehpa ve tepsinin boş ya da dolu olmasına göre yapacağınız işler değişiktir. Eğer başlangıçta, masa ve tepsi dolu, sehpa boş ise, yukarıdaki programa göre önce masadakileri sehpaya, sonra tepsidekileri masaya ve en sonunda sehpadakileri tepsiye koyar ve odadan çıkarsınız. Burada odaya girmekle programa başlıyor, odadan çıkmakla programdan çıkıyorsunuz.

Bilgisayar programları da buna benzer bir biçimde çalışırlar. Yukarıdaki örnekte masa, sehpa ve tepsi program değişkenlerini oluşturmaktadırlar. Bunların başlangıçtaki değerleri, yani dolu ya da boş olması, program değişkenlerinin giriş verilerini oluşturmaktadır ve program içinde değişkenlerin değerleri değişebilmektedir. Programda kullanılan değişkenlerin aldıkları değerlere göre, programda hangi adımlardan geçileceği değişebilir. Bazı bilgisayar programlarında, bazı giriş verileri için program sonuna ulaşılamayabilir, yani program hiç durmayabilir. Yukarıdaki örnekte, eğer masa, sehpa ve tepsi, her üçü de dolu olarak programa başlanırsa, durmaksızın 2., 3. ve 4. adımlardan geçilmesi gerekecek ve hiç bir zaman program sonu olan 5. adıma ulaşılamayacaktır.

Durma Problemi (D.P.), herhangi bir bilgisayar programının verilen giriş değerleriyle birlikte, durup durmayacağına karar verilmesini ister. Herhangi bir bilgisayar programının durup durmayacağına karar veren genel bir algoritma var mıdır? Buna en kolay çözüm, verilen bilgisayar programının giriş verileriyle birlikte bir bilgisayarda duruncaya kadar çalışmasını gözlemektir. Ancak, durmayacak bir programın sonsuza kadar gözlem gerektirmesi ihtimali problemdeki zorluğu ortaya çıkarmaktadır. Program henüz durmadıysa, gözleme ne zaman son verip, programın durmayacağına karar vereceğiz? Eğer biz durmayacağına karar verdikten biraz sonra program duracaksa, gözlemimizi biraz daha uzatmamız gerekmektedir. Böylece gözlem süresine bir sınır koyulamayacağı ortaya çıkıyor. Turing, D.P.'nin her hali için başarıyla çalışacak bir algoritmanın varolamayacağını, yani D.P.'nin karar verilemez bir problem olduğunu ispatlamıştır.

Emil Post'un bir çalışmasıyla birlikte, problemlerin birbirlerine indirgenebilirliği kavramı ortaya atılmış oldu. İndirgenebilirlik kavramı, çözülebilen ve çözülemeyen problemler arasında bir sınır koymada en önemli araç haline geldi. Eğer B problemini çözen bir algoritma kullanılarak, A problemini çözecek bir algoritma oluşturulabiliyorsa, A problemi B problemine indirgenebilir, denmektedir. Örneğin D.P., H.10.P.'ne indirgenebilen bir problemdir.

D.P.'nin, H.10.P.'ne indirgenebilirliğinin gösterilmesiyle birlikte, H.10.P.'nin de karar verilemez bir problem olduğu ispatlanmış oluyor. Eğer H.10.P. karar verilebilir bir problem olsaydı, H.10.P.'nin çözümü için önerilen algoritma D.P.'ni çözmek için düzenlenebilecek ve D.P.'ni çözebilen bir algoritma ortaya çıkacaktı. Ancak D.P.'ni çözen bir algoritmanın olamayacağı Turing tarafından ispatlandığına göre, H.10.P.'nin de karar verilemez bir problem olması gerekiyor.

Bu noktada, bir problemin çözülebilirliği ile önerilen bir çözümün doğruluğunun denenebilir olmasını birbirinden ayırmak gerekiyor. Diophantine Denklemlerini çözecek genel bir metod bulunmamasına karşın, önerilen bir çözümün doğruluğunu denemek kolaydır. Örneğin $x = 3$, $y = 2$ ve $z = -1$ tamsayılarının, verilen denkleme bir çözüm olup olmadığını anlamak için birkaç aritmetik işlem yapmak yeterlidir. Ancak D.P. için, eğer program durmayacaksa, önerilen çözümün doğruluğunu denemek mümkün değildir.

Bazı çözülemeyen problemler bir kenarda unutulup giderken, bazıları ise matematikçi, mantıkçı, bilgisayarcı ve meraklıları daha uzun süre uğraştıracak gibi gözüküyor. □

SİZ OLSAYDINIZ?

(Satranç Dünyasındaki soruların cevapları)

Çözüm : I :

1..Axc3! 2.bxc3 (2.Vxd6 Ae2 taş kazanır.) 2..Kei kazanır, 3.Kxei Vxd3 ya da 3.Şg2 Kxdı var.

(Catalan - Tatai, Dubai 1984)

Çözüm II :

1.F5!! (1..g5? 2.Vf6 ya da 1..Şe7 2.Fxg6 Kf7 tehdidi ile veya 1..Ve5 2.Fxg6 Şg7 3.Af5 son varyant 1..Kxe4 2.Fxg6 Kf4 3.Vh8 ve arkasından Vxh7) 1..Axe4 2.Fxg6 Ff2 3.Kxf2 Axf2 4.Vf6 kazanır.

(Waagener - Cordara, Hamburg 1984)

Çözüm III :

1.Kg3! Kxg3 (1..Kxf2 2.Kg8 mat ya da 1..Vd6 2.Kxg2 Fxg2 3.Vxg2 Vxd4 4.Şc2 Fg7 5.f8V Fxf8 6.Af7 mat) 2.Vxg3 Fg7 3.Ag6! Vxg6 (3..hxg6 4.Vh4 Fh6 5.Vxh6 mat) 4.Vb8 kazanır.

(Makropoulos - Kallai, Calimanesti 1984)