

Şu Bilgisayarın İnsana Ettikleri

Birkaç yıl önce beni tedirgin eden bir an yaşamıştım. Bilgisayarımın başında otururken, bir programı çalıştırmak için ikonuna tıkladım. Yazılım, emirlerimi yerine getirip, programı diskte kayıtlı olduğu yerden okuyup, belleğe yüklemeye başladı; ancak hemen sonra sessizce toparlanıp, ekranı terk etti. İkona tekrar tıkladım, program da aynı gösteriyi tekrarladi. Üçüncü kez tıklayıp, gösteriyi bir kez daha izledim. İşte tam o anda, ikona bir kez daha tıklamaktan vazgeçtim. Buna kendimle ilgili yaptığım keşfim engel oldu. Keşfim, dördüncü denemenin ilk üç denemeden daha farklı olacağına inanıyorsam, bilgisayarın nasıl çalıştığına dair çok garip zihinsel bir modele sahip olduğumu anlamamdan başka bir şey değildi.

DAHA ne kadar tıklayıp duracaktım? Yoksa makineyi kendimin daha inatçı olduğuna mı ikna edecektim? Yeterince uzun süre tıklamaya devam edersem, bilgisayarın direncini kırabilir miydim? Bilgisayarın tekrar ve disiplinle eğitilebilecek bir hayvan olduğunu mu düşünüyordum yoksa? Ya da, aynı konuyu kafasına vura vura öğreteceğim, tutuk bir çocuk olduğunu mu düşünmüştüm? Bu olaydan sonra karşılaştığım bilgisayar hatalarını (bug) ve onlara verdiğim tepkileri bir günlükte tutmaya başladım. Ne zaman bilgisayarla ilgili bir şeyler ters gittiyse, not ettim: Neler oldu, ne yapıyordum, ne zaman oldu, karşılığında ben ne yaptım vs. Bu hata günlükleri şu anda 25 sayfaya ulaştı. Bu kayıtları tutma nedenlerimden biri, bilgisayar hatalarının doğası hakkında bir şeyler öğrenebilmekse de, ilgimi en çok çeken, insanların bilgisayara (ve hazırladığı kü-

çük sürprizlere) verdiği tepkiler. Bir başka deyişle, amaç bilgisayarı hatalardan arındırmaktan (debug) çok, kendimi hatalardan arındırmaktı.

Bir bilgisayar programını değil de, bir arabayı çalıştırıyor olsaydım art ar-

da iki üç deneme yapmak o kadar da saçma görünmezdi. Konağı çevirdiğiniz ilk seferde motor çalışmazsa, yapacağınız ilk iş çekiciyi çağırmak olmaz elbette. Çalıştırmak istediğim şey bir çim biçme makinesi olsaydı, kabloyu



art arda birçok kez çekip bırakmak gerekebilirdi. Nitekim, ilk çekişte çalışan bir çim biçme makinesi biraz şaşırıcıdır. Peki, bir bilgisayar bu makinelere niye farklıdır?

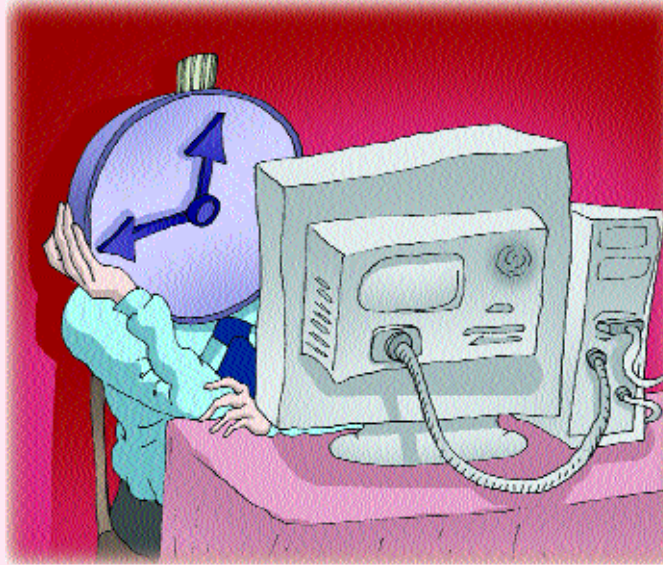
Gerçekte, bilgisayar çim biçme makinelerinden birçok nedenden ötürü farklıdır. Kuramsal bir bakış açısıyla, dijital bilgisayar, yaptığı her şey en küçük ayrıntısına değin öngörülebilir, belirlenimci (deterministik) bir makinedir. Daha özel bir biçimde söyleyecek olursak, bilgisayar belirlenimci bir sonlu-durum otomatu (finite-state automaton) ya da kısaca FSA'dır. Adından da anlaşılabilir gibi, bu tür bir makinenin mevcut sonlu sayıda olası durumu ya da konfigürasyonu vardır. Elektrikli aydınlatma anahtarları, sadece iki kararlı durumu bulunan, fazlasıyla basit bir FSA'dır. Bir dijital bilgisayardaki durum sayısıysa çok fazladır. Kabaca söyleyecek olursak, m bit bilgi saklayabilen bir bilgisayarın 2^m olası durumu vardır. m değerleri genellikle 100 milyonu aşar.

Bazı teknik ayrıntıları bir kenara bırakacak olursak, belirlenimci bir FSA şöyle çalışır: Makine iyi tanımlanmış bir başlangıç durumunda çalışmaya başlar, daha sonra gelen her girdiye göre, yeni bir duruma geçiş yapar ve olasılıkla birtakım çıktılar da üretir. Bilgisayar sözkonusu olduğunda, başlangıç durumu güç ilk kez verildiğinde belirlenmiş olur. Girdiler, bir tuşa basma ya da fareye tıklamak olabilir. Çıktılarsa, ekranda bir bilginin görüntülenmesi olabileceği gibi, bilginin yazıcıya gönderilmesi de olabilir. Her iki olasılıkta da, bilgisayarın durumu, bellekteki tüm bitlerin konfigürasyonu, merkezi işlem birimindeki yazmaç (register) bitlerin konfigürasyonu olarak tanımlanır. Manyetik diskte saklanan bitler, diskin bilgisayarın bir parçası mı, yoksa ikincil bir aygıt mı olarak tanımlanacağına bağlı olarak, bilgisayarın durumunun bir parçası olabilir de, olmayabilir de.

Bir sonlu-durum otomatu, bir sonraki durumu ve bir sonraki çıktısı, o andaki durumuna ve o anki girdisine

bağlıysa belirlenimcidir. Örneğin q durumunda makine α girdisini alıyor ve p durumuna geçiş yapıp β çıktısını veriyor. Böylece bir dahaki sefere makine q durumundayken, α girdisine aynı yanıtın (β çıktısı) verileceğini biliyoruz. Yani değişikliğe ve çeşitliliğe yer yok. Makineyi q durumuna getiren olaylar dizisi hiç önemli değil. Dış dünyadaki olaylar da öyle. Sadece o anki durum ve o anki girdi.

Bilgisayar işlemlerinin bu belirlenimci yapısı düşünülünce, bir çim biçme makinesiymişçesine bilgisayara yaptıklarım biraz gülünç görünüyor. Bilgisayar q durumundaydı, ben bir α girdisi üretiyordum. Denediğim her seferde bilgisayar bir işlem döngüsüne giriyor ve hemen sonra tekrar q durumuna geri dönüyordu. Bu durumda α



girdisi kaçınılmaz olarak tüm döngüyü yeniden başlatıyordu. Bozuk bir ampüllü aydınlatma anahtarına basarak yakma şansım neyse, programın da çalışma şansı oydu.

Indiana Üniversitesi'nden Douglas Hofstadter, benimki gibi davranışları tanımlamak için *sfeksçe* sözcüğünü üretmiş. *Sphex* cinsi yabancılara, yavrularına besin olacak, paralyze ettikleri bir cırcır böceğini öldürmeden önce kısa bir ritüel gerçekleştiriyor. Ritüel sırasında böceği oynatacak olursanız, arı ayine yeniden başlamak zorunda kalıyor. Böceği oynatmayı sürdürdüğünüz sürece, arı da aynı hareketleri yapmaya devam ediyor. Bir gözlemci bu numarayı bir arıya art arda tam 40 kez yineletmiş. Hofstadter'in de belirttiği gibi bu durumda, gözlemci de

en az arı kadar sfeksçe davranıyor. Başka bir gezegenden gelen bir gözlemci, hangi türün idare edilen, hangisinin idare eden olduğunu anlamada zor zamanlar yaşayabilirdi. Bilgisayarla yaşadığım çekişmede, bu sfeksçe dansı sürdürebilmek için ikimizin de birlikte çalışması gerekti.

Mikrodurumlar ve Makrodurumlar

Bir böcekte rastlanan zihinsel alışkanlıklarını sergiliyor olmanın ne kadar utandırıcı olduğunu bir düşünün! Ancak, günlüklerimi okumaya devam ettiğimde, davranışımın o kadar da sfeksçe olmadığını düşünmeye başladım. İnkâr edilemez sayıda kanıt, tekrar tekrar denemenin kimi zamanlarda gerçekten işe yaradığını gösteriyordu. Hatta, durmayıp dördüncü kez tıklasaydım belki de program başarılı bir biçimde çalışacaktı.

Günlük tuttuğum sıralarda başım bir iletişim yazılımıyla derde girmişti. Programı çalıştırabiliyordum; ancak ne zaman bir telefon numarasını aramasını istesem, program ölüyordu. Bir ay gibi bir süre içinde bu üç kez başıma geldi. Her seferinde programı, ilk seferinde yaptıklarımın aynısını yaparak,

yeniden başlattım ve o da düzgün biçimde çalıştı. Böyle ara sıra olan hataların nedenlerini asla bulamadım.

Bir başka sefer, bir program (kendi düşüncelerimi bu şekilde inceleme altına almama yol açan program) beni yukarıda anlatılanki gibi sfeksçe bir döngüye, pek de haber vermeden soktu. Yalnız bu sefer yeniden başlaması için tıkladığımda düzgün biçimde çalıştı. Sabit diske yüklemeye çalıştığımda bir program ilk seferinde düzgün kurulmamıştı. Ancak aynı işlem dizisini bir kez daha tekrarlayınca sorun çözüldü. Günlüklerim hâlâ devam eden ya da aradan 15 gün geçince çözülen sorunlarla dolu. Sonuç olarak demek istiyorum ki bilgisayar işlemlerinde de, ısrarcı çim biçme makinesi yaklaşımı işe yarayabiliyor.



Hata günlüklerim, sorunlarımın çoğunun tekrarladığını, ancak istendiği zaman tekrar üretilemediğini gösteriyor. Örneğin, en sık (dokuz olay) başıma gelen sıkıntı, sözcük işleme programımdaki bir "glitch" sonucu, ekrandaki belgede basılamayan bazı sembollerin belirmesi. Hatalar belli bir dizi işlemin ardı sıra oluşuyordu, ancak bu işlemler her seferinde hatayı kesin olarak oluşturuyordu. Hatalar ancak % 50 olasılıkla gerçekleşebiliyordu.

Koleksiyonumdaki en çıldırtıcı hatalardan birisi, görevi menülerin ve altmenülerin hiyerarşik bir biçimde görüntülenmesini sağlamak olan küçük yardımcı programın işlevini aksatıyordu. Yararlı bir yardımcı programdı; ancak hata oluştuğunda tüm sistem çöküyordu (hâlâ çöküyor). Daha ayrıntılı anlatayım; iç içe menülerden oluşan dizin içinde orta sıralardan bir menü seçecek olursam, çökmenin gerçekleşme olasılığı yaklaşık olarak yüzde ondu. Kısacası, çökmelerin sıklığı tehlikeli bir biçimde yaşamama ancak engel olabilecek oranda.

Peki bu rastgele hatalar ve iyileşmeler, bilgisayarın tümüyle belirlenimci bir makine olduğu görüşüyle nasıl uzlaşıyor? Yanıtlardan biri, fiziksel bilgisayarın gerçek anlamda belirlenimci olmadığıdır. Bilgisayar biliminin sonlu-durum makinesi, tıpkı geometrideki boyutsuz nokta kavramı ya da temel fiziğin sürtünmesiz makineleri gibi, bir soyutlama ya da idealleştirme. Gerçek bir bilgisayar, mükemmel olmayan parçalardan imal edilmek zorundadır. Muhtemelen donanım hatalarından ya da tasarım kusurlarından

ötürü, makine iki "ayrık" (discrete) durum arasında takılıp kalabilir. Kuantum dalgalanmaları sistemin kendiliğinden bir durumdan ötekine kaymasına yol açabilir. Belli birtakım girdiler –örneğin bellek yongasının içinden geçip giden bir kozmik ışın, enerji nakil hatlarına düşen bir yıldırım ya da mantıksız bir kullanıcının tekme-tokat bilgisayara saldırması vs.– makineyi belirlenimci olmayan (indeterminate) bir durumda bırakabilir. Tüm bu olgular, bilgisayarı sonlu-durum otomatu statüsünden uzaklaştırır.

Bu tip hatalar günlükümde pek az yer kaplıyor. Kurak kış aylarında bilgisayarımın başına oturduğum üç seferde parmağımdan fareye doğru bir kıvılcım sıçradı ve ilginç sonuçlara yol açtı. Ancak dış etkilerin ya da durumdaki rastlantısal değişikliklerin, günlükümdeki yüzlerce olayı açıklaması pek



mümkün değil. Sözcük işleme programımdaki saçma semboller hatası kozmik dalga saldırısı yüzündense, nasıl oldu da bu dalgalar her dokuz seferde (hem de üç farklı bilgisayarda) doğru biti bozabildiler? Bu tip açıklamalardaki sorun, çok fazla açıklama yapmalarıdır. Başınıza gelen hemen her durumu açıklayabilirler.

Çok ender durumlarda, bilgisayarlar tam bir sonlu-durum makinesi gibi davranırlar. Makineyi aynı duruma sokup, aynı girdiyi sağlarsanız, hep aynı sonucu alırsınız. Buradaki nokta, bilgisayarın her zaman aynı duruma sokmanın pek kolay bir iş olmamasıdır. $2^{100.000.000}$ olası durumu bulunan bir bilgisayarın, görünüşte birbirinin aynı, ancak iç ayrıntılarda farklı olan çok sayıda durumu bulunacaktır.

Belli bir durumu yeniden oluşturmak için, aynı program kümesini çalıştırıp, aynı veriler üzerinde aynı komutları işletmeyi deneyebilirsiniz. Bilgisayar donanımının bu görülen boyutuna makrodurum (macrostate) denebilir. Ancak her makrodurum için –temeli oluşturan bit örüntüsüne denk gelen– çok sayıda olası mikrodurum vardır. İki makrodurumda da çalışan aynı programlar, belleğin farklı alanlarına yüklenebilir ve çok farklı mikrodurumların oluşmasına yol açar. Aynı komutlar verilmesine karşın, görülemeyen ve bilgisayarın çalışmasını sağlayan çok sayıdaki geri plan işlemleri (ekranı tazelemek, imlecin yanıp sönmelerini sağlamak, klavyeden girdi beklemek vs.) nedeniyle farklı biçimde eşzamanlı kılınırlar. Makrodurum statik gibi görünse de, mikrodurum merkezi işlemci yongasının saatinin her periyodunda (saniyede 10, 50, 100 milyon kez ya da daha fazla) değişir.

Bir bilgisayar $2^{100.000.000}$ mikrodurumdan herhangi birine rastgele geçişler yapıyorsa, ikinci defa aynı duruma gelmesi olasılığı ihmal edilebilir (saniyede 100 milyon durumdan geçiş yapılırsa bile tekrarlama ortalama 10^{15} yıl alır). Elbette, geçişlerin hepsi rastgele değildir ve tahmin edileceği gibi tekrarlar çok olasıdır. Yine de, şu anda kullanılan her bilgisayarın içinde on yıllar boyunca geçen milisaniyeleri sayan dahili saatler bulunur. Bilgisayar, takvim başa dönmediği sürece aynı duruma bir daha uğramayacaktır.

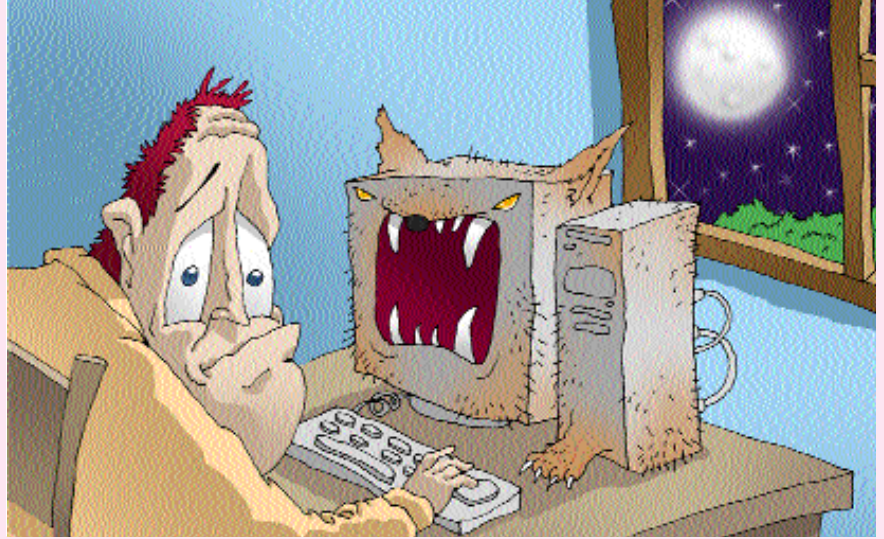
Makrodurumlar ve mikrodurumlar arasındaki birden-çoğa eşlemeler yüzünden, bilgisayarlar dahili çalışmalarında kararlı bir biçimde belirlemeci kalırken, klavyenin başındaki çaresiz kişiye çok kaprisli davranıyor görülebilirler. Bu garip kombinasyon, bilgisayarın mantık hatalarını düzeltme ya da kullanıcı üzerinde yarattığı psikolojik yaraları iyileştirme yaklaşımlarından hangisinin en iyi olduğu konusunda beni kararsız bırakıyor!

Ayın Evreleri

Bu konular hakkında deneyimi olan bir arkadaşım bir seferinde beni "Acelen olduğunu bilgisayarın anlamasına asla izin verme" diye uyarıyordu. Sorun, elbette, makinenin geciktirilmesi mümkün olmayan son teslim tarihinizi hissedip, çökmek için en uygun zamanı seçmesidir! Hatta bazı akıllı, bilgisayarlar Acele Posta Servisi'nin cumartesi günleri hafta içi günlerden daha erken kapandığını biliyor gibi davranır.

Silikonun içinde yaşayan ve bizi sürekli zor duruma düşürmeye çalışan yaramaz bir ruhun varlığına inanmamıza bizi zorlayan böylesine durumlarla dalga geçmek kolaydır. (Başka bir arkadaşım bana, "Bilgisayarları insan biçiminde düşünme, bundan nefret ederler" diye uyarıyordu). Yine de, acelesi olan bir kişiye bilgisayarın daha farklı davrandığı fikri, ilk bakışta görüldüğü kadar inanılmaz değildir. İlk, acelesi olan bir kişi farklı çeşitlilikte insan hataları yapar. Ben acele ederken de, boş zamanımda da hata yapabilirim ancak bunlar farklı tipte hatalar olurlar. Yaptığım farklı tipteki hatalar, bilgisayar yazılım ve donanımındaki farklı hataları ortaya çıkarırlar.

Bazen hızlı bir dizgici hiç yanlış yapmaksızın bir felaket yaratabilir. Bilgisayarların tarihinde keşfedilen en büyük hatalardan biri, 1980'lerin ortasında Therac-25 adlı bir ışın terapisi makinesinin kontrol yazılımındaydı. Hata, dehşet verici dozlarda ışının hastalara verilmesine ve sonuç olarak üç hastanın ölümüne yol açmıştı. Bu hatayı oluşturma yollarından birisinin, veri giriş ekranını hızlıca geçme olduğu anlaşılmıştı. Bunun sonucu olarak, bilgisayar doğru ayarları yapmaya zaman bulmadan, hastaya ışın verilmeye



başlanıyordu. Klavyeden aynı bilgilerin daha yavaş girilmesi hiçbir soruna neden olmuyordu.

Hatalar o kadar düzensiz ve gizemli şeylerdir ki, eski programcılar bunun ayın evrelerine bağlı olduğunu söyleyip, şaka yaparlardı. Ancak yine de böyle bir hata gerçekten vardır. (Öykünün orijinali Eric Raymond'un *New Hackers' Dictionary* kitabında bulunabilir.) O sıralarda MIT'de bulunan Guy L. Steele tarafından yazılan bir program, Ay'ın belli evrelerinde kendi veri dosyasını okumayı reddedebiliyordu. Veri dosyalarının bir zaman etiketi (time stamp) bulunuyordu. Bu etikete, normalde yer alan tarih ve saat bilgilerine ek olarak Steele, oyun olsun diye, Ay'ın evresini de eklemişti. Ay'ın belli evrelerinde zaman etiketi satır uzunluğu olan 80 harf sınırını geçiyordu, sonuç olarak dosyalar da okunamıyordu.

Bu kadar gizli ve anlaşılması zor hatalara sürekli olarak maruz kalmak, insanların bilgisayara doğüstü korkuyla yaklaşmasına yol açabilir. Kendi davranışlarının nedenlerini anlamaksızın, en son seferde işe yarayan yöntemlere ya da numaralara bağlanıp kalırlar. Kullandıkları programın yeni sürümünü yüklemenin sistemin harikulâde dengesini bozacağına inanırlar. Daha derinden etkilenmiş kişiler, klavyeyi temizleme ayınları yapmayı ya da kızgın bilgisayar tanrılarını hediyeleyerek teskin etmeyi deneyebilirler. Elbette, sfeksçeliğin başka bir türü olarak gördüğüm böyle bir davranış uygun bulmuyorum, ama yine de daha iyi sonuçlar vermesi kesin olan bir seçenek de öneremem.

Günlüğüme kayıtlı hatalı çalışmaların hepsinin mantıklı ve akılcı birer açıklaması var. Bundan kesinlikle eminim. Buna karşın, sadece bir avuç dolusu olayda mantıklı ve akılcı nedeni bulabilmem gerçeğiye ortada. Bu birkaç tanı başarısında da, ya programları kendim yazmışım ya da onu geliştiren kişinin doğrudan desteğini almışım. Programın kaynak koduna ulaşmaksızın, bir yazılım yanlısını gerçekten anlama umudu çok azdır. Bu durumda, boş inançlar sorunlarla başa çıkmak için en az başka yöntemler kadar iyidir.

Burada boş inançların genellikle benim de sığınağım olduğunu da eklemeliyim. Birkaç yıl önce *American Scientist*'in editörlük bürolarında "hijyenik font" kampanyası başlatmışım. Amacım bu kasvetli bilgisayar hastalıklarını tedavi edebilmektir. Sonuç olarak sorunlar ortadan kalktı, ancak ayrıntılı önlemlerimin bu sonuçta bir yeri oldu mu, hiç emin olamadım.

Hata Spektrumu

Ciddiyetlerine göre hataları beş farklı kategoride sınıfladım. En kötü olay, tüm bilgisayar sisteminin işlemez bir hal aldığı, çökme (donma ya da bomba olarak da bilinebilir) durumu. Bu durumdan kurtulmak genellikle tekrar önyükleme (reboot) gerektiriyor. Hataların ikinci sınıfı, toptan program hatası. Bu durumda bir program çalışmaktan vazgeçse bile, sistem düzgün çalışmasına devam ediyor. Diğer üç kategoriye büyük program hataları, küçük program ha-

talari ve kozmetik kusurlar adlarini verdim.

Peki bu bu kategorilerdeki hata spektrumunun genisligi ne? GUnluklere baslamadan once, dagilimin depremler, orman yanginlari ya da baska turdeki dogal afetlerinki gibi olacagini tahmin etmistim: Birçok küçük ve az sayıda büyük olay... Tablonun da gösterdigi gibi, gerçek dagilim tam tersi oldu. Çökmeler en sık rastlanan olay, ardından toptan program hatalari geliyor; onun da arkasinda daha az ciddi sorunlar var.

Raporlama önyargim bu spektrumun şeklini degistirmiş olabilir. Muhtemelen kozmetik kusurlari rapor etmede olmam gerektiği kadar dikkatli degildim. Ayrıca benim büyük olarak nitelediğim bazı hatalar, belki de başka bir gözlemci tarafından küçük olarak sınıflandırılabilir. Ancak yine de bu spektrumun ima ettiği en önemli nokta doğru: Bir bilgisayar hata yaparsa, büyük hatalar yapar.

Bu narinliğin nedeni o kadar gizemli değil. Birçok bilgisayar yazılımı ve donanımı en küçük yanlış çalışmayı kaldıramaz. Hatalı bir tek bit, tüm hesabı bozabilir. Bilgisayarlarda da genetikte olduğu gibi, mutasyonlar sadece zararlı değil, öldürücüdür de. Bir program yanlış bir yola saparsa, kurtulma umudu hiç yok gibidir. Narinlik, ayrık durumlara göre çalışan bir dijital mimarinin etkilerini dengeleyebilmek için ödenen ücrettir. Makine ya kursosuzca çalışır ya da hiç çalışmaz.

Bu işin erbabı, istatistiklerimin bellek korumasız, çok görevli bir işletim sistemi (memory protected multitasking) kullanılan bir mikrobilgisayarda elde edildiğinin farkına varacaktır. İşis-tasyonlarında ya da büyük boy bilgisayarlarda, sistem çökmeleri daha enderdir; çünkü işletim sistemi her programı ona ayrılan alanda çalıştırır. Yanlış çalışan bir program sadece kendisini yok edebilir. Bellek koruma teknolojisi kaçınılmaz olarak küçük boy bilgisayarlara da yerleşecektir. Bunun sonucu olarak sistem çökmelerinin sayısı azalacaktır. Bu önemli bir kazanımdır. [Çevirenin Notu: Yazar, makalesini Linux ve Windows95 gibi işletim sistemleri mikro bilgisayarlarda yaygınlık kazanmadan önce yazmıştır. Bu işletim sistemleri, yazarın bahsettiği bellek koruma teknolojisini desteklemektedir.]

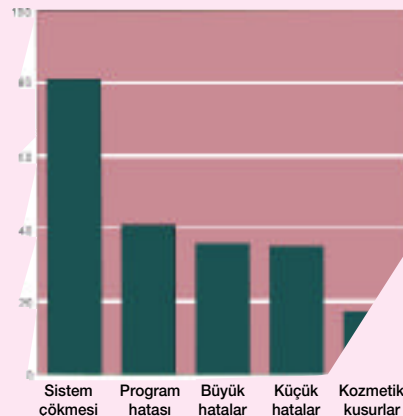


Bizi bekleyen diğer gelişmeler bu kadar umut verici değildir. Bilgisayarlar daha da güçlendikçe, mikrodurumların ve makrodurumların sayısı üstel olarak artar. Bu da, giderek daha fazla sayıda parçanın hata yapması ve bu parçalar arasında karesel (kuadratik) bir etkileşimin olması anlamına gelir. Gelişen paralel işleme, olası hatalara yeni bir boyut kazandırabilir. Belge merkezli hesaplama adı verilen yeni yazılım mimarisi işleri daha da beter duruma sokabilir. Hepsini aynı belge üzerinde çalışan yarım düzineden fazla uygulamanız varsa ve bunlardan birisi belge üzerinde diğerlerinin hoşlanmayacağı bir değişiklik yaparsa ne olacak?

Bu sorunlarla başa çıkmak için daha iyi araçlara gereksinim var. Hata ayıklamada şu ana değin kullanılan en iyi araç programcılar içindi ve ancak kaynak koduna tam erişim sözkonusu olduğunda işe yarıyordu. Bunlar bilgi-

leri bilgisayarın mikrodurumları şeklinde sunduklarından, bilgiyi kullanmanın dünyasındaki olaylarla ilişkilendirmek güçtür. Gereksinim duyulan, makrodurumdaki hatalara tanı koyabilen bir araçtır. Sözelimi, bilgisayarın iki programın aynı donanım kaynağına erişmeye çalışması yüzünden çöktüğünü, ya da art arda üç kez çalıştırmaya çalıştığı programın o anda mevcut olan bellekten daha fazlasına gereksinim duyduğunu söyleyecek bir araç. [Sözkonusu uyarıların bir bölümü Linux, Windows95 gibi işletim sistemlerinde yer almaktadır. ç.n.] Bu gibi bilgiler sayesinde bilgisayarlar şu an oldukları kadar çilgin olmaya devam etseler de, insanlar daha normal kalabileceklerdir.

Böylesine hataya düşebilen bilgisayarlarla birlikte yaşamaya çalışma mücadelesinde en iyi yanıt, bir şekilde karşılaştıkları ilk sorunda çökmeye engel olan donanım ve yazılımlar yaratmaktır. Eninde sonunda, kahve, tost ya da çim biçme makineleri kadar iyi çalışan bilgisayarlar üretmeyi başarabiliriz. Ancak kahve makinesi kadar güvenilir bir bilgisayar üretmek göz korkutucu bir iddia gibi görünüyor. Bu çaba bazı yönlerden daha ünlü olan, insan beyni kadar iyi bilgisayar üretme yarışına benzetilebilir. Başarı ancak uzun vadede gelecektir.



Hayes, B.
"Debugging Myself",
American Scientist, 83:5, 1995
Çeviri: Murat Maga
Resimleyen: Yiğit Özgür