COMMODORE

AYLIK BİLGİSAYAR DERGİSİ Temmuz 1987

Metin işlem hakkında bilmek istediğiniz her şey

Çizgili Forum

DİSKET SÜRÜCÜNÜZÜ TANIYOR MUSUNUZ?

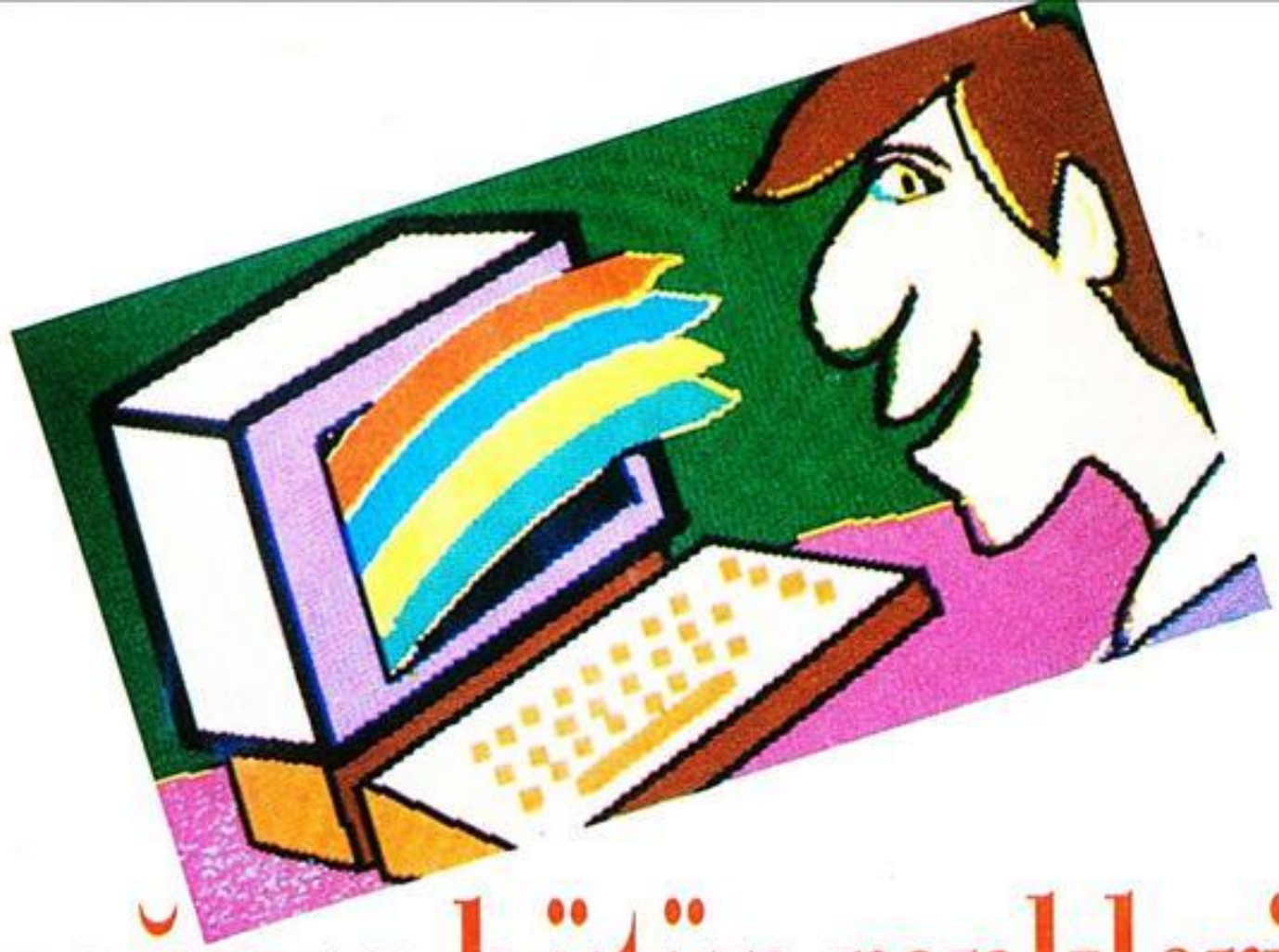
PROBLEMLERİNİZİ PROGRAMLARINIZ ÇÖZSÜN

SAYI 17

1500 TL (KDV Dahil)

Program Döküm Dergisi ekiyle birlikte

yolbulduran-cdemir



Gökkuşağının bütün renkleri UEM Işıklı Kalem'le evinize gelsin!

Şimdi UEM Işıklı Kalem'e sahip olabilmenin çok basit bir yolu var:

Aşağıdaki kuponu doldurun, UEM Uluslararası Elektronik Makine San. ve Tic. Ltd. Şti.'nin aşağıdaki bankalardan birindeki hesabına 10.000 TL yatırın. (Işıklı Kalem'i çalıştıran programı diskette istiyorsanız 11.000 TL.)

Kuponu ya da fotokopisini, banka dekontunun fotokopisiyle birlikte adresimize gönderin. UEM Işıklı Kalemi'niz Acele Posta Servisi'yle adresinize gönderilecektir.

Commodore'unuz varsa, UEM Işıklı Kalem'i mutlaka alın. Ekranda yepyeni "alaimisema"lar yaratın.

Banka Hesap Numaraları:

Türk Ticaret Bankası Site Şubesi	839
Türkiye İş Bankası Osmanbey Şubesi	1683
Yapı Kredi Bankası Osmanbey Şubesi	1877
Akbank Osmanbey Şubesi	3911

UEM IŞIKLI KALEM



UEM 80220 Halaskargazi Cad. Pekin Apt. 210 3 Osmanbey Istanbul
Tel: 147 03 54 - 141 08 77

ULUSLARARASI ELEKTRONİK MAKİNA SAN. ve TİC. LTD. ŞTİ.

UEM Işıklı Kalem İstek Kuponu

Adım, Soyadım .....

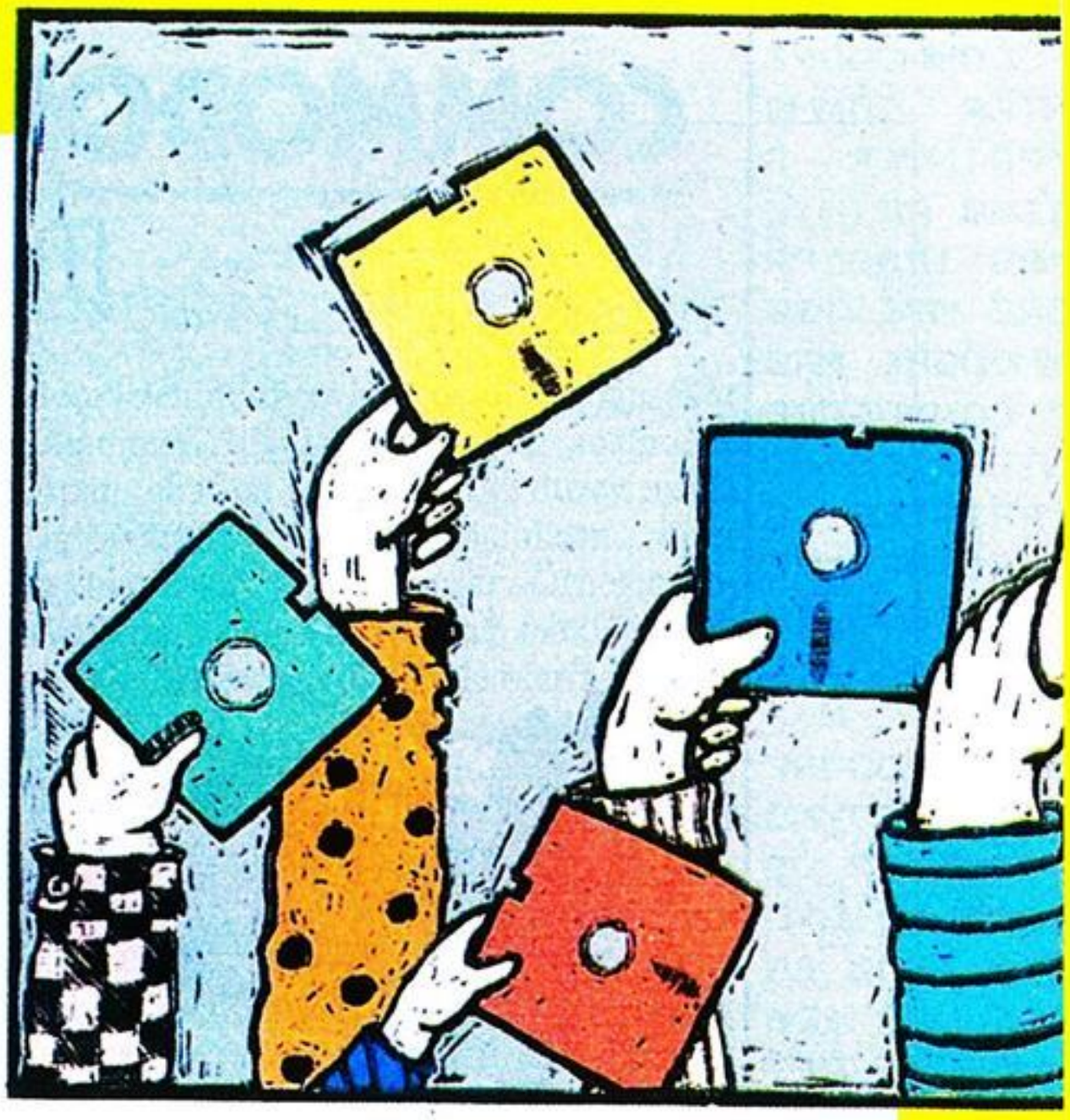
Adresim .....

.....

Kasette (10.000 TL)☐ Diskette (11.000 TL)☐

İmza .....

İçindekiler



TOPLUM

Dünyadan, Türkiye'den, Teleteknik'ten 6
Bilgisayar dünyasından son haberler

Okuyucu Mektupları 8
Sorularınız ve cevapları

Problem Çözem Programlar 10
Bilgisayar dillerindeki gelişmeye değişik bir yaklaşım.

İnsan Düşüncesinde Recursion'un yeri 22
Gödel, Esher ve Bach kitabından son derece önemli bir çeviri.

ÖZEL BÖLÜM

Kelime Denizi 22
Bilgisayarlarda yazı yazmak nasıl bir iş?

Metin İşleme 26
İyi bir metin-işlem sistemi nasıl kurulur?

Metin İşlem'e Giriş 33
PC'lerde Metin-İşlem programları

DONANIM

1541'i Anlamak,, 42
Disket Sürücünüzün Bilmediğiniz yönleri.

Memory Nedir? 48
Bilgisayarınızın hafıza sistemi gözler önünde.

DİLLER

LOGO (1) 56
Logo öğrenmeye başlıyoruz.

Pascal (5) 58
Pascal'da yeni adımlar ve döküm ekinde örnek programlar

Makine Diline Merhaba (7) 63
Makine Dili serüveni sürüyor.

TANITIM

Birbirinden güzel üç yeni oyun: 16
TOMAHAWK, TOPGUN, KRACKOUT

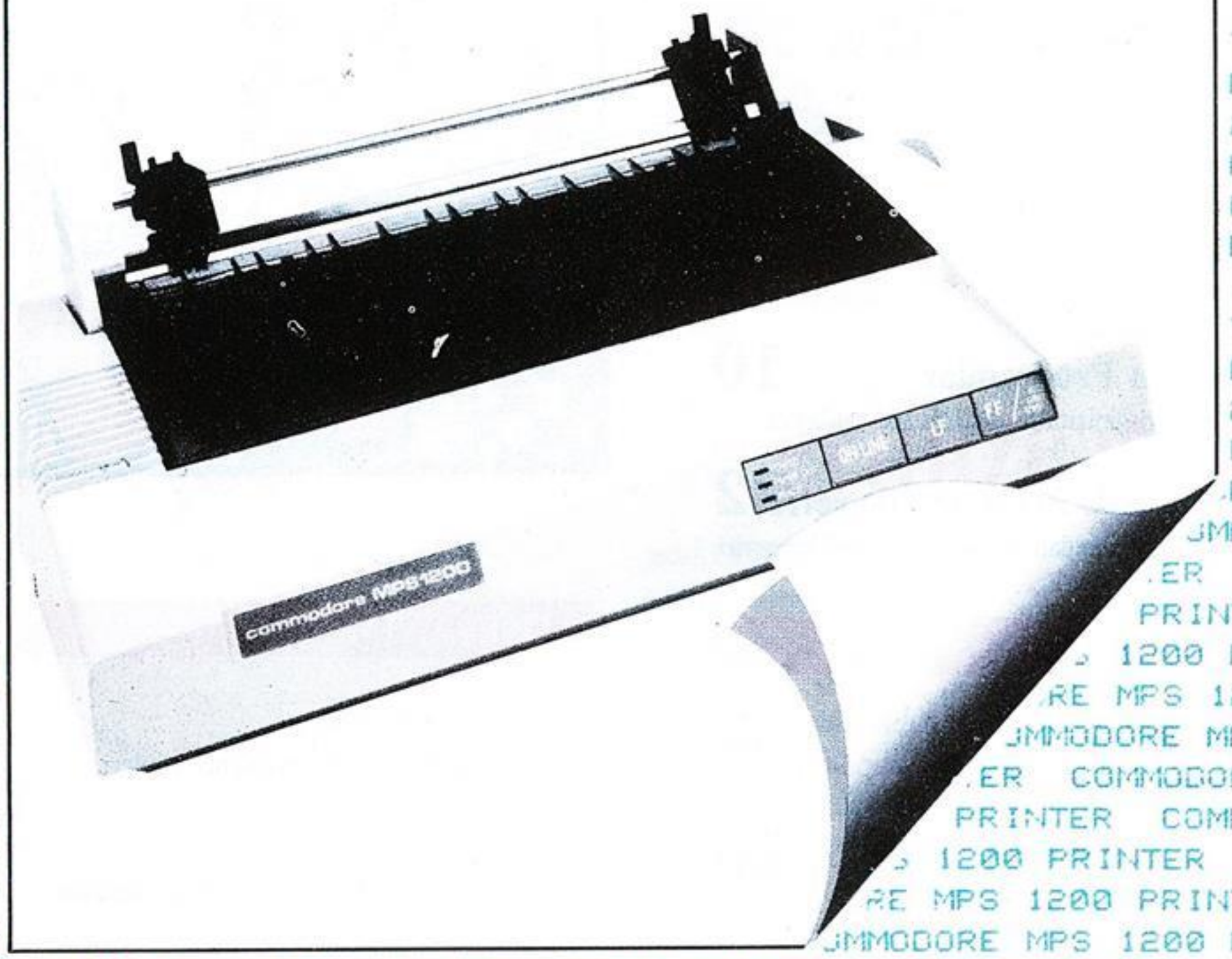
YAZILIM

C-128 İçin Püf Noktaları 52



COMMODORE MPS 1200™

DOT MATRIX PRINTER



BAYİLERDE

Yazma Metodu: *Çift yönlü dot matris / Yazma kafasında 9 vuruş iğnesi* • ***Yazma Hızı:*** *Normal kalitede 120 karakter/saniye / Mektup kalitesinde 24 karakter/saniye* • ***Karakter Seti:*** *Text modunda büyük/küçük karakterler, grafik karakterler, uluslararası karakterler* • ***Karakter Genişliği:*** *5,6,8.5,10,12,17 veya 20 CPI* • ***Kâğıt Genişliği:*** *Tek sayfa 3-10 inç / Sürekli form 8-10 inç / Rulo kâğıt 8.5 inç* • ***Kâğıt Kalınlığı:*** *Maksimum üç kopya (0.06-0.3 mm)* • ***Boyutları:*** *Yükseklik 90 mm / Genişlik 402 mm / Derinlik 255 mm / Ağırlık 8.2 lbs* • ***Güç:*** *Voltaj 220 V / Frekans 50-Hz* • ***Şerit Ömrü:*** *2 milyon karakter* • ***Kafa Ömrü:*** *100 milyon karakter* • ***Karakter Matrisi:*** *Standart 9×9 dot matris / Mektup kalitesi (NLQ)17×17 / Çift vuruş 9×9 / Vurgulu yazma 9×10 / Genişletilmiş yazı 9×19 / Grafik karakterler 8×9*

COMMODORE
AYLIK BİLGİSAYAR DERGİSİ

Teleteknik A.Ş. Adına
Sahibi ve Sorumlu Yazışleri Müdürü
OSMAN BABAOĞLU

Yayın Yönetmeni
İSKENDER SAVAŞIR

Yayın Koordinatörü
TURHAN TANDOĞMUŞ

Danışma Kurulu
BİLAL SADİ OĞUZ
FATİH ARIMAN
ÖMER ERKMEN
EMRE SENAN
ÜMİT KIVANÇ

Yayın Kurulu
KENAN AKKERMAN
TUNA ERTEMALP
MUSTAFA GÜVEN
OSMAN HIZAL
AYHAN KALAYLIOĞLU
SAMET KUMAŞ
TAN ORAL
AFİF SAY
AYŞEGÜL TARIM

Kapak
EMRE SENAN

Teknik Üretim
PERKA A.Ş.

Fotoğraflar
MEHMET ÖZCAN

Reklam ve Halkla İlişkiler
AYSIN AYDEMİR

Reklam Rezervasyonları
145 13 30 (4 HAT)

Katkıda Bulunanlar
BİRGÜL ASENA
BİLGE AYAN
SİLVA AVEDİKOĞLU
SEMRA ÇAĞLAYAN
SEYFETTİN ÇANKAYA
FETHİ SERİM
ANKİN SİNANYAN
AZİZ ŞENLEN
MAHMURE ÜNYELİ

Renk Ayrımı
EBRU GRAFİK

Baskı
ADSAN A.Ş./MERTER

Dağıtım
GAMEDA

Teleteknik A.Ş.
Merkez
RIHTIM CAD. NESLİ HAN NO: 207/4,
80030 KARAKÖY-İSTANBUL
TEL: 145 13 30 (4 HAT)
TELEKS: 25731 TLTK TR
Bilgisayar Danışma Merkezi
SİLAHHANE CAD. RALLİ APT.
59/3 80200 TEŞVİKİYE-İSTANBUL
TEL: 131 15 49-50

YAYINCININ NOTU

METİN-İŞLEM HERKES İÇİNDİR!

Sadece izlenim düzeyinde de olsa, bilgisayar kullanımıyla ilgili olarak bizim gözümüze çarpan tuhaf bir şey var: Farklı düzeylerdeki bilgisayarlar fazlasıyla uzmanlaşmış birer makine olarak algılanıyorlar. Örneğin potansiyel bir PC alıcısı, çoğunlukla alacağı makineyi yalnızca muhasebe ve stok kontrol problemlerine bir çözüm olarak düşünüyor. 64 kullanıcıları ile ilgili olarak ise iki kategori var. Eğer sözkonusu kullanıcının sadece bir dataset'i varsa, makinesinden de sadece bir oyun makinesi olarak yararlanma eğiliminde oluyor. Yok eğer kullanıcı bir disket sürücüden de yararlanma imkânına sahipse, o zaman makinesini çok daha yaratıcı bir biçimde kullanabilmeye başlıyor: Grafik programları ile tanışıyor; müzik programları ile tanışıyor. Ama konumuz bakımından asıl önemli olanı: Metin-İşlem programları ile tanışıyor.

Ama mesele şurada ki, metin-işlem kullanımının sadece disket sürücülü 64 sahipleri ile sınırlı kalması hiç de gerekli değil. Tamam; dataset'li 64'lülerin metin-işlem programlarından uzak durması nisbeten anlaşılır bir şey. Çünkü kasetlere sığabilen metin-işlem programları henüz Türkçeleştirilmemiş durumda (ama bunu gerçekleştiren niye siz olmayasınız?). Oysa PC'ler için durum öyle mi ya?! Bugün PC'ler için geliştirilmiş hepsi de Türkçe konuşabilen birbirinden güzel bir dizi program var.



Bir metin-işlem programı yazım süreci için olduğu kadar düşünce süreci için de yeni ufuklar açan bir alettir. Ne gibi ufuklar mı? İşte bu sayımızın özel bölümündeki yazılar bu soruyu yanıtlamaya çalışıyor.

Bu yazının üslubunu fazla savruk bulabilecek okurlar kusuruma bakmasın. Ayağımın tozuyla yazıyorum. Biraz önce Londra'daki Commodore Show'dan döndüm.

Tahmin edileceği gibi Show'un yıldızı Amiga'ydı. Amiga için geliştirilen ve Show'da sunulan son yenilikler, örneğin Touchscreen ve Desktop Publishing Kit ilerki sayılarımızda ayrıntılı olarak tanıtılacak. Ben de gelecek sayıda Show'la ilgili genel izlenimlerimi sizinle paylaşmaya çalışacağım. O vakte kadar, kalın sağlıcakla... □

OSMAN BABAOĞLU





Görüntü Aktarımında Yeni Bir Boyut

Üç boyutlu bir sayısala çeviriciyi, güçlü bir grafik işliği (grafiğe yatkın mini-bilgisayar) ile kullanarak; üç boyutlu nesneleri sayısal veriye çevirip, üzerinde değişiklik yapmak artık olanaklı. Cyberware Labs tarafından geliştirilen bu sistemde, nesne durağan bir tablanın üzerine yerleştiriliyor. Tablanın çevresini onbeş saniyede dönen bir kol sayesinde, üç boyutta 250.000 ölçüm yapılıyor. İnsan kafası büyüklüğünde nesnelerin yüzeyi, bu ölçme işlemiyle taranarak tanımlanabiliyor.

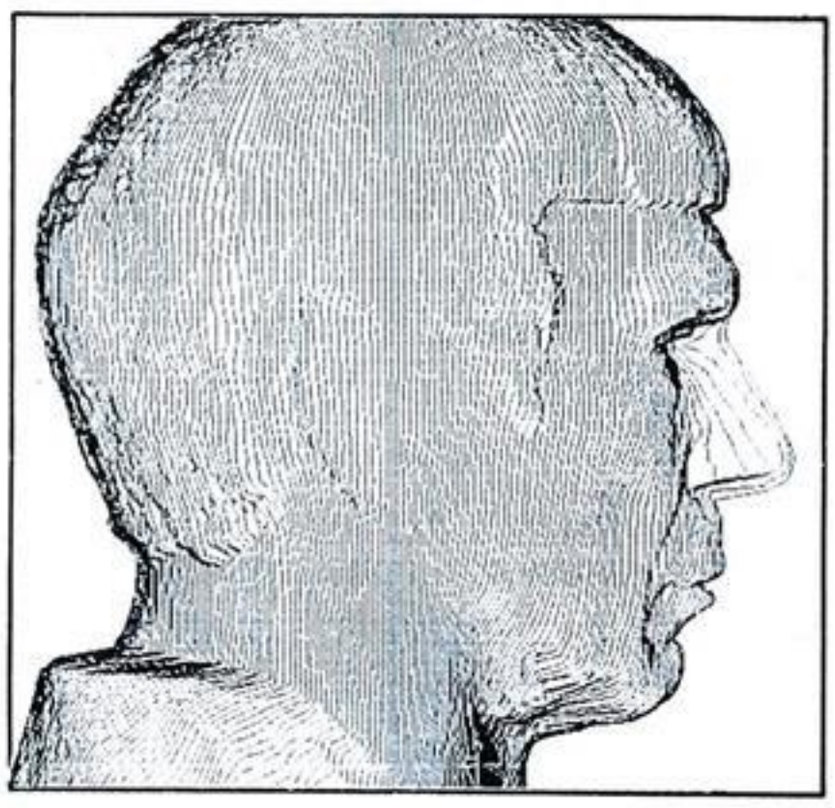
Sistemde, bir lazer ışığı kaynağından çıkan dikey bir ışık çizgisi, nesnenin üzerine düşürülüyor. Kol nesnenin çevresinde dönerken üzerindeki "Vidikon" kamera, ışığı iki değişik açıdan gözlüyor. Kamera optiği öyle düzenlenmiş ki, nesnenin yüzeyinden yansıyan lazer ışını her iki yandan toplanarak kamerada tek bir görüntü oluşturuyor. Böylece gölgeleme sorunu ortadan kalkıyor ve her açıdan ölçme yapılabiliyor.

Yapılan ölçümler bilgisayara iletiliyor. Bilgisayar her noktanın üç boyutta koordinatlarını hesaplayarak sonuçları bellekte belirtiyor. Değişik 512 yatay ve 512 düşey açıya karşılık ölçüm sonuçları 250 bin "byte" olarak diskte saklanıyor.

Depolanan bilgi, bilgisayar ekranında yatay dilimler ya da düşey profiller biçiminde izlenebiliyor. Herhangi bir yükseklikteki yatay dilim ekrana çağırılabiliyor. Düşey profiller ise ekranda anında döndürülerek istenen bir yatay açıdan gözlenebiliyor. Görüntünün tümünün çeşitli izdüşümleri yine ekranda izlenebiliyor ya da istenirse bilgisayara bağlı çiziciyle (poletter) kâğıda aktarılabiliyor. Bu arada mini bilgisayardaki grafik hesap programlarını kullanarak eksik noktaları doldurmak ve yüzeyleri yumuşatmak, seçilen noktaların koordinatlarını okumak olanaklı.

Görüntüyü ekranda üç boyutlu model olarak uzayda herhangi bir açıdan görmek, döndürmek, ışık ve gölgelerin etkisini izlemek için ise daha gelişmiş bir grafik işliği kullanmak gerekiyor.

Sistemin ayrıştırma gücü ve doğruluk derecesi insan kafası-boyutlarındaki nesnelere uygun seçilmiş. Hata payı, 40 cm. yüksekliğinde ve 30 cm. çapında bir nesne için yüzde 1; ayrıştırma sınırı ise her iki eksende 1 mm. olarak veriliyor. Böylece sayılara çevrilmiş olan görüntünün, bilgisayar tarafından kumanda edilen işleme aletleriyle (torna, freze vb.) üç boyutlu modeli de kolayca gerçekleştirilebiliyor. Bu sistemle neler yaratabileceğinizi düşünmek de size kalıyor.





Mr. Spock'un Cyberware'in grafik digitizer'ı kullanılarak üç boyutlu bir görüntü elde edildi (üstteki resim). Silicon Graphics'in IRIS grafik işliği kullanılarak cisimler ışıklandırılmış kaynak modeller aracılığı ile aktarılabilmekte. Görüntüler daha sonra çevrilebiliyor, ayarlanabiliyor, dönüştürülebiliyor ve çıktı alınmak üzere freze makinesine aktarılabiliyor (alttaki resim).

GEOS STORY

San Fransisco'nun kuzeyindeki Berkeley kasabasının çok katlı binalarının birinde yaptığı bir tek program ile tüm dünyada kendinden söz ettiren Berkeley Softworks'un bürosu bulunur. Bu yazıda "C64'ün ikinci işletim sistemi" diye adlandırılan Geos'un arkasında kimlerin bulunduğunu ve nasıl gerçekleştiğini göreceksiniz.

1983 yılının eylül ayında dört programcı ve işadamı Imagic ve Mattel firmalarından ayrılıp, Berkeley Softworks'ü kurdular. İlk zamanlarda firma teknik müşavir olarak siparişleri yerine getirdi. Örneğin, Activision ve Sega için bir portatif bilgisayar yapıldı. Ayrıca PRINTMASTER'ın bazı bölümleri yazıldı. Bu arada, kuruculardа bizzat bir program yazma isteği doğdu ve bu alanda ilk atılım 1985 yazında Chicago'da gerçekleştirildi. Sayıları onbeşe varan elemanlarıyla Berkeley Softworks, C64 için Development Tools adlı yardımcı bir paket hazırladı.

Yine 1985 Ağustosunda Geos projesini gerçekleştirmek için ilk adımlar atıldı. Bu projenin öncüsü olan Brian Dougherty uzun zamandan beri ev bilgisayarları için Macintosh'a benzer bir işletim sistemi planlıyordu. Ekonomik sebeplerden dolayı bu sıralarda eleman sayısı sekiz ile sınırlandırılmıştı.

Bugün Berkeley Softworks'de çalışan 30 kişiden 20'si programcı ve bunların da büyük bir çoğunluğu öğrenci. Bu yüzden programlama birçoğu için bir yan uğraş.

Hazırlanan bir Geos programı üç basamakta test ediliyor. İlk olarak bizzat programcılar, daha sonra ise diğer elemanlar ve dergilerde programı eleştirecek yazarlar test ediyor. Son olarak da normal kullanıcılar davet ediliyor ve tüm bu aşamalardan sonra oldukça hatasız bir program elde ediliyor.

Şu anda birçok yeni Geos programı hazırlanmakta. Bunlardan Geocalc ve Geofile piyasaya çıkmış durumda.

Hazırlanmakta olan bir başka program ise Geochart. Bu program ile verilen bir dizi sayının istatistiki bilgilerini çeşitli şekillerde ekranda göstermek mümkün.

Tüm bu sayılan Geos programlarının özelliği birbirlerini kullanabilmeleridir. Örneğin Geopaint ile hazırlanmış resimler, Geofile ile hazırlanmış bir kart üzerinde bulunabiliyor veya Geofile'a verilen bilgiler Geospell ile kontrol edilebiliyor.

Geos alanında yapılan yenilikler sadece alt-programlarda olmamış. Yeni çıkan Geos 1.3'de işletim sistemi değiştirilmiş ve C64'ün yeni çıkan Ram-Floppy'leri ile çalışma imkânı sağlanmış. C64'e 256 veya 512 Kbyte'lık ek hafıza sağlayan 1750 ve 1764 Ram-Floppy'leri bir kez hafızaya yüklenen bir programı ikinci bir sefer istendiğinde aşağı yukarı 200 kat hızlı geri getirebiliyor. Bu sayede örneğin Geopaint'te ekranın kaydırılması inanılmaz bir hızla gerçekleşiyor.

Görüldüğü gibi Berkeley Softworks yetkilileri C64 için yapılan yenilikleri günü gününe takip ediyorlar ve bunları Geos'a adapte etmeye çalışıyorlar.

Bu nedenle C64'ün yeni 3.5 inç'lik disket sürücüsü 1581 için de bir işletim sistemi hazırlanmaya başlanmış.

Tüm bu yeniliklerin yanında Geos için hazırlanan bir Mouse da piyasaya sürüldü. Bu mouse'un özelliği, üzerinde bir gerçek zaman saati taşıması ve bilgisayara bağlandığında zamanı ve tarihi bildirmesi.



Berkeley Softworks'ün GEOS'u yaratan genç kadrosu toplu halde.

OKUR MEKTUPLARI



Turbosuz çok uzun.

Sayın Yetkili, ben yaklaşık 6 ay önce C-64 sahibi olmuş bir lise öğrencisiyim. O zamandan bu yana bilgisayar hakkındaki bilgim epey fazlalaştı. Fakat Makine Dili bilgim çok yetersiz. Dergiden bu konuyu izliyorum, ancak konular ağır geliyor.

Tesadüfen derginin 1. sayısı elime geçti. Program kopyacısına notlar bölümünde POKE 44,15§ ile programın auto-run etmesini engellendiği belirtilmiş. Bunu yaptım ve POKE 44,8 ile programın asıl listesini aldım. Daha sonra bu oyun programını SAVE ettim, fakat 28 sayaçlık programın SAVE'si 100'ü geçti. Acaba 28 sayaçlık programı yine 28 sayaçlı makine kodunda (diğer oyun programları gibi) SAVE edecek POKE, SYS veya programı var mı? Bunu bana bildirirseniz memnun olurum.

Şu anda Freeze Frame'im yok. Almak istiyorum, ancak çok pahalı. Acaba daha ucuz kopyalama programları var mı? Bunları nereden elde edeceğimizi ve fiyatlarını yazarsanız sevinirim (Disketim yok).

Diğer bir soru Reset Switch'in ne işe yaradığı ve nasıl kullanıldığı? Bu konuda bilgi edinirsem bu aleti alacağım. Fiyatını da yazmanızı istiyorum.

Sorduğum bu soruları cevaplarsanız gerçekten çok mutlu olacağım. Saygı ve sevgilerimle...

TAHİR TEKELİ
Ergenekon Cad. Beyaz Apt.
No: 21/8 Yenişehir/KARABÜK

Sayın Tahir TEKELİ,
Save ettiğiniz program turbosuz olduğundan, sayacın 100 bloğu geçmesi normaldir.

Bir "Turbo-Save" programı ile (derginin 1. sayısında var) kayıt etmeyi deneyin.

Piyasada satışa çıkmış bir kopya programı yok. Ancak Freze-Frame benzeri, kırıcı-kopyalayıcı kartuş programları var. Freze-Frame'den daha ucuz bir kartuş bulabilirsiniz.

Reset-Switch ile çalışan programı kırabilirsiniz. Ancak bu program bilgisayarın hafızasından silinmez. İstediğinizde programı tekrar işler kılabilirsiniz.



Hata Nerede

Sayın Commodore Dergisi, 10. sayınızda yeralan "Infiltrator" adlı oyunun açıklamasını okuyup oyunu aldım, fakat dergideki açıklama tutmadı. Başında bir helikopterin dışında duran adam var, helikopterin içine giremiyorum. Bazı binalara giriyorum, araştırma yapıyorum, fakat sonuç yok. Lütfen bana yol gösterin.

Derginizin buna ilgi göstereceğini sanıyorum. Saygılarımla...

ONUR GÜLFİDAN

Sayın Onur GÜLFİDAN,
Programınız bozuk olabilir. Satın aldığınız yerden sorununuza çözüm bulabilirsiniz.

Program girişinde dikkatli olmalı.

Sayın Teleteknik, ben 17 yaşında, Lise II. sınıf öğrencisiyim. Derginizi çok beğeniyorum. Commodore 64 sahibiyim. derginizin ilk sayısında MDE programı yayınlamışsınız, çok güzel bir şey. Program daha hızlı çalışıyor, kısa programları var. Ama MDE ile ilgili hiçbir program yazamıyorum.

Nedeni şu:

Derginizin I. sayısında MDE yayınlamışsanız ama %100 bir hata var yazdığınız programda, çünkü derginizdeki bu programı 2 defa yazdım, yanlışları düzelttim. Programınız (MDE programınız) çok güzel çalıştı. Şunlar çıktı:

Programın ismi soruluyor, cevapladım. Daha sonra başlangıç adresi soruluyor, burada (0807) olarak yazıp (RETURN) tuşuna basıyorum. Hatalı veri deyip geri çeviriyor. Nedeni nedir? Açıklamaları teker teker okuyup aynısını 50 kere değişik değişik denedim, hiçbir sonuç vermedi. Nedenini bir türlü anlayamıyorum. Acaba yanlış bir şey mi yazıyorum bilgisayara anlamadım. Bana bu konuda açıklayıcı bilgi gönderirseniz memnun olurum.

HÜSEYİN YARIMKAYA
Mehterler Sok. Yeşim Apt. 35/9
Etlik/ANKARA

Sayın Hüseyin YARIMKAYA,
Dergideki program kontrol edilmiştir ve kullanılmaktadır. Girdiğiniz veri de hatalı değil. Programı yazarken bir hata yapmış olabilirsiniz. Kodları bir kez daha kontrol edin. Ayrıca bu programı çalışır durumda dergiden temin edebilirsiniz.

Yeni yıla Modem

SAYIN Commodore Yetkilileri; derginizi ilk sayısından bu yana takip ediyor ve çok beğeniyorum (Özellikle de Program Döküm Eki'ni). Eğer bu dergi olmasaydı Commodore Dünyası'nı bu kadar yakından tanıma olanağı bulamazdık.

Benim sorunuma gelince, Makine Dili konunuzdan bir şey anlamıyorum. Biraz Basic bilgim olmasına rağmen asıl amacım Makine Dili öğrenmek. Ama dediğim gibi Makine Dili yazınızdan bir şey anlamadım.

Sorularımı size şöyle özetleyebilirim:

1- Makine Dili yazısını daha basite indirgeyerek tekrar yayınlayabilir misiniz?

2- Olmazsa bunu nereden öğrenebilirim?

3- Bir bilgisayar kursuna gitmeyi düşünüyorum; kursta Makine Dili öğretiyorlar mı?

4- Bu konu ile ilgili bulabileceğim Türkçe kaynaklar var mı?

Ayrıca merak ettiğim bir başka konu da, Commodore'u telefona bağlayarak sağlanan haberleşme imkânının olup olmadığı. Böyle bir şey varsa beni bu konuda aydınlatırsanız sevinirim. Sizlere şimdiden teşekkür ederim.

CEM AKAOĞLU

SAYIN Cem AKAOĞLU,
Öncelikle dergimize gösterdiğiniz ilgi için teşekkür ederiz.

Makine Dili yazısını daha basit yayınlama isteğinizi maalesef yerine getirmemiz mümkün değil. Ancak kısa bir süre sonra bu konuda kitap yayınlanacağı müjdesini verebiliriz. Ayrıca bu konuda Programcının El Kitabı isimli yayınımızdan ve yabancı dildeki çeşitli yayınlardan yararlanabilirsiniz.

Bilgisayar kursları için doğrudan kurs veren kurumlara başvurmanızda yarar var. Türkçe kaynak olarak Programcının El Kitabını öneriyoruz.

Commodore'un telefon bağlantısı ile haberleşebilmesi MODEM adı verilen araçlar gerektirmektedir. Haber aldığımıza göre PTT bu konu ile ilgilenmektedir ve önümüzdeki yıl içinde MODEM kiralamaya başlayacaktır. Dergimiz de bu konuyla ilgili olarak bir yazı yayınlayacaktır.



B.B.İ.M.D.

SAYIN Yetkililer, kar tatilinden faydalanarak Commodore dergilerini gözden geçirirken ilk sayınızdaki Sayın Baytan Bitirmez'in "Basic Bilenler İçin Makine Dili" adlı yazısını okudum. Bu yazıda verilen makine dili kodlarını ASCII-CHRS koduna çevirerek 49152-49172 adresleri arasına poke'ladım. En son olarak SYS49152 yazıp "return" tuşuna bastım. Bilgisayarım "run/stop" ve "restore" tuşlarına basılmış gibi oldu. Sizden istediğim Sayın Baytan Bitirmez'in verdiği sayıları sizin de bir C-64'e uygulayıp aşağıda verdiğim adrese göndermeniz.

Teşekkürler...

OKAN ŞENGÜL
Faziletli Sokak Dikilitaş Apt. No: 2/1
Bakırköy/İSTANBUL

Sayın Okan ŞENGÜN,
C000 ya da 49152 adresi program çalışırken bilgisayar tarafından kullanılır. Siz bu programı 2000 hex ya da 8192 adresinden itibaren yazın.

2000: AO OO B9 OE C0 20 O2 FF
2008: C8 CO O7 DO F5 60 4D 45
2010: 52 48 41 42 41

Ya da Poke 8192, 160: Poke 8193, 0..... ve SYS 8192

Kısa kilit

Sayın Teleteknik Yetkilileri, size birkaç sorum var. Cevaplarsanız sevinirim.

1) Turbolu programı, uzatmadan turbosuz yapacak bir program var mı, varsa nasıl temin edebilirim?

2) FREZE FRAME teypten teybe çekme yapabilir mi?

3) SYS'li kilit (list için) nasıl yapılır?

Yakın ilginize teşekkür ederim.
Saygılarımla...

BURÇ ÖZER
Kabataş Erkek Lisesi
2 Fen L Sınıfı
Ortaköy/İSTANBUL

Sayın Burç ÖZER,

1- Turbosuz bir programın kısa sürede save edilmesi olanaksızdır. Derginin 1. sayısında bir Turbo Programı yayınlanmıştır. Bu programı denemenizi tavsiye ederiz.

2- Freze-Frame ile bellekteki bir programı kaset veya diskete çekebilirsiniz.

3- Yaptığınız programın ilk satırına, örneğin 0 satırına

0 Poke 775,200

komutunu ekleyin.

Programı çalıştırdıktan sonra list etmeniz olanaksızdır.

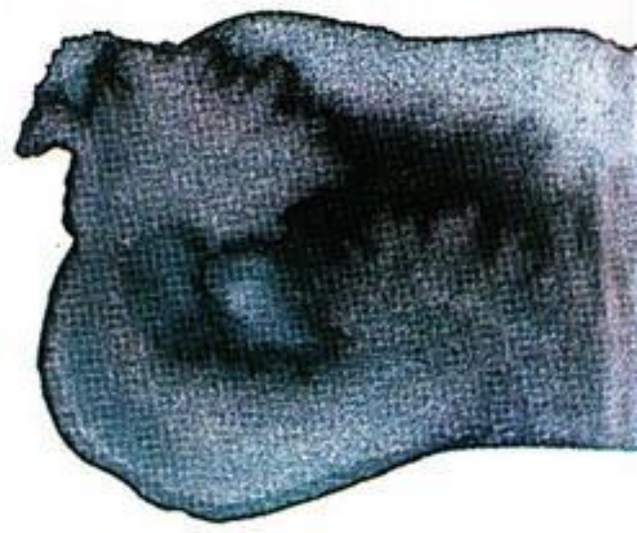
Problem çözen program nasıl olabilir?



M. ALİ KUNTHAN

Psikolojideki öğrenme konusuna bir göz atacak olursak; hayvanların daha çok içgüdüleri ile yaşamlarını sürdürdüklerini, içgüdülerin öğrenilmeden yapılan kalıtsal davranışlar olduğunu, diğer bazı davranışların ise öğrenme yoluyla sonradan kazanıldığını görürüz. Biyolojik açıdan içgüdüler, doğuştan olan ve sinir sistemindeki sinaptik bağlardan oluşan birtakım değişmeyen sinirsel örüntüler olarak açıklanabilir. Öğrenme olgusuna geri dönecek olursak: Öğrenme, organizmanın çevresine uyum sürecinde geliştirdiği bir davranış değişikliği, ihtiyaçları daha iyi karşılayacak biçimde davranışları düzene koyma ya da karşılaşılan yeni ve beklenmedik bir durum karşısında davranışlarda da yeni bir düzenlemeye gitme vb. biçimlerde tanımlanabilir. Öğrenmenin biyolojik temelini ise yine sinir sistemi oluşturmaktadır. Ancak bu kez, içgüdülerden farklı olarak, sinir sistemindeki yeni sinaptik bağların kurulması ile oluşan örgütlenmeler sözkonusu olmaktadır. Sonuç olarak, öğrenme bir tür davranış değişikliğidir. Fakat iki tür davranış değişikliğini birbirinden ayırt etmek gerekir. Bazı davranış değişikliklerini makineler ve ilkel organizmalar da gerçekleştirebilir. Eğer bir makine karşılaşacağı olası durumlarda nasıl bir davranış değişikliğine giderek o durumdaki problemi çözeceği konusunda bilişi ile donatılmış ise, yani bu gibi olası durumlar karşısında nasıl davranacağına önceden programlanmış ise, makinenin karşılaştığı çeşitli durumlara göre değişik alt-programları yürürlüğe koyarak gerçekleştirdiği böylesi davranış değişikliklerini bir tür öğrenme sayamayız. Çünkü bu değişikliklerin kendileri de, kapsayıcı bir programın parçası olmak bakımından önceden belirlenmişlerdir. Bu nedenle bu tür durumlarda, sistemin davranışını gerçek anlamda değiştirdiğini söylemek doğru olmayacaktır. Tek ve değişmez bir davranış biçimi vardır; bu da alt-davranış biçimlerini yöneten ana programın kendisidir.

Gerçek anlamda bir davranış değişikliği, yani "öğrenme" hatta "buluş" olarak nitelenebilecek bir davranış değişikliği, nasıl karşılık verileceği önceden belirlenmemiş bir durum karşısında organizmanın o duruma uyum yapabilecek biçimde davranışlarını değiştirebilmesidir. Başka bir deyişle, organizmanın (makinenin ya da bilgisayarın), beklenmedik bir durum karşısında *nasıl* davranacağı konusunda önceden hiçbir bilişiye sahip olmadığı halde, karşılaştığı problemi çözebilmesidir. Ancak sistemin bu gibi durumlara ilişkin *hiçbir* programı yoksa, sistemden herhangi bir çözüm arama çabası beklemek mümkün değildir. Sistemin genel anlamda bir "problem çözen programı" elbette olmalıdır. Fakat bu genel problem çözücü program, *belirli* bir problemin çözümüne ilişkin olmamalı; *herhangi* bir problemin, *herhangi* bir biçimdeki çözümüne yönelik olmalıdır. Gerçek anlamda öğrenmeden ya da bir problemin çözümünü "bulmak"tan, bunun gibi genel bir programın işleyişini anlamak gerekir.



Öğrenmek... Bilgisayarcılar için büyülü bir sözcük bu. Öğrenen ve öğrendiğini kullanan bilgisayarlar çağın ilk hedefi.

Şimdi bu anlattıklarımızı bir örnek üzerinde inceleyelim:

Yandaki şekil farelerin denek olarak kullanıldığı basit bir dolambacı (labirenti) göstersin. A ucundan içeri bırakılan fareler önce dolambacın içinde gelişigüzel dolaşıyorlar. B noktasına geldiklerinde, bir karar vererek sağa, sola ya da geriye dönmek seçeneklerinden birisini seçmeleri mümkün. Sola dönmüşlerse, D ucuna vardıklarında bu ucun kapalı olduğunu görecekler ve geri dönmekten başka seçenekleri kalmayacak. Sağa dönmüşlerse, açık olan C ucundan dışarı çıkabilecekler ve böylece dolambaçtan kurtulmuş olacaklar. "Dolambaçtan kurtulmak" bile bir bakıma başlı başına bir ödül sayılabilir. Fakat aç, yiyecek bulmaya güdülenmiş bir farenin, C ucundan çıktıktan sonra bir yiyecek bulması ve böylece fizyolojik bir gereksinimini karşılaması kuşkusuz çok daha değerli bir ödüldür. Böylesi güdülenme ve ödüllenmelerin öğrenme adı verilen olgu için önemli etkenler olduğunu deneysel psikoloji göstermektedir. Farenin dolambacı "öğrenmesi" adını verdiğimiz bu olgu şöyle gelişiyor: Aynı fare, birçok kez aynı dolambaca A ucundan bırakıldığında, ilk kez yaptığı gibi sınama-yanılma yoluyla, ikinci kez ve daha sonraki denemelerde de çıkışı buluyor. Fakat artan sayıdaki deneyimler gösteriyor ki, farenin çıkışa varmak için yaptığı çabalamaların sayısı giderek azalmaktadır. Sonunda öyle bir duruma geliniyor ki, artık fare her denemede, hiçbir sapma yapmadan en kısa yol olan A-B-C yolunu izleyerek ödülüne ulaşabiliyor. Sonuç olarak, farenin gelişigüzel davranışlar arasından karşısındaki problemi çözmeye yönelik en uygun davranışlar dizisini belirleyerek yeni bir davranış geliştirmiş olduğunu, başka bir deyişle, farenin dolambacın çözümünü öğrendiğini söyleyebiliyoruz.

Bu örneği asıl amacımız açısından önemli olan bilgisayar programları ile ilişkisini kurarak incelemeye devam edelim: Örneğin bilgisayar oyunlarındaki gibi "T" biçiminde düzenlenmiş bir dolambaç ve bu dolambacın içinde bir "sprite"ı gelişigüzel hareket ettiren bir bilgisayar programımız olsun. (Bu tür görsel oyunlara aslında gerek yok. Bunlar problemin sunuluşunu kolaylaştırmaya yarıyorlar.) Bu programa sprite'ın ya da farenin 'fizyolojik programı' "F" adını verelim. Fizyolojik program hakkında şunlar söylenebilir:

1. Nasıl ki fare, duyu organları, sinir sistemi ve diğer uzuvları ile hareket halinde olup ileri, geri, sağa, sola dönmek gibi eylemleri otomatik olarak yapabiliyorsa, bilgisayar programımız da sprite'ı sürekli hareket ettirmeli ve random sayılarla işleyen bir karar mekanizması yardımı ile B noktasına geldiğinde sağa, sola veya geri dönmek, D noktasına geldiğinde ise sadece geri dönmek gibi seçeneklerin seçimini gerçekleştirmelidir. Bu hareketler, aç bir farenin yiyecek arama çabasına benzetilebilir; fakat bu genel çabalamalar belirli bir problemin belirli bir biçimde çözümüne yönelik değildir. Sprite'ın (farenin) çıkışa varması tümüyle rastlansal olarak gerçekleşmektedir. Bu arada, sprite'ın hızının değişmediğini varsayıyoruz.

2. Farenin "acıkması", "açlıktan ölmesi" ve "açlığını gidermesi" gibi olgulara karşılık olmak üzere bilgisayar programımızın da "mide"si diyebileceğimiz bir alt-programı olsun. Bu programın çalışması şöyle olabilir: Fare dolambaca girdikten sonra çalışmaya başlayan bilgisayarın iç-saati bu amaç için kullanılabilir. Saatin alabileceği en büyük değer Tm olsun. Bu Tm değeri, yukarıda anlattığımız random sayılarla işleyen programın çıkışa ulaşabilmesine *ortalama olarak* imkan verecek kadar büyük seçilmelidir. Burada bir "ortalama değer"den söz etmek gerekir; çünkü "F" programı random sayılarla işlediğinden salt rastlansal olarak en kısa yolu izleyerek çıkışa ulaşmış olabileceği gibi, gene salt rastlansal olarak çok uzun bir süre geçtiği halde çıkışa ulaşamamış olabilir. Şimdi, ortalama varış zamanı olan Tm'den daha uzun bir süre geçtiği halde sprite çıkışa varamamışsa bunun anlamı "açlıktan ölmek" olsun, Tm kadar veya daha kısa bir sürede çıkışa vardığında "yiyeceğe ulaşmış", yani "açlığını gidermiş" sayılsın ve saatin değeri sıfırlansın.

3. Sistemin, geçireceği deneyimle-



rin kaydını tutan bir de belleği olsun. Bilgisayarların belleği bu amaç için yeterlidir.

Buraya kadar "öğrenme" ya da "çözüm bulma" ile ilgili herhangi bir şeyden söz etmedik. Aynı program yürürlükte olduğu sürece benzer olaylar tekrarlanıp duracaktır. Böyle bir fizyolojik programla sistemin dolambacın çözümünü bulması mümkün değildir. Bu noktada şöyle bir programımızın daha olduğunu varsayalım: Bu yeni programa "Çözüm Öneren Program-M" adını verelim. Varsayalım ki bu programın şu işlevleri olsun:

1. "M" programı "F" programının içindeki herhangi bir satırı tümüyle iptal edebilsin.

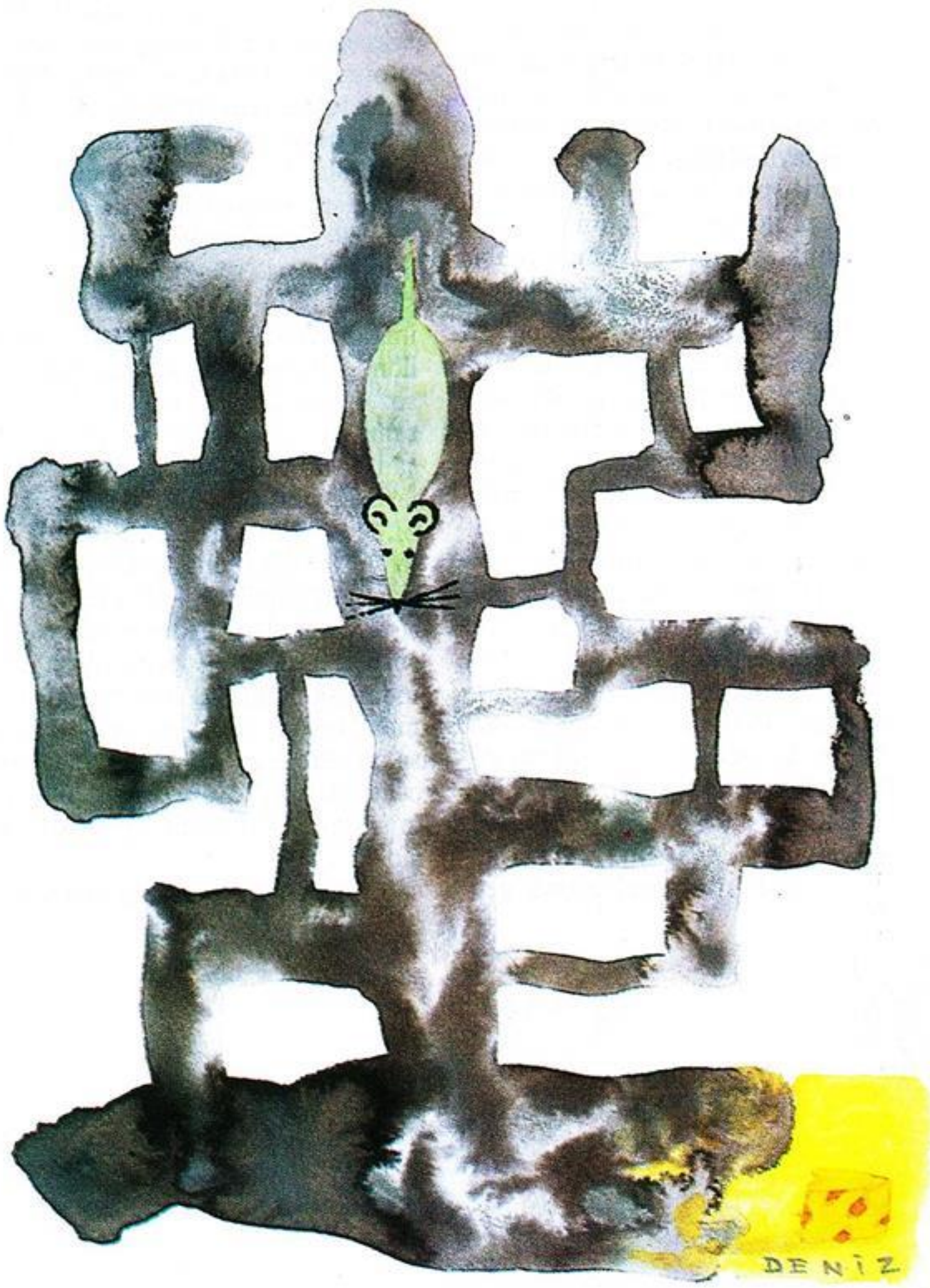
2. "M" programı "F" programının içindeki bir satırı oluşturan çeşitli komutlar, işlemler, semboller, parametreler vb. üzerinde değişiklikler yapabilsin.

3. "M" programı "F" programının satır no. vb. üzerinde değişiklikler yaparak "F"nin akış şemasını değiştirebilsin.

4. "M" programı "F" programına gelişigüzel oluşturulmuş bir satır ekleyebilsin.

"M" programı, bu değişiklik türlerinden hangisini, nasıl gerçekleştireceğine tümüyle random sayılarla işleyen bir mekanizma ile karar vermektedir; fakat bu değişiklikler bilgisayarın kullandığı dilin sentaksına uygun olmak zorundadır. Bu nedenle "M"nin ürettiği her değişiklik, yani her çözüm önerisi, hemen "F" üzerine uygulanmamalı, önce değişikliği sentaks yönünden inceleyerek uygun olup olmadığına karar veren bir "denetleme" programının onayından geçirilmelidir.

Önerdiğimiz "M" programının ürettiği değişiklikler ile biyolojik sistemlerin evriminde önemli bir rol oynayan "mutasyonlar" arasında sıkı bir benzerlik vardır. Canlılarda genetik yapı her yeni kuşaklara aktarılışı sırasında mutasyonlar vb. nedenlerle değişime uğrar. Bu değişimlerden bazıları organizmanın çevresine daha iyi uyum yapmasını sağlarken kendi geçerliliklerini de onaylatmış olurlar. Yani, gelecek kuşaklara aktarılma şansını elde ederek türün bir özelliği olmaya hak kazanmış olurlar. Olumsuz yönde ortaya çıkan değişimler ise, eğer bireyin ölümüne neden olacak kadar önemli iseler, daha işin başında, bireyin ölümü ile birlikte eleneceklerdir. Fakat değişimler genellikle bu iki türden de değildirler. Ne sistemin çevresine daha iyi uyum yapmasını sağlarlar, ne de sistemin ölümüne neden olacak kadar kötüdürler. Bu üçüncü tür değişimler türün evriminde çok önemli bir görev yaparlar. Bunların varlığı sayesinde belli bir oluşuma giden yolların sayısı artar ve hemen her türlü çözüm yolunun denenebilmesine olanak sağlanır. Aynı konuya aşağıdaki "c" maddesini incelerken tekrar değineceğiz.



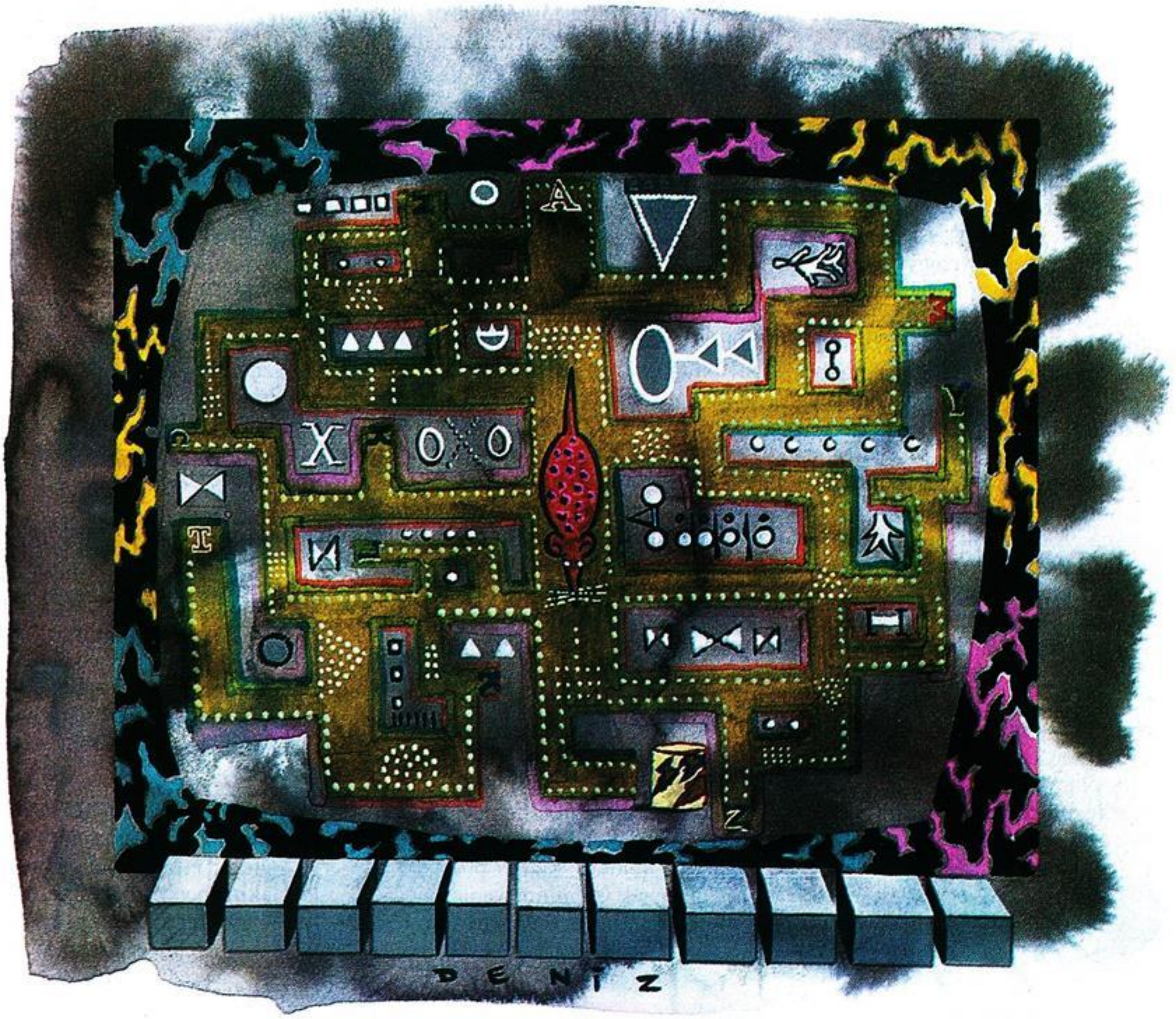
Problem çözmek bilgi ve deneyim birikimi gerektirir. Öncelikle bu oluşturulmalıdır.

Problem çözme metodumuzun işleyişi de buna benzer bir yoldan olacak. Metodun işleyişini şematik bir biçimde inceleyelim:

1. Rasgele (random) hareketlerle sprite'ı çıkış noktasına Tm veya daha kısa sürede (ortalama olarak) ulaştırabilen bir başlangıç programı F1 olsun.

2. Çözüm öneren program tarafından üretilmiş ve sentaksa uygunluğu onaylanmış olan bir değişiklik F1 üzerine uygulanarak F1, F2 gibi bir programa dönüştürülsün. Fakat F1 tümüyle yitirilmesin ve onun bozulmamış ilk biçimi bellekte saklansın.

3. Yürütme organları tarafından F2



yürürlüğe konsun. Başka bir deyişle, F2 işletilerek onun yeterli bir sürede çıkışa ulaşıp ulaşamadığı ve *ortalama olarak* ne kadar sürede ulaşabildiği belirlensin. Karşımıza şöyle durumlar çıkabilir:

a) Eğer F2'nin ortalama varış süresi Tm'yi aşmışsa (fare açlıktan ölmüşse), bu başarısızlık F1 üzerindeki değişiklikten kaynaklanmıştır, o halde bu mutasyon olumsuz bir mutasyondur. F2 iptal edilip, başka bir değişiklik denenmek üzere F1'e geri dönülmelidir.

b) Eğer F2'nin ortalama varış süresi F1'den daha kısa ise (genel olarak: Fn'nin ortalama varış süresi Fn-1'den daha kısa ise), bu son program öncekinden daha iyi bir program demektir. O halde yapılmış olan değişiklik onaylanabilir. Onaylama işlemi, bellekteki mevcut programın yerine son programın geçirilmesidir.

c) Değişikliklerin bazıları ise ne "a"daki gibi Tm'den daha uzun bir sürede, ne de "b"deki gibi daha önce yürürlüğe konmuş en iyi programdan daha kısa bir sürede varış noktasına ulaşabilmişlerdir. Bu tür programlar başarısız bir program olarak nitelenerek iptal edilmemeli, başarılı bir program olarak ana belleğe de aktarılmamalı; fakat bunlar üzerinde ikincil değişikliklerin yürütülmesine devam edilmelidir. Bu tür değişiklikler, problemin çözümüne giden yolun tıkanmaması ve hemen her olası çözümün denenebilmesini sağlamak için gereklidirler. Eğer böylesi bir esneklik getirilmemiş olsaydı, en son onaylanmış programdan daha başarılı bir programı bir tek değişiklik ile elde etmenin olanaklı olmaması durumunda sistemin işleyişi tıkanacaktı. Çok iyi olmayan programlara da geri dönülmesi ve bunlar üzerinde de değişikliklerin denenmesine olanak tanınması, inişli çıkışlı bir süreç izlenmesine yol açmakla birlikte, çözüm arama yollarını zenginleştirerek er veya geç en iyi çözümün bulunabilmesini güvence altına almaktadır.

Bu son anlattıklarımız şu tür bir eleştirinin ileri sürülmesini de engellemektedir: Örneğin F1 üzerinde yapılan ilk değişiklik olumsuz olduğu halde, F2 iptal edilmeyip onun da üzerinde bir veya birkaç değişiklik yapılmasına izin verilse idi, elde edilecek olan Fm gibi bir program problemin çözümüne F1'den çok daha uygun olabilirdi. Oysa yukarda anlatılan metod yalnızca bir tek değişikliğe izin vermektedir. Bu doğrudur, fakat amacımız açısından Fm gibi bir programa hangi değişiklikler dizisini izleyerek ulaştığımız önemli değildir. Fm gibi bir programa her denemede yalnızca bir tek değişikliği değerlendirerek pek çok değişik yoldan ulaşabilme olanağımız gene vardır ve bu olanak "c"de anlatılan, "çok iyi

olmayan" programların da üzerinde değişikliklerin yapılması yoluyla sağlanmaktadır.

Benzer düşünceler "M" programının ürettiği değişiklikler için de ileri sürülebilir. "F"ler üzerinde yapılmak istenen bir tek değişiklik sentaksa uygun olmayabileceği halde, her biri teker teker ele alındığında sentaksa uygun olmayan değişikliklerin uygun bir dizisi sentaksa uygun olabilir. Burada da sözkonusu değişikliğe pek çok yoldan ulaşabilmenin olanaklı olduğunu ve "M" programının içinde buna uygun düzenlemelerin yapılmış olduğunu varsaymakla yetinelim.

4. Yukarıda anlattığımız süreç, en kısa yoldan çıkışa varan program elde edilinceye kadar ya da en azından oldukça iyi olarak nitelenebilecek bir çözüm yolunun bulunuşuna kadar sürdürülebilir.

Bu anlattığımız metoda genel ola rak şöyle eleştiriler yöneltilebilir:

Yapılabilecek olan olası değişikliklerin ve bunların çeşitli kombinasyonlarının sayısının sonsuz olabileceği ileri sürülerek en kısa yolu bulmanın pratik olarak olanaksız olduğu söylenebilir. Bu eleştiri geçerlidir. Fakat *tam* çözümün bulunması çok uzun bir süreyi gerektirdiği için olanaksız olsa bile, *oldukça iyi* çözümlerin gerçekleştirilebilmesi beklenebilir. Çoğu kez, çok karmaşık labirentlerin içinde farelerin, hattâ insanların da yapabileceği böyle nisbeten iyi çözümlerden ibarettir. Üstelik pek çok seçeneği kısa bir sürede sınamak ve bıkmadan uzun süre çalışmak gibi niteliklere sahip olan bilgisayarlar tam da bu tür çözüm yolları için geliştirilmiş en uygun araçlardır.

Örnekteki gibi bir dolambacın çözümünün, şeklin geometrik özellikleri gözönüne alındığında hemen bulunabileceği ileri sürülerek anlattığımız metodun gereksiz bir zaman kaybı olduğu da söylenebilir. Oysa biz burada sistemin daha önceden karşılaştığı durumlara hiç benzemeyen yeni ve bilinmeyen durumlardaki davranışları ile ilgilenmekte ve bu gibi durumlar için genel bir çözüm yolunun nasıl olabileceğini tartışmaktayız. Dolambacın çözümünü bir bakışta çözmek dolambaç problemine benzer bir durumun daha önceden çözülmüş olmasını ya da en azından bu gibi durumlarla önceden pek çok kez karşılaşılmış olmasını gerektirir. Eski deneyim ve alışkanlıkların pek bir işe yaramadığı durumlarda ise, insanların da çoğu kez yaptıkları gibi, denenebilecek tek geçerli yol, anlattığımız metoda benzer bir yoldur. "Kanser problemi" gibi karmaşık ve bilinmeyenlerle dolu problemlerin çözümünde, tüm bilimsel bilgi ve deneyim birikimine rağmen problemin çözümüne götürecek olan belli başlı aşamaların gerçekleştirilmesi için yapılagelen çalışmalar ve buna ilişkin "bilimsel bulma metodunun" özü, yukarıda anlattıklarımızdan pek farklı değildir.

Bir problemin çözümünde, problemi oluşturan ögelerin niteliklerini ve ögeler arasındaki ilişkileri sezinleme, analoji vb. yollardan faydalanma gibi etkinliklerin de önemli olduğunu yadsımıyoruz. Fakat bütün bunlar, ancak belli bir bilgi ve deneyim birikiminden sonra etkili olabilirler. Bizim üzerinde durduğumuz metod ise, sistemin deneysel birikiminin *içeriğinin* oluşturulmasını sağlayacak olan metoddur.

Hattâ belli bir bilgi ve deneyim birikiminden sonra bile gerek bilgisayarlarda, gerek insanlarda önerdiğimiz metoda benzer bir biçimde, çeşitli olası durumların sınanmasına dayanan, bedensel, zihinsel ve çoğu kez de bilinç altından yürütülen pek çok sayıdaki sınama-yanılma deneyimlerinin bulunduğunu gösteren olgular vardır.

Anlattığımız metodun her türlü öğrenme, bulma ve dolayısıyla bilginin temelini teşkil ettiği kabul edilirse, insanlardakine benzer öğrenme, bulma ve bilme gibi olguları bilgisayarların da gerçekleştirebileceği söylenebilir.

BİL-MER BİLGİ İŞLEM

SÜPER OYUNLARIMIZ
(Kaset ve Diskette 500 TL.)

1) Burnin Rubber
2) Gods Heros
3) N.o.m.a.d (nvl)
4) Seabase Delta
5) F.ball of Year
6) He-man Joystick
7) Jammin
8) Scalextric
9) Koko
10) Kapriolen
11) Imhotep
12) Short Circuit
13) Xeious/Ics
14) The Madness
15) Legions of Death
16) Rocin Westle
17) Reption 3
18) Touch me
19) Marble Madn.

EN UCUZA BİZDE!
COMMODORE 64 VE İLGİLİ TÜM ÜRÜNLER VE TELEVİZYONLARI BİZDEN TEMİN EDEBİLİRSİNİZ.

FİYATLARIMIZ

20 oyun 1. kasette 1900 TL.
20 oyun 1. kasette 2500 TL.
20 oyun 1. kasette 3000 TL.
20 oyun 1. kasette 3500 TL.
20 oyun 1. kasette 4000 TL.
Makine tipi Joystick 7700 TL.

SUPER COPY PAKETLERİ
KASET DİSKET 1000-10.000
SUPER PAKETLER Serisini
birden alana 3. paket bedava

ÖDEMELİ OLARAK GÖNDERİLİR
Hesap Numaramız:
Türkiye Halk Bankası 5945 B.köy/İST.

BİL-MER BİLGİ İŞLEM

Halkçı Sok. No:33/7 Bakırköy İş Hanı ve Çarşısı Bakırköy/İstanbul Tel: 570 34 90

concord

ELEKTRONİK - PROGRAMLAMA

Commodore Bilgisayarları ve Yan Ürünleri Satış Merkezi

- Eğitim programları
- İş programları
- En son oyun programları
- Her marka ithal joystickler.

Commodore 64 Oyun Programları Satış Listemiz

Kasete/diskete oyun çekimi	500 TL.
Tek kasetlik süper oyun	1000 TL.
Tek disketlik süper oyun	2000 TL.
20'li,30'lu,40'lı oyun kasetleri	3500 TL.
Etiketli, kapaklı boş kaset	600 TL.
Çift taraflı ithal disketler	1500 TL.
CONCORD Reset Switch	1500 TL.

Ayrıca iki tane paket oyun alana, veya 7000 TL.lık oyun alışverişi yapana bir CONCORD Reset Switch bedava...

Concord Elektronik Programlama:
İstiklâl Cad. (Odakule Karşısı) Beyoğlu İşmerkezi Ortaçarşı No: 65, Beyoğlu-İSTANBUL Tel: 151 31 23

BİLGİSAYARLARIN DÜNYASINDA YAŞAMAYA HAZIR MISINIZ?

Bilgisayar alacaksınız! Hangi bilgisayar neyi, ne kadar yapar? Karşılaştırmak, en uygun olanı seçmek için bilmeniz gereken her şey BİLGİSAYAR ANSİKLOPEDİSİ'nde var.

Bilgisayarınız var! En iyi nasıl kullanacaksınız? Daha iyi nasıl yararlanacaksınız? Temel bilgilerden başlayarak öğrenmek, öğrendiklerinizi uygulamak için BİLGİSAYAR ANSİKLOPEDİSİ okuyun.

Bilgisayarla çalışıyorsunuz! Daha iyi ve daha etkin programları nasıl yazacaksınız, işinize nasıl uyarlayacaksınız? Basic, Fortran, Cobol, Pascal, Assembler veya Logo dillerini öğrenmek için BİLGİSAYAR ANSİKLOPEDİSİ alın.

Gelecekle ilgileniyorsunuz! Bilgisayar çağında ne olacak? İnsanlar nasıl yaşayacak, ne yapacak? Bilgisayarların toplumsal düzenine hazırlanmak için BİLGİSAYAR ANSİKLOPEDİSİ'ne başvurun.



7 cilt, 2008 sayfa, sayısız şema ve tablo, ofset baskı, 1. hamur kâğıt, lüks cilt.

Kullanılan terimlerin Almanca, İngilizce, Fransızca karşılıkları.

Commodore Kulüp üyelerine % 10 indirimlidir

7 ciltte 6 Bilgisayar dili öğretiyoruz.

© Armando Curcio Editore, İletişim Yayınları.

T.C. Milli Eğitim, Gençlik ve Spor Bakanlığı tarafından okullara tavsiye edilmiştir.

		Peşin	TAKSİTLİ SATIŞ 4 ay	8 ay
BİLGİSAYAR ANSİKLOPEDİSİ (7 CİLT)	TOPLAM FİYAT	69.300	77.950	86.650
	PEŞİNAT		22.750	20.250
	TAKSİT		4×13.800	8×8.300

Adım: .....
Soyadım: .....
Adresim: .....
Telefonum: .....

Yazışma Adresi: İletişim Pazarlama-Klodfarer Cad. No: 7 34400 Cağaloğlu/İST.
Tlf: 512 17 32-33

BİLGİSAYAR
ÇAĞDAŞ TEKNOLOJİ
ANSİKLOPEDİSİ

İletişim Çağında Bilgisayar;
Bilgisayar Çağında İletişim

İletişim Yayınları • İletişim Yayınları • İletişim Yayınları • İletişim Yayınları • İletişim Yayınları • İletişim Yay

TANITIM



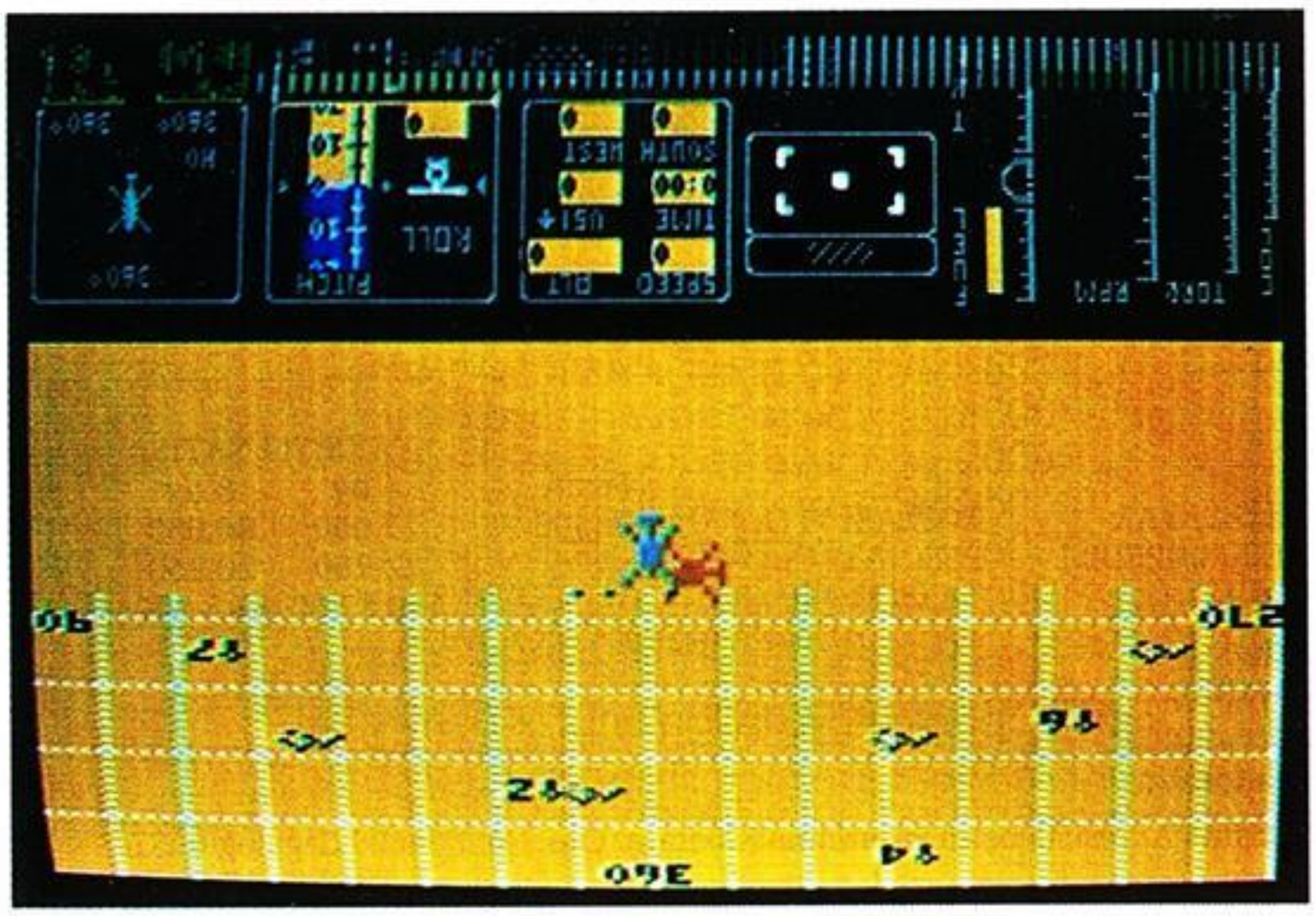
TOMAHAWK

TOMAHAWK şimdiye kadar yapılmış helikopter simulasyonları içinde en iyilerinden birisi. Simulasyonda kullanılan helikopter Amerikan ordusunda bulunan AM 64A APACHE saldırı helikopteri. Bu helikopter korkunç özelliklere sahip. Saatte 180 milin üstünde bir hız yapabilmekte ve çok öldürücü silahlar taşıyabilmekte.

Program çalıştırıldığında karşımıza çıkan menüde seçilebilecek dört ana seçenek mevcut. İlk olarak hangi görev bölgesinde çalışacağımızı seçeceğiz. Daha sonra oyundaki istediğimiz zorluk derecesini belirleyeceğiz. Uçuş alıştırmasını seçerek vurulma korkusu olmadan oyunu sürdürebilirsiniz. Bu fırsattan yararlanarak ilk başlarda helikopterinizin hakimiyetine ve mevcut olan çeşitli kontrollere alışabilirsiniz. Menüdeki diğer seçenekleri kullanarak uçuşunuzun bulutlu veya bulutsuz bir havada olmasını ve uçuş zamanının gece veya gündüz oluşunu ayarlayabilirsiniz. Hatta bulutlu havayı seçerseniz bulut tabakasının hangi yükseklikte başlaması gerektiğini de ayarlamanız mümkün. Ama bütün bunlar yetmiyorsa işinizi daha da zorlaştırmak için çapraz rüzgâr ve turbulans akımlar da ekleyebilirsiniz. Bütün ayarlamaları bitirdiğinizde menünün sonuna gelip ateş tuşuna basarak oyuna geçin.

Şu anda helikopterin içinde bulunuyorsunuz. Tavsiyemiz havalanmaya başlamadan evvel elinizin altındaki aletlere şöyle bir göz atmanız. Ekranın üst kısmı o anki seçiminize göre ya helikopterin camından görünen bölgeyi ya da nerede olduğunuzu belirten haritayı gösterir. Eğer önünüzde harita bulunuyorsa haritada sizin bulunduğunuz yerden başka, size en yakın düşman helikopterin yeri de belirtilir. Bu açıdan şanslı sayılırsınız, çünkü savaş planınızı hazırlarken düşman helikopterinin yerini bilmek size büyük bir avantaj sağlayabilir.

Ekranın alt kısmında ise çok güzel olarak hazırlanmış, okunması ve yorumlaması gayet kolay alet panelleri bulunuyor. Bu kısmın sol tarafında motor, pervane ve tork aletlerinin durumu gösteriliyor. Bunların hemen sağında hedef panosu ve helikopteri-

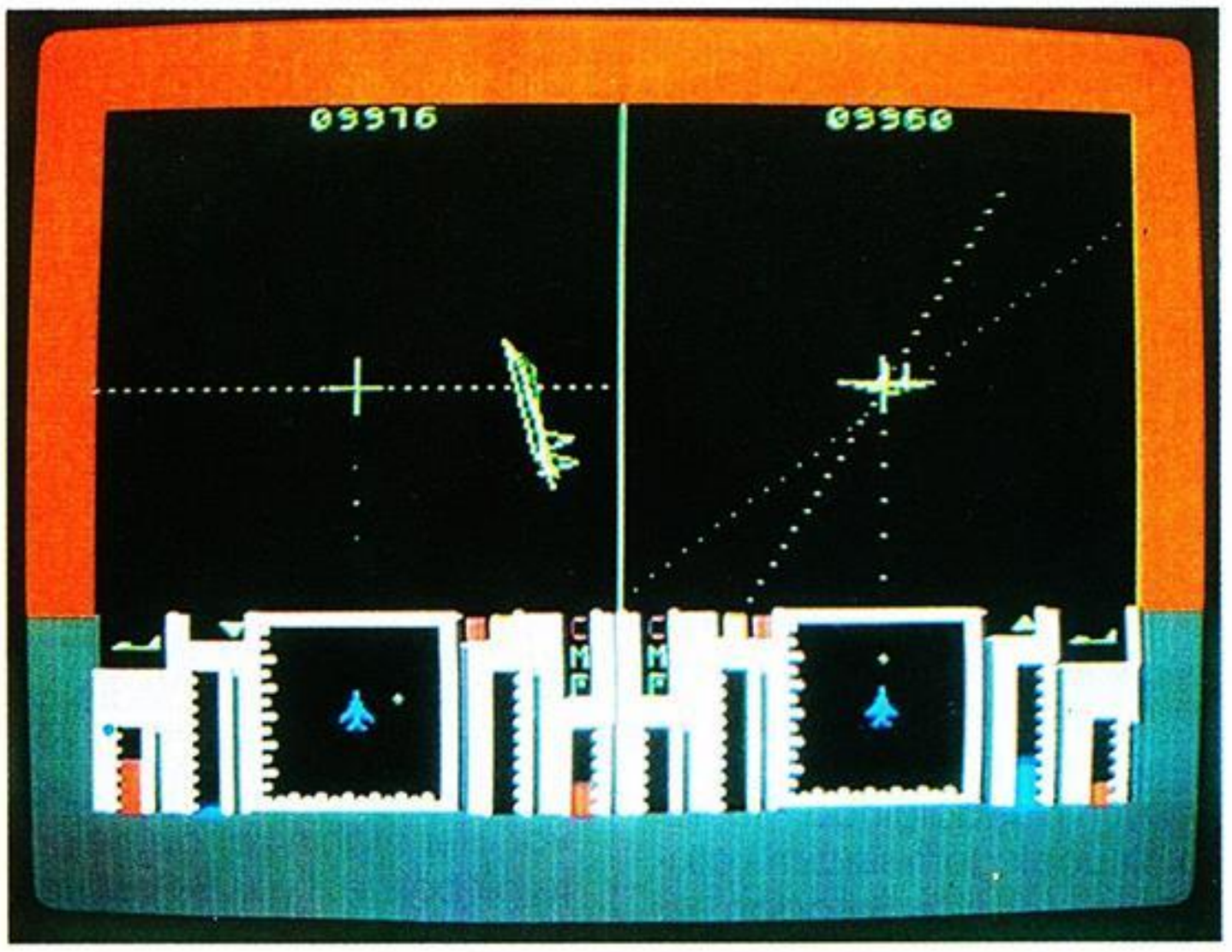


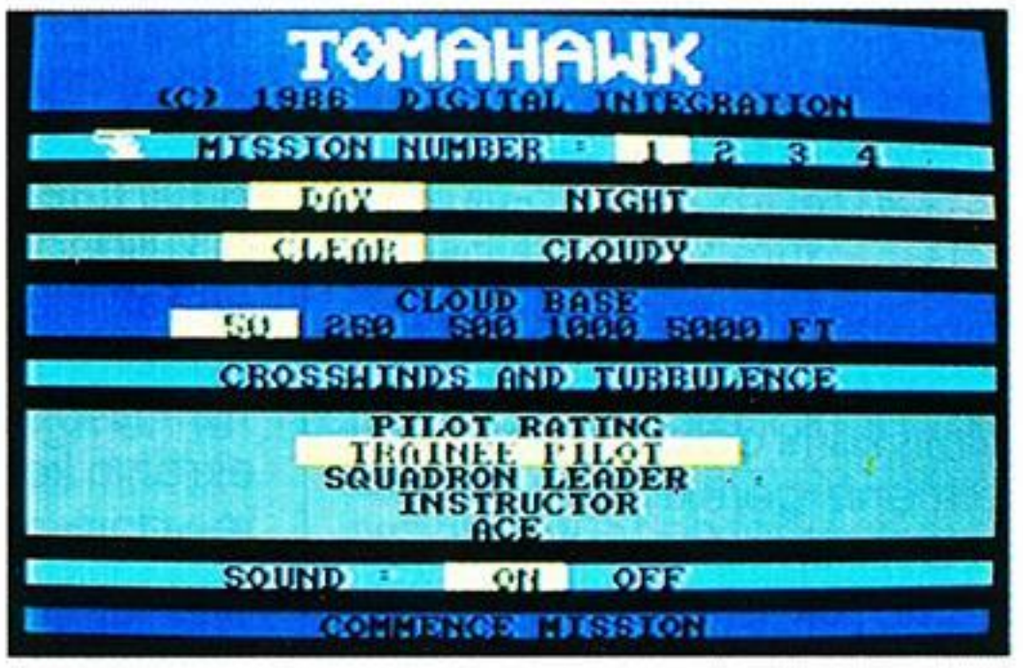
nizin hız, yükseklik, dikey hız, seçtiğiniz hedefe uzaklığın bildirildiği panel bulunuyor. En sağda helikopterin yönü, hedefe göre yönü, seçtiğiniz hedefin cinsi belirtiliyor. Bunların altında çeşitli aletlerin durumlarını izlemeniz mümkün. Bunların solunda helikopterin yatay pozisyona göre konumu ve burnunun durumu veriliyor.

Sıra geldi elinizin altındaki silahlara. Oyun sırasında aldığınız puanlar yok ettiğiniz hedeflere bağlı ama kullandığınız silah zorlaştıkça aldığınız puan artıyor. Silahların en zoru 30 mm'lık makineli tüfeğiniz. Makineli tüfeğin menzili sadece 2000 feet. Daha da kötüsü makineli tüfeğiniz tanklar karşısında çaresiz kalıyor. Makineli tüfeği kullanabilmeniz için hedefinizin hedef panelinde görüntüye girmesi gerekir. Hedefi vurabilme açısından, misillerin de hedef panelinde görüntüye girmesini sağlamalısınız ama misillerinizin etki alanı 4000 feet ile sınırlı. Bu misillerden helikopterinizde 38 tane bulunmakta.

Elinizde bulunan sekiz lazer güdümlü füze en etkili ve kullanılması en basit alacağınız puan da fazla değil. Lazer güdümlü füzelerinizin etki alanı üç mil. Bütün yapmanız gereken lazer güdümlü füzenizi hedefe kilitlemeniz, bundan sonra misil ateşlendikten sonra hedefini otomatik olarak kendisi takip edecektir.

Elinizin altındaki silahlar ve aletlerle tanıştığınıza göre artık rahatlıkla kalkabilirsiniz. Helikopterinizi istediğiniz gibi geri doğru uçurabilmeniz de mümkün. Helikopterinizi istediğiniz yere Joystick yardımı ile çevirebilirsiniz ama bu arada yüksekliğinize ve dikey hızınıza dikkat edin, kolaylıkla yere çakılabilirsiniz. Dağlarla karşılaştığınızda belirli bir yüksekliğin altına düşerseniz dağa çarpmanız kaçınılmaz olacaktır.





Helikopterin burnunu aşağı doğru çevirerek hızlanmasını sağlayabilirsiniz ama, 210 mph'u geçmemeye dikkat edin. Geçerseniz pervane kopacağı için helikopteriniz düşmekten kurtulamaz.

Bu hiç de hoş bir şey olmaz.

Silahlarınızı tazelemek için bir üste dönmek isterseniz size en yakın üstü seçin. Bu durumda üste uzaklığınız ve üstün yönü gibi size gerekli bilgileri elde edebilirsiniz. Üste yaklaştıkça yavaş yavaş hızınızı ve yüksekliğinizi azaltın. Üste vardığınızda ağır ağır inip yere dokunur dokunmaz bütün gücü kesin. Bundan başka helikopterin hızı 60 mph'a düştükten sonra yere inebilirsiniz. Eğer üstün uzağına inmişseniz bu şekilde yerden üste ulaşabilirsiniz.

Oyunda kullanmanız gereken kontrol tuşları

Tuş Fonksiyonu!

- C Hedef turu seçimi (tank, yer üstü silahı, helikopter)
- M Harita
- H Oyunu dondurma
- J Oyunu yeniden başlatma
- Q Gaz verme
- A Gaz kesme
- W Güç verme
- S Güç kesme
- O Seçilen hedef türlerinin görülmesi
- P Silah seçme (önce C tuşuna basıp silah türlerini belirten imleci görün)

Burak KİPER

TOP GUN

TOP GUN, konusunu aynı adlı filmden almış ve ortaya uçuş simulasyonu ile havada düello yaptığınız güzel bir oyun çıkmış. Filmde F16 savaş pilotları konu edilmekteydi. Oyunda da bir F16'nın içinde bilgisayara karşı veya bir arkadaşınıza karşı zor bir savaş vereceksiniz.

Oyun başladığında F16'nızla bir uçak gemisinden kalkışınız gösteriliyor. Buradan ekran uçağın içinden görünüşe geçiyor. Bu kısımda ekran ikiye ayrılmış durumda: Sol tarafta kendi kullandığınız uçak, sağ tarafta ise düşman uçağı bulunuyor. Oyunda başarılı olabilmek için ekranın size ayrılmış kısmından başka, düşman uçağına ayrılmış kısmı da dikkatlice takip etmeniz gerekiyor.

Oyunun uçuş simulasyon kısmı çok güzel yapılmış. Ezberlemeniz gereken yüzlerce kontrol paneli ve tuş yok. Bundan başka uçağı yerden kaldırmak veya yere indirmek gibi bir zahmete girmenize de gerek yok.

Takip etmeniz gereken kontrol panelleri ekranın size ayrılmış bölümünün alt kısmında yeralıyor.

Elinizdeki en önemli aletiniz düşman uçağına kilitlenen radarınız. Diğerleri yükseklik, hasar durumu, makineli tüfeğin sıcaklığı (devamlı ateş ederseniz makineli tüfeğiniz ısınacağı için tek tek ateş etmeye başlar), güdümlü misilin kilitlenme süresini gösteren belirteç ve misil uyarı sinyali.

Ekranı enlemesine geçen nokta nokta çizgi ufuk hattını temsil ediyor.

Kullanabileceğiniz üç tip silahınız var. O anda bu silahlardan hangisini kullandığınızı kontrol panelinize bakarak öğrenebilirsiniz. Kullandığınız silahın hangisi olduğunu anlamanın bir kolay yolu daha var. Her silahın kendisine göre bir belirteci var. Bu belirteç silahı seçtiğinizde ekrana çıkıp nişan almanızı kolaylaştırıyor.

İlk silahınız artı şekline benzer bir belirteç kullanan makineli tüfeğiniz. Makineli tüfeği kullanarak düşmanınızı yok etmek isterseniz 25 tane tam isabet kaydetmeniz gerekir. Tam isabet kaydederseniz ekranın kenarları kırmızı renkte yanıp sönecektir.

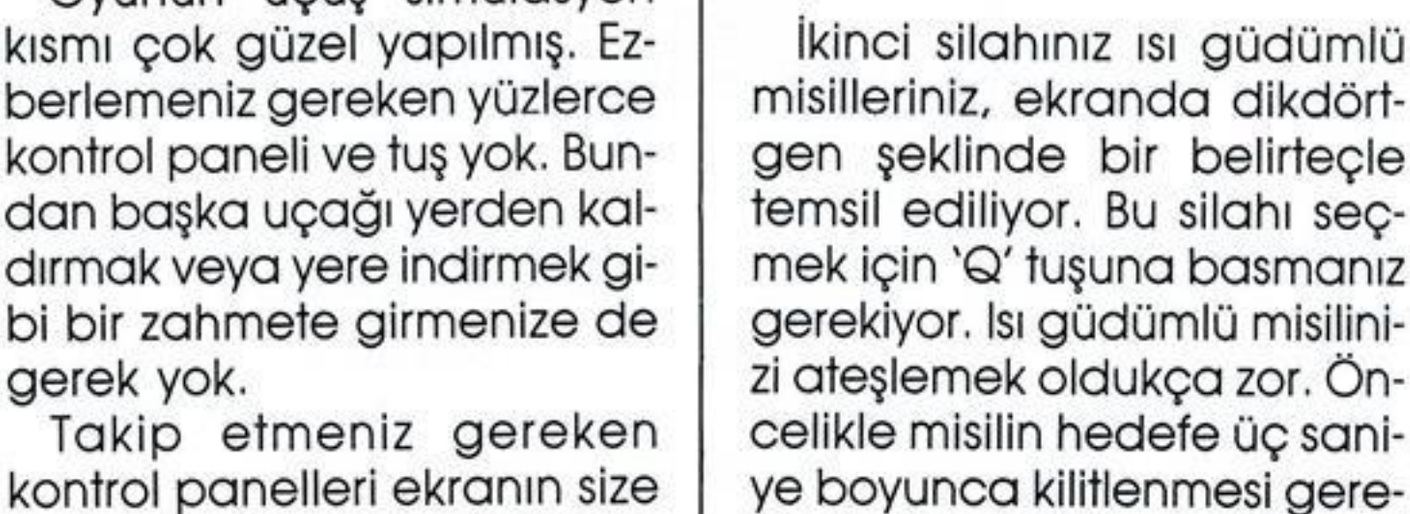
İkinci silahınız ısı güdümlü misilleriniz, ekranda dikdörtgen şeklinde bir belirteçle temsil ediliyor. Bu silahı seçmek için 'Q' tuşuna basmanız gerekiyor. Isı güdümlü misilinizi ateşlemek oldukça zor. Öncelikle misilin hedefe üç saniye boyunca kilitlenmesi gerekiyor. Ama bir kere kilitlendi mi artık düşmanınızın bundan kurtulması hemen hemen imkânsız olur.

Düşmanınızın size ısı güdümlü misil atması halinde tek kurtuluşunuz FLARE atarak misilleri yanlış yöne göndermenizdir. Son olarak belirteci olmayan bir misiliniz daha var. Bu misil hedefe atıldıktan yirmi saniye sonra patlar.

Oyunda her düzeyde üç hakkınız var. Yok etmeniz gereken üç düşman uçağını yok ettiğinizde bir sonraki düzeye geçiyorsunuz. Her düzey bir evvelkine nazaran daha zorlaşıyor. Mesela ilk düzeylerde düşman uçak ısı güdümlü misillerden kurtulamazken ilerleyen düzeylerde FLARE kullanarak kurtulabiliyor.

Eğer bir arkadaşınızla beraber oynuyorsanız karşılaşmadan sağ çıkan oyunu kazanıyor.

GRAFİK	:8
SES	:6
ZORLUK	:9
OYNANABİLİRLİK	:9
GENEL	:8



KRAKOUT

Şeytani bir canavar tarafından binlerce tuğla ile kaplı bir labirente hapsedilmiş bulunuyorsunuz. Tuğlalarla kaplı bir hapisaneden sopanızı ve topunuzu kullanarak kurtulmanız gerekiyor.

Bu oyunun en belirgin özelliklerinden birisi size sunduğu çok sayıdaki seçenek. Size sunulan seçenekleri içeren bir seçim ekranı bulunmakta. Şimdi size sunulan seçenekleri inceleyelim.

SES EFEKTLERİ: Ses efektlerini açıp kapamanızı sağlar.

MÜZİK: Müzik istiyorsanız seçiminizi belirtin.

AKİS: Oyunda görüntünün hangi yönde ilerlemesini ve-

ya hiçbir yöne ilerlememesini seçmeniz için.

TOPUN HIZI: Topun hareket hızını ayarlamanız için.

MODES: Sopanızın ve topunuzun rengini seçebilirsiniz.

RESET: Oyundan çıkarak C64'ün açılış ekranına dönmenizi sağlar. Eğer RESET düğmeniz yoksa kullanışlı olabilir.

SOPA TİPİ: Sopanızın iki ayrı hıza sahip olmasını sağlarsınız. Tamamen şansa bağlı olarak bir hızdan diğerine geçiş olur. Bu sayede işinizi biraz daha zorlaştırmış olursunuz.

Gelelim oyunda nasıl ekstra puan kazanacağınıza. Ekstra puan kazanabilmeniz için üstünde harf bulunan tuğlayı vurmanız gerekiyor. Her harf başka bir anlama gelmekte, her harfin size sağlayacağı yeteneğe göre ekstra fazla puan kazanmanız mümkün olabilir. Harflere göre kazanacağınız yetenekleri görelim.

G: GLUE kelimesinin ilk harfi. Anlamı yapışkanlık. Bu harfi taşıyan tuğlayı vurduğunuzda top sopanıza yapışacak.

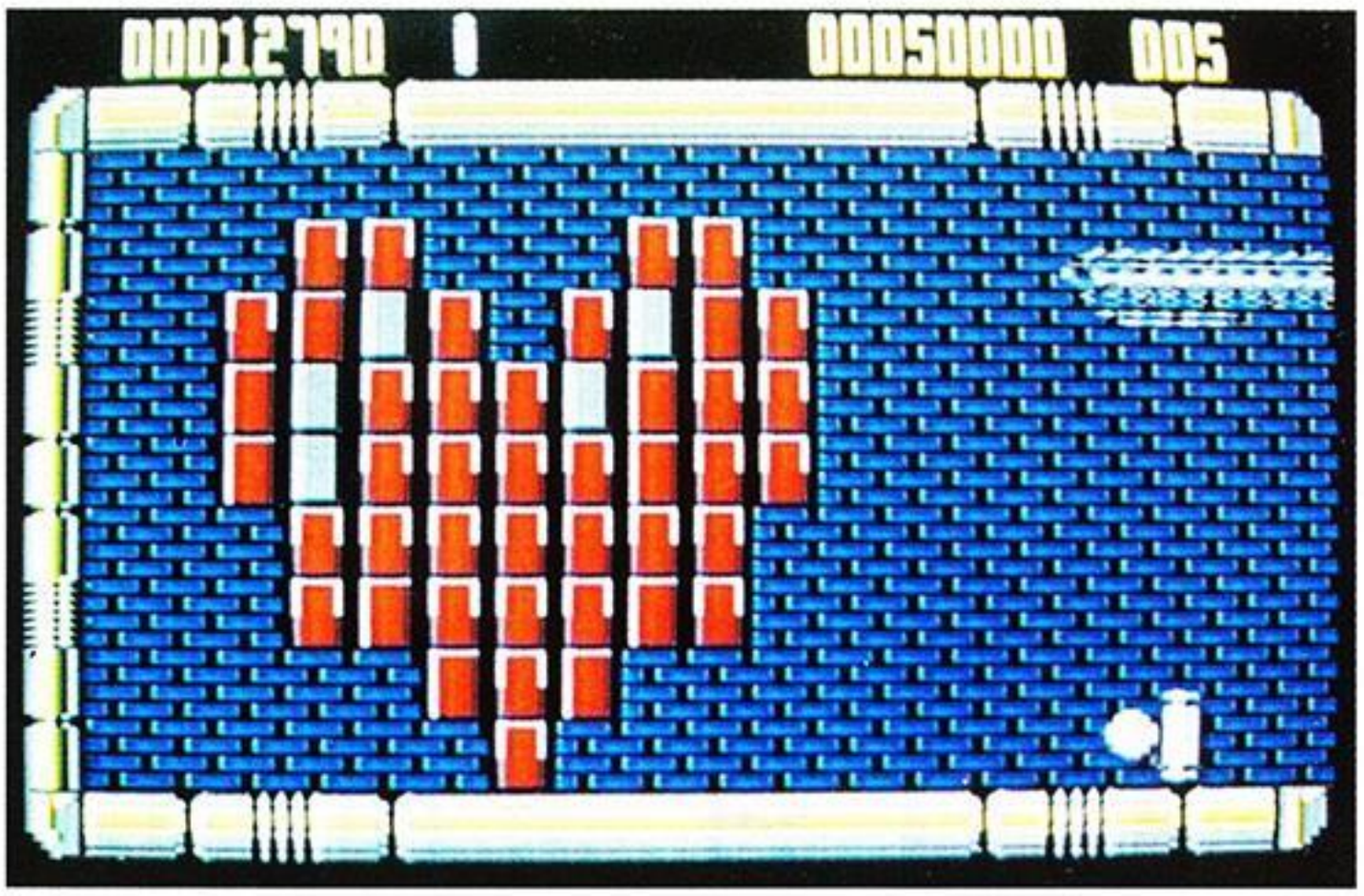
B: BOMB kelimesinin ilk harfi. Anlamı bomba. Bu harfi taşıyan tuğlayı vurduğunuzda patlama sonucunda etrafındaki tüm tuğlalar yok olacak.

S: SHIELD, kalkan anlamındaki kelimenin ilk harfi. Ekranın açık kısmında da bir duvar oluşur. Böylece top dışarı çıkamaz, siz de dinlenmiş olursunuz.

M: MISSILE, misil anlamındaki kelimenin ilk harfi. Bir sıra tuğlaya doğru misil atmanızı sağlar. Böylece o sıradaki tüm tuğlaları yok etme şansını elde ediyorsunuz.

SD: SLOW DOWN, yavaşlamak. Topun yavaşlamasını sağlar.

X2: Elde edeceğiniz skorların iki katını alırsınız.



D: Sopanızın önüne bir sopa daha konur. Böylece topu yakalamanız kolaylaşır.

X: Fazladan bir sopa sahibi olursunuz.

E: Sopanız iki katına çıkar. Tabii topu yakalamanız da o derece kolaylaşır.

Bir kerede sadece bir tek ekstra özellik kazanma şansınız bulunmasına rağmen tesadüfi puanlar alma ihtimali de mevcut bulunuyor. Nasıl tesadüfi puan alacaksınız? Bazen ekranda hızla uçuşan cisimler göreceksiniz, bunların size çarpmasını sağlarsanız bazen 10000 puan birden alabilirsiniz. Bazen arkada bir duvar oluşur, bazen birçok tuğla üstünde harf belirir.

Oyunun genelinde bir tek top bulunuyor ama iki topu birden idare etmek istiyorsanız ekrandaki buluta benzer şekle doğru topunuzu gönderin.

Bundan başka ekranda dolaşan yabancı cisimlerle de karşılaşacaksınız.

Şaşırmayın ama bunlar sizi öldürmeyecek. Peki sopalarınızı nasıl kaybedebilirsiniz? Eğer bir topa sopanızla vurmayı başaramayarak topu arkaya kaçırırsanız bir sopanızı kaybedersiniz.

Ekranda dolaşan cisimlerin size bir zarar vermeyeceğini söylemiştik ama bunlardan bir tanesi arıya benzer bir cisim fırlatıyor size. Eğer kaçmayı başaramazsanız sopanız olduğu yerde kala kalıyor. Tabii siz de topu kaçırıp sopanızı kaybediyorsunuz.

Diğer cisimler de, her ne kadar zararsızsalar da topunuz bunlara çarptığında yön değiştiriyor. Bazen yön değiştirmesi yüzünden topu arkaya kaçırıp sopanızı kaybedebiliyorsunuz.

Son olarak onyedinci düzeyden sonra bazen yuvarlak bir cisim çıkıp topunuzu yakalayıp yiyor.

Sopanızı Joystick ile kontrol edebilirsiniz, bundan başka boşluk çubuğuna basarsanız bulunduğunuz düzey bittiğinde başlangıç menüsüne dönersiniz. Tekrar boşluk çubuğuna basarsanız oyuna kaldığınız yerden devam etmeniz mümkün olur. PORT 1'e takılı Joystick'iniz varsa gelişigüzel hareket ettirerek oyunu dondurmanız mümkün olabilir.

GRAFİK	:8
SES	:8
ZORLUK	:8
OYNANABİLİRLİK	:10
GENEL	:9

BİLTEK
BİLGİSAYAR VE TEKNOLOJİ TİCARET

T.E.P.K.İ.

DAHA İYİ HİZMET İÇİN GÜÇBİRLİĞİ

YİNE BİLGİSAYAR DÜNYASININ TÜM YENİLİKLERİ

HARDWARE ÜRÜNLERİMİZ

DOLPHINDOS 2.0:

25 kız hızlı yükleme, 749 blocks free, monitör üzerinde, her türlü centronics printerle çalışma imkânı. 55.000 TL.

SPEEDDOS PLUS:

17 kez hızlı yükleme, özel programı ile 220 block 4 saniyede yüklenir. Gerçek zaman monitörü üzerinde. 20.000 TL

EPROM PROGRAMMER:

2732-2764-27128-27256 tip EPROM'ları programlar. Özel programı ile bütün programları kart üzerine hazırlar. 2764 8 saniyede programlanır. 50.000 TL

I.C. TESTER:

Her türlü TTL ve CMOS entegreyi test eder. Silinmiş ise numarasını gösterir. piyasada bulunan bütün kırıcı ve modül kartları üzerindeki entegreleri okur. Disk ve teyple çalışır. 55.000 TL

802 GRAPHIC ROM II:

802 printerleri 801, 803 ve EPSON uyumlu hale getirir. Printshop, print master ve 802 ile çalışmayan bütün programlarla çalışır. 15.000 TL

MODÜLKART:

Her türlü BASIC ve machine language program modülkart halinde hazırlanır.

DESTEK 64 TEYP:

Kartuş üzerinde teyp için programlar. (EXTRA MONİTÖR, TEYP KAFA AYARI, TURBO, PROGRAMLI FONKSİYON TUŞLARI, OLD, RENUMBER, TEK TUŞLA BASIC). 14.000 TL

DESTEK 64 DİSK:

Kartuş üzerinde disk için programlar. (FAST LOAD, DİSK MONİTÖR, DİSKTEN TEYBE, TEYPTEN DİSKE KOPYA PROGRAMLARI). 14.000 TL

EXPERT CARTRIDGE:

Programları tek parça halinde kırar, istediğiniz bir turbo ile kasete çekme imkânı sağlar. 40.000 TL

ICE MACHINE SUPER PLUS:

Programları diskten veya kasetten kopyalamanın en ucuz ve en kolay yolu. 25.000 TL

FINAL CARTRIDGE:

16 yeni kolay kullanım komutu. Diskten hızlı yükleme, program kırma, ekrandaki grafiklerin yazıcıya dökümü, mükemmel assembler-monitör programı. 25.000 TL

DOUBLE CARTRIDGE:

ICE MACHINE ve FINAL CARTRIDGE birarada. Daha ucuz, daha kullanışlı. 35.000 TL

PRINTER INTERFACE:

Her türlü printer Commodore 64'e adapte edilir.

- BİLGİSAYAR VE YAN BİRİMLERİ SATIŞI
- SPECTRUM, ATARİ PROGRAMLARI
- ÜRÜNLERİMİZLE BİRLİKTE GENİŞ AÇIKLAMA KİTAPÇIKLARI
- İMALATLARIMIZDA 8 AY GARANTİ
- 10000 TL VE ÜSTÜNDEKİ SİPARİŞLERDE ÖZEL EĞLENCE KASETİMİZ ÜCRETSİZ VERİLİR
- ÖDEMELİ İSTEKLER GÖNDERİLİR

BİLTEK
BİLGİSAYAR VE TEKNOLOJİ TİCARET

EBUZZİYA CADDESİ, MARMARA
ÇARŞISI KAT 1 No:105/106.
BAKIRKÖY/İSTANBUL

SOFTWARE ÜRÜNLERİMİZ

ANSYCLOPEDIA 64:

Commodore 64'de bilmek istediğiniz her şey. BASIC, machine language, grafik, müzik, sprite özellikleri. Tamamı Türkçe açıklamalı. 4500 TL

ART BASIC:

Commodore 64 ve Commodore 128'de en iyi grafik dili. Gelişmiş high-resulotion komutları. Sanatsal ve geometrik çizim olanakları. Türkçe el kitabı ile birlikte. 4500 TL

SHOWY 64:

C-64'ün tüm özelliklerini daha önce görmediğiniz şekilde sergileyen, grafik, müzik, speech ve spritelerden oluşan 60 süper demo programı. 2500 TL

LETTER MAKER:

Müzikli, hareketli reklam ve demolarınızı kendiniz hazırlayın. 2000 TL

COPYPACK II:

GM TURBO (Müzik ve grafikle yükler).
COUNTDOWN TURBO II (Diskte fastloadla kasette turbo v.3 ile çalışır.)
DISMON MONİTÖR, DİSK TO TAPE COPY, TAPE TO DISK COPY, TAPE TO TAPE COPY.
CR II (Kasetteki kilitli programlarınızı diskete kopya eder.)
HR TURBO II (Programlarınızı high-resulotion grafikle yükler.)
PROTECTOR (diskete korumalı kayıt). 5000 TL

SUPER UTILITIY II:

Kaset ve disk için 50 süper utility programı. (SIMON'S BASIC, EX BASIC, SPRITE ve KARAKTER EDİTÖRLERİ, TURBO SİSTEMLERİ, ASSEMBLER VE DISASSEMBLER PROGRAMLARI). 5000 TL

EXPERT 2.10:

Expert kart sahipleri için özel program, oyun grafiklerini yükleme, değiştirme ve programlarınızın içinde kullanma olanağı. 7500 TL

KUDI 1.2:

Komple matematik sistemi. Fonksiyonların açılımları, logaritma, türev, integral hesapları. Üst düzeylerde denklem çözümü. Almanya'da 1986 computer matemathic ödüllü program. 5000 TL

İŞ PROGRAMLARI PAKETİ:

Adres defteri, banka defteri, ajanda, cari hesap, bordro, kelime işlemci, hesap tabloları. 10.000 TL

MUSIC BOX:

C-64'ünüzü profesyonel müzik sistemine dönüştürün. (KAWASAKİ, ELECTRA SYNTHESİZER, BEATLES ALBÜM ve diğer ünlü müzik programları.) 3500 TL

ROCK ALBÜM:

Commodore'unuzda ALAN PARSONS PROJECT, PINK FLOYD, DEEP PURPLE, LED ZEPPELIN, SCORPIONS, DIRE STRAITS, IRON MAIDEN'in seçme eserleri.2500 TL

POP ALBÜM:

MODERN TALKING, C.C.CATCH, BAD BOYS BLUE, MADONNA, KIM WILDE, DURAN DURAN, SANDRA, AHA ve NENA'dan seçme eserler. 2500 TL

DOCTOR 64:

Commodore 64 arızalarını bulan program. Bilgisayarınızı tamire götürmeden arızasını bu programla öğrenin. 2000 TL

T.E.P.K.İ.

HALKÇI SOKAK, SAKOĞLU ÖZBEY
İŞHANI ALTÇARŞI No: 4
BAKIRKÖY/İSTANBUL



Kelime Denizi

Çeviren: Güven Savaş Kızıltan

Bir metin sistemine gereksinim duyduğunuzu biliyor muydunuz? Eğer bilmiyorsanız, neden böyle bir sisteme gereksinim duyduğunuzu aşağıdaki yazıdan öğrenebilirsiniz.

Doğruyu söylemek gerekirse, bir ev bilgisayarıyla yalnızca oyun oynanabileceğini ileri süren bir kişi C-64 ile gerçek bir çalışma ortaya çıkarmayı büyük bir olasılıkla henüz hiç denememiştir. Dolayısıyla, neler kaçırmış olduğunun bilincinde de değildir.

Hiç kuşkusuz, ev içinde de elektronik metin işleme çağının koşullarına uygun bir ortam yaratılabilir. Üstelik hem görece ucuz hem de yararlı bir iştir bu. Bu arada, özellikle C-64 için olağanüstü programlar sunan geniş bir seçenekler yelpazesi mevcuttur. Bu programlar sayesinde C-64, normal olarak bir kişisel bilgisayar tarafından gerçekleştirileceği pek umulmayan işler çıkarabilmektedir. Bunu bir örnekle somutlaştıralım: Bir iş dilekçesi yazdığınızı düşünün. Ama o ne?.. Bir "R" harfi yerine "E" yazmamış mısınız! Daktilo hatalarının iyi bir izlenim bırakmadığını bildiğiniz için yazmakta olduğunuz kağıdı çıkarıp çöp sepetine atıyorsunuz. Tabii bu arada sinirleriniz de bir parça geriliyor ister istemez ve böylece ikinci bir daktilo hatası programlanmış oluyor.

Eğer iş dilekçenizi bilgisayarla yazmış olsaydınız bu tür bir sorunun tümden ortadan kalkmış olduğuna emin olabilirdiniz. Bunu sağlamak için, bilgisayarın yanı sıra bir metin-işleme programına ve bir yazıcıya gereksinim duyacaksınız doğal olarak. İş dilekçenizin çok güzel bir yazı karakterine sahip olmasını düşündüğünüzü varsaydığımızdan, biz size bir teker-yazıcıyı tavsiye ederiz. Bunun çok pahalıya geleceğini sanıyorsanız, teker-yazıcı fiyatlarının 595 Mark'tan başladığını ve bilgisayar bağlantılı bir daktilo makinesinin de pek öyle dünyanın parasına malolmadığını söyleyelim.

Bir metin programının alışılagelen bir yazı makinesine oranla (bellekli yazı makineleri dışında) sahip olduğu büyük avantajı kuşkusuz biliyorsunuz: Bu sayede daktilo hataları çok kolay düzeltilebilmektedir. Yanlış harfin üstüne doğrusunun öylece yazılmasıyla düzeltim işi tamamlanmış olmaktadır. Metnin arasına bir kelimenin ya da tüm bir cümlenin eklenmesi de aynı derecede kolaydır. Bunun için nereden ekleme yapılmak isteniyorsa yazmaya oradan başlamak ve diğer bütün kelimeleri öylece sıralamak yeterli olmaktadır. Metnin geri kalan bölümü ise eklenen kelimelerin sayısına göre aşağıya doğru kaymakta ve bu işlem metnin genel düzeninde herhangi bir olumsuz değişikliğe yolaçmamaktadır.

İyilikler de (Kötülükler Gibi) Ender Olarak Tek Başlarına Gelirler

Silme işleminde de benzer bir durum söz konusudur. Tek tek harfleri ya da kelimeleri silme tuşunu kullanarak düzeltmek mümkündür. Daha büyük metin bölümlerini ise, silme işleminden önce işaretlemek gerekmektedir. Böylece, metnin hangi noktadan başlayarak ve ne ölçüde silinmesi

Bir cümle yanlış bir yerde mi duruyor? Olsun varsın. Kaygılanmaya hiç gerek yok. Hemen düzeltilir.

gerektiği tarafınızdan belirlenmiş olur.

Bir cümle yanlış bir yerde mi duruyor? Olsun varsın. Kaygılanmaya hiç gerek yok. Cümleyi ve cümlenin asıl durmak zorunda olduğu yeri işaretledikten sonra bir-iki tuşa da bastınız mı, bu cümlenin hemen ait olduğu yere geçmiş olduğunu göreceksiniz. Buna kaydırma işlemi denilmektedir. Bir metin parçasını kaydırmak yerine başka bir yere aktarmanız da mümkündür. Bu size, örneğin aynı türden pek çok satır içeren çizelgelerin hazırlanmasında epey zaman kazandırır.

Sıklıkla kullanılan bir başka yöntem de "Arama ve İkame Etme" işlemine dayanır. Buna bir kez alışan kolay kolay vazgeçemez bundan. Bu yöntemle, uzun adlar ve karmaşık terimler birer simge ile kodlanırlar ve daha sonra istenilen kelimeler bu simgelerin yerlerini alırlar. Örneğin, "Yaz Festivali" sık sık kullandığınız bir kelime ise, bu kelime için her defa "*" simgesini yazarsınız. Metni yazıp bitirdikten sonra test programına "*" simgesini aratabilir ve bu simgenin "Yaz Festivali" kelimesiyle ikame edilmesini sağlayabilirsiniz. O halde, kelimenin sadece bir kez doğru yazılması yeterli olmakta ve bunun dışında kalan her şey bilgisayar tarafından yerine getirilmektedir.

Ama iyi bir metin işlemenin sunduğu imkânlar hiç de bunlarla sınırlı değildir. Mektuplarınızın her birinin üzerine "Gönderen" olarak adresinizi yazmak sizin de canınızı sıktı mı hiç? Ya da adres bölümü dikdörtgen bir boşluk bırakılacak tarzda hazırlanmış zarfa, alıcının adresinin tam nasıl oturtulacağını her defasında yeniden çözmeye mi çalışıyorsunuz hâlâ?. İşte bilgisayarınıza gönül rahatlığı içinde teslim edebileceğiniz görevlere ilişkin iki çarpıcı örnek. Bunun gerçekleşmesi pratikte şöyle olmaktadır: Bir mektup örneği, istenirse, grafik olarak çok da güzel bir antet ile, tek bir defa hazırlanır. Siz bir defa hazırlanan bu mektup örneğini metninize yükleyebilirsiniz artık. O zaman bilgisayar günün tarihini ve her alıcının adresini sizin belirlemiş olduğunuz kısımlara yerleştirir. Böyle kaç mektup (tabii her biri ayrı bir adrese) yollayabileceğinize gelince, pek önemli değildir bu. Zaman zaman parti davetleri mi veriyorsunuz, bunlara küçük birer kişisel not da ekleyebilirsiniz. Böylece, konuklarınızdan her biri, basit mektup nüshaları yerine özel olarak kendisi için yazılmış bir mektup almış olur. Bir metin sistemi bu imkânı yaratmaktadır. Kişisel bir hitap içeren sirkülerlerin alıcı üzerindeki etkisi artık pek çok firma tarafından anlaşılmıştır. Kuşkusuz siz de, size adınızla hitap eden ve posta ile kitlelere ulaştırılan yazılarla karşılaşmışsınızdır. Bunu sağlamak için bir adres veri yığını hazırlanmakta ve metin programına bunlardan hangilerini mektuba aktarmak zorunda olduğu bildirilmektedir. Böylelikle de, adres ve hitap dışında aynı mektup otomatik olarak basılmış olmaktadır. Bundan sonra yapılacak şey, yazıları zarfa yerleştirmek, pulları yapıştırmak ve mektupları posta kutusuna atmaktan ibarettir.

DOĞRU KULLANIM

Görüldüğü gibi, bir metin-işleme aracılığıyla bazı şeyler yapılabilmektedir. Bu arada şunu da eklemek gerekir ki, işaret etmiş olduğumuz bu imkânlar, yapılabilecek şeylerin sadece küçük bir bölümünü oluşturmaktadır. Ancak, bir metin-işlemenin sınırlarının da içtenlikle belirtilmesinde yarar vardır. Örneğin, kazanç vergisiyle ilgili bir formüleri bilgisayarla doldurma girişiminde bulunmak anlamsız bir iş olur. Bu formülerlerin formatlarının belirlenmesi öylesine ayrıntılı bir çalışmayı öngörmektedir ki, her alt bölümün yerli yerine oturtulabilmesi yüzlerce denemeden sonra ancak mümkün olabilmektedir. O nedenle bu konuda en iyi sonuç veren aygıt emektar daktilo makinemizdir hâlâ. Ama öte yandan, eğer pek çok sayıda formüler doldurma zorunluluğu varsa (örneğin, bir faturayla ya da bir banka havalesiyle ilgili olarak), bu formüleri bir defalık olmak üzere düzenleme zahmetine değecektir. Bir defa düzenlendi mi, artık bu konuda kafa yormanıza hiç gerek kalmaz, çünkü bilgisayar sizin tarafınızdan programlanmış şeyleri doğru olarak



TABLO 1 Bir Metin Programının Seçimine İlişkin Ölçütler

- ☐ Ekran Görüntüsü (satır başına 40/80 simge, yatay kaydırma)
- ☐ Yazımlama (simge ya da blok tarzında)
- ☐ Metin Bölümlerinin Aktarılması (Copy)
- ☐ Metin Bölümlerinin Kaydırılması
- ☐ Floppy-Yazma Parametresi
- ☐ Çizelge Oluşturucuları (Tabülatör)
- ☐ Yazıcının Özel İşlevlerine Uygunluk
- ☐ Farklı Yazıcılar için Özel Yazma-Modu
- ☐ Metnin Diskete Aktarımı ve Saklanması
- ☐ Hesaplama İşlevi
- ☐ Veri Yönetimi için Port
- ☐ Satır Sonunda Kelime Ayrılması
- ☐ Yanaştırma Komutları (Blok-cümle , sola/sağa yanaştırma)
- ☐ Formüler Fonksiyonu
- ☐ İşleme Hızı
- ☐ Metin Deposu Hacmi
- ☐ Silme İşlevi
- ☐ DOS-Komutları Disketleri
- ☐ Metin Kaybı Olmadan Disket Kataloğu (Directory)
- ☐ Ek Metin Deposu
- ☐ Sayfa Numaralaması
- ☐ Değişik Metinlerin Birbirlerine Bağlanması
- ☐ Yazı Makinası İşlevi
- ☐ Üst ve Dipnotlar
- ☐ Grafik Simgeleri
- ☐ Alman Klavye
- ☐ Almanca'daki ''Ünlü Değişimi'' (CBM-Yazıcılarında da)
- ☐ Metin Bölümlerinin Bulunması
- ☐ Sayfaların Bölümlenmesi
- ☐ Metin Sayfalarının Hızlı Çevrimi
- ☐ Farklı Interface'lere Uyum
- ☐ Metinde Yer Tutucu
- ☐ Renk Ayarı
- ☐ Yardımcı Menüler
- ☐ DFÜ-Modu

ve istediğiniz sıklıkta adım adım uygulayacaktır. Dolayısıyla, bir metin-işlemede önemli olan, gerçek kullanım amacının doğru saptanmasıdır. Sonuç olarak hangi aygıt konfigürasyonun tavsiyeye değer olduğunun belirlenebilmesi de buna bağlı olmaktadır. Genel olarak fiyat/verimlilik/teknik bağlamında üç farklı ilke arasında bir seçim yapmak sözkonusudur. Esas olarak yazıcılar teker yazıcılar ve matris yazıcılar olarak ikiye ayrılır. Matris yazıcılar ise, iğne yazıcılar, mürekkep fışkırtmalı yazıcılar ve termal yazıcıları kapsar. Bu ilkelerden her birinin kendine göre avantajları ve dezavantajları vardır. Kural olarak, özel metin-işleme için en uygunu ''Near Letter Quality'' denilen güzel yazı karakterine sahip olan iğneli matris yazıcılardır. Bu yazıcılar, yüksek düzeyde bir esnekliğe, hıza ve çok iyi yazı karakterine sahip olma avantajlarını kendilerinde toplamışlardır. Ayrıca da, bunları genel olarak ucuza maletmek mümkündür. Ama öte yandan, ilerde çalışmak istediğiniz program da tıpkı aygıt donanımı kadar önem taşır, çünkü sizin için hangi imkânların mevcut olduğunu bu program saptamaktadır. C-64 Yazılım pazarında sunulan programların çeşitliliği karşısında bu programlardan herhangi birinde karar kılmak kolay olmasa gerek. Bununla birlikte, öbürleri arasında dikkati çeken ve neredeyse bütün talepleri yerine getirebilen yarım düzine programın varlığından sözedilebilir. Kendi amaçlarınız doğrultusunda hangi programı kullanacağınız öncelikle bir beğeni meselesidir. Örneğin, avantajları ve dezavantajları vardır. Kural olarak, özel metin-işleme için en uygunu ''Near Letter Quality'' denilen güzel yazı karakterine sahip olan iğneli matris yazıcılardır. Bu yazıcılar, yüksek düzeyde bir esnekliğe, hıza ve çok iyi yazı karakterine sahip ''menü'' ile yönlendirilen metin programlarının yanısıra, fonksiyonların doğrudan çağrılabildiği programlar da vardır. Bir ''menü'' yönlendirimi kuşkusuz pek hoş bir şeydir, ama ne yazık ki çok zaman alır. En ideal çözüm, her iki seçenek arasında bir orta yolun bulunmasıdır. Buysa, iyi bir metin-işlemenin en önemli özelliğinin ne olduğuna bir göndermedir: Metin-işleme olabildiğince esnek olmalıdır ve burada hangi yolun seçileceğine ilişkin karar size bırakılmalıdır. Örneğin bazı programlar yükleme işleminden kısa bir süre sonra çok değişik soruların yanıtlanmasını şart ko-

TABLO 2

- ☐ Grafik Yeteneği
- ☐ Simge Cümleleri (CBM, Uluslararası Simgeler)
- ☐ Yazma Hızı: Saniye Başına Düşen Simge
- ☐ Kağıt Türleri: Termal, Rulo, Tek Kağıt
- ☐ Özel İşlevler
- ☐ Commodore İçin Interface
- ☐ Kullanım Kolaylığı: Satır ve Sayfa Ölçüleri için Tuşlar, Bazı Sürekli İşlevler için DIL Şalterine Ulaşma
- ☐ Servis Kolaylığı
- ☐ Tampon Deponun Büyüklüğü
- ☐ Ses Düzeyini Gösteren Aygıt
- ☐ Kullanım Talimatı
- ☐ Fiyat: Hangi ek donanımlar buna girmektedir? Örneğin bir Interface fiyata dahil midir?
- ☐ Escape-P-Standardı?

Yazı Türleri:

- ☐ Yazışma Basımı
- ☐ Kalın Harfler
- ☐ Çifte Basım
- ☐ Seçkin Yazı
- ☐ Orantılı Yazı
- ☐ Büyütülmüş Yazı
- ☐ Yazının Altının Çizilmesi
- ☐ ''Near Letter Quality'' Tarzında Yazı
- ☐ İtalik Yazı
- ☐ Sık Yazı
- ☐ Revers Basımı
- ☐ Özel İşlevler
- ☐ Değişik Nokta Yoğunluklarında Grafik Yeteneği
- ☐ Satırların Ayarlanması
- ☐ Kağıt Uzunluğunun Saptanması
- ☐ Sol ve Sağ Kenarın Belirlenmesi
- ☐ Geri Adım
- ☐ Sayfa Sonunun Saptanması
- ☐ Programlanabilen Yazıcı Reseti
- ☐ Kağıdın Geriye Aktarımı
- ☐ Sesin Azaltılması için Basım
- ☐ Hızının Küçültülmesi

Sonuç olarak yorgun argın, daktilo hatalarıyla, eskimiş tuşlarla başetmeye çalışmak hiç de çağdaş, günümüze uygun değil artık.

şarlar; yazıcının ne türden olduğuna ilişkin soru gibi. Ayrıca, tarihin verilmesi ve aygıt konfigürasyonun saptanması gerekmektedir. Gerçi bu tür şeyler ilk bakışta çok hoş görünürler, ama eğer siz parametreniz konusunda bir karara varmış iseniz, genel olarak zaman alırlar. Bu parametrelerin bir defa saptanması ve ardından da sağlam bir biçimde depolanması çok daha iyidir. Bundan sonra program her zaman otomatik olarak parametrenizi yükleyebilecek güçte olmalıdır.

GÜÇ SEÇİM

Bir metin sisteminin (ki buna her zaman bir yazıcı da dahildir) satın alınması konusunda tavsiyede bulunmak kolay değildir. Çünkü, burada çok farklı beklentiler ve talepler gündeme gelmektedir. Kaldı ki, belirli hiç bir programı mutlak olarak önermek de mümkün değildir. Bununla birlikte, programın satın alınmasından önce 1. ve 2. tablolarda belirtilen ölçütler ışığında gerek duyulan verimlilik özelliklerinin biraraya toplanmasında yarar olabilir. Elde böyle bir liste olunca uzman firmanız da kuşkusuz sizi bu konuda daha iyi aydınlatma imkânına sahip olacaktır. Gerçi şu ya da bu metin-işlemenin mevcutların en iyisi olduğunu söylemek bir bakıma kolaydır, ama bu sizin kendinizi belli bir metin programına uydurmak zorunda kalmanıza yolaçar. Seçimi sizin yapmanız ve kendiniz için en uygun düşenin hangisi olduğuna sizin karar vermeniz çok daha iyidir. Bu konuyla ilgili olarak kesinlikle söylenebilecek bir şey varsa, o da şudur: Yorgun ve tuşlara vurmaktan aşınmış parmaklarla yazı makinesinin önünde oturmak ve sinirleri daktilo hatalarına kurban etmek hiç de çağdaş bir görünüm sunmamaktadır.



Teknoloji yazıyı ekrana aktardı

Metin İşleme

Çeviren: Güven Savaş Kızıltan

Metin işlemenin bir sistem olarak adlandırılması pek de yersiz sayılmaz, çünkü metin işleme olabildiğince iyi aygıtlar ve programlar satın almaya bağlı değildir yalnızca, bunlardan öte bir anlam taşır. Bu bağlamda önemli olan şey, bütün bölümlerin büyük bir uyum içinde bulunmalarıdır. Burada size harmonik bir yapıya sahip 10 değişik sistemi tanıtmak istiyoruz.

Metin işleme çok farklı ilgilerin karşı karşıya geldiği bir alandır. Bu konuya el atmış olan herkes bütün taleplere uygun düşebilecek aygıtlar ve yazılım biçimi oluşturulmasının zor olduğunu bilir. Optimal sonuçlara ulaşmak için her zaman değişik yollar mevcuttur, ama bunlar arasında asıl tavsiye edilmeye değer olan tek yol daha sonra betimlenmiş olanıdır. Şüphesiz siz kendi bilgisayarlarınızla neler yapmak istediğinize dair belli tasarımlara sahipsinizdir. Ağırlığı oyun sektörüne ya da kullanım yazılımına vermiş olabilirsiniz. Tercihlerinize bağlı olarak, bilgisayar sisteminiz de farklı bir görünüme bürünmüş olacaktır. O halde, yapılması gereken ilk şey, bir sistemden hangi fonksiyonları yerine getirmesinin beklendiği konusunda açık bir görüşe sahip olmaktır.

Bu bağlamda ilkin bir sistemin ne olduğunu açıklamak gerekiyor. Bir bilgisayar sistemi denildiğinde, optimal olarak birbirleriyle uyum içine sokulmuş tek tek yapı taşlarından oluşan bir grup anlaşılmaktadır ki, burada sistemin her parçasının sıkı bir biçimde tanımlanmış bir görev alanı vardır. Bir bilgisayar sisteminde sözkonusu olan şey, sistemin bir bölümünün sonucunun diğer bir bölümün girdisini oluşturmasına dayanan salt bir kurallar döngüsü değildir. Bunun arasında "işletme süreci" denilen bir işlem yeralmaktadır. Bütün bunları daha açık ve somut olarak ifade etmek gerekirse şöyle diyebiliriz: Klavye giriş aygıtıdır, metin işleme programı işlem yapan bölümdür, Floppy, Yazıcı ve Monitör ise çıkış aygıtlarıdır. Monitör ve kimi zaman da yazıcı özel bir konuma sahiptirler, çünkü bunlar insana metin işleme sürecini, denetleme imkânını sağlayan kontrol aygıtları olarak hizmet görürler. Örneğin, eğer siz bir hata yapacak olursanız, ekranda yanlış bir harf belirecek ve bu da sizi hatanızı düzeltmek için harekete geçirecektir. Bu açıdan bakıldığında, burada küçük çapta bir kurallar döngüsünden bile sözedilebilir. Bu sistemin her bölümü arasında kesim bölgeleri bulunmaktadır. Siz bu kesim bölgelerinden bazılarıyla ilgilenmek zorunda değilsiniz, çünkü bunların taşıdıkları önem sistemin iç yapısıyla ilgilidir ve bunlar bizzat bilgisayar tarafından denetlenirler. Bilgisayarınızın depo yönetimi buna örnek gösterilebilir. Ama bütün dışsal kesim bölgelerinde durum farklıdır. Kuşkusuz bu kesim bölgelerinden en çok bilineni insan ile bilgisayar arasında yeralan



Resim 1: PC 10 Sistemi- Üzerinde durulacak bir sistem.



Resim 2: Protext, C128-Sistemi'ne profesyonel yetiler sağlıyor.

klavyedir. Eğer bu kesim bölgesi yanlış kurulmuşsa, sık sık yanlış yazacağınızdan dolayı, hata oranı büyük ölçüde artar. Bu arada şunu da belirtmek gerekir ki, lastik klavye denilen klavyeler metin işleme konusunda hepten en uygunsuz olanlarıdır.

Öte yandan, bilgisayar ile monitör ve Floppy ile yazıcı arasında da kesim bölgeleri vardır. Herhangi bir şeyin işlenememesinden çoğunlukla bu geçiş bölgeleri sorumludurlar. Bir bilgisayar satış firmasına gittiğinizde, eğer orada sergilenen aygıtların sadece pırıl pırıl dış görünüşlerini incelemekle kalmayıp bunların ötesini araştıracak olursanız, sizi şaşkınlığa uğratacak şeyler keşfedebilirsiniz. Hemen her aygıt başka bir kesim bölgesiyle donatılmıştır. Buysa, bilgisayara ilgi duyan, ama bu alanda deneyim sahibi olmayan kişinin epey canını sıkan karmakarışık bir görünüm sunar. Bunun yarattığı şaşkınlıktan bilgisayar satış firmasının uzman personeli de tam anlamıyla kurtulmuş değildir sıklıkla. İşte bu yüzden, müşteriye çalışmayan sistemlerin de tavsiye edildikleri görülmektedir. Gerçi müşterinin de satıcının işini kolaylaştırdığını söylemek zordur; çünkü tüm sistemi bir defada satın alanlar azınlıktadır, pek çokları bunu ekonomik durumları elverdiğince parça parça edinirler. Aslında sistem, değişik kesim bölgelerinin çeşitliliğinin ardına gizlenmiştir. Doğal olarak birçok yapımcı, bilgisayarları için kendi ürettikleri çevre aygıtlarını satmak ister. Commodore'un seri halindeki IEC-Taşıtı'nın yalnızca Commodore'da bulunmasının nedenlerinden biri de herhalde budur. Yapımcılar tarafından yaygın olarak kullanılan bir başka hile de şudur: Sinyal aktarımı yönteminde standardı korumak (gelişim giderlerinden tasarrufu sağlamaktadır bu), ama öte yandan da bilgisayara kendi geliştirmiş oldukları yuvaları takmak... Böylece, bunların hiçbir yerde serbestçe satışa çıkarılması güvence altına alınmış olmaktadır. Gene de, bazı kurnaz yapımcılar ve aracı firmalar bu yapımcının aygıtlarını başka bir yapımcının bilgisayarına bağlamanın yolunu bulmuşlardır. Bunlar, her iki aygıt arasında devreye sokulabilen Interface'ler oluşturmuşlardır. Bu Interface'ler Bit ve Byte birimleriyle ilgili olarak çevirmenlik görevini üstlenen araçlardan başka bir şey değildirler. Bu araçlar anında çeviri yoluyla farklı aygıtların birbirlerini anlamalarını sağlarlar. Ama ne yazık ki, her "dilmaç" değişik yapımcıların geliştirdikleri bütün dilleri anlayamaz. Nerdeyse her biri için başka bir dilmaça gereksinim duyulur. Bütün bu söylenenler sizde büsbütün mü şaşkınlık yarattı acaba? Umarız yaratmamıştır. Hem zaten bu karışıklığa -konuyla ilgili daha küçük bir alana el atarak da olsa- artık bir son verip sırasıyla on adet sistemi tanıtmak istiyoruz size. Bunlar sınanmış ve çalışan sistemlerdir. Bu arada her keseye göre bir sistem seçilmesine dikkat ettiğimizi de belirtelim. Burada en önce PC 10'u ele almamızın nedeni de budur.



Resim 3: Referans yazıcımız Fujitsu DX 2100, C128-Sistemi'ne en uygun olanıdır.



Resim 4: FX-85 ve C64: İyi bir öneri

Almanya'da sunulan seçenekler çok yakında ülkemizde de yazarların hizmetine sunulacaktır.

SÜPER SİSTEM

PC 10 (resim 1) bir MS-DOS bilgisayarıdır. Adını, onun sistemini programlamış olan Microsoft-Disk Operating System adlı yazılım firmasından almıştır. MS-DOS kişisel bilgisayarlar için CP/M'nin yanısıra en yaygın sistemdir. MS-DOS'un temelinde Intel 8086 ya da 8088 mikroişleyici bulunur. Inter 8086'nın gerçek bir 16-Bitlik mikroişleyici olmasına karşılık Intel 8086'dan farklı olarak yalnızca 8-Bit-Bilgi taşıtına sahiptir. MS-DOS için herkesin hoşuna gidecek hemen her türlü program mevcut-

tur. Microsoft Word (Fiyatı: 1708 Mark, resim II) ya da Mikropro Wordstar (Fiyatı: 1470 Mark) gibi üst düzeyde programlar bunlar arasındadır. Ama aynı zamanda bir Vizawrite PC de mevcuttur. Bu sistemin avantajı işleme sürecinin yüksek hızından, programların mükemmel fonksiyon donanımlarından ve 80 simgeli ekran görüntüsünden kaynaklanır. Bir PC 10 ile kural olarak yalnızca monokrom bir monitör üzerinde çalışılabilir. Ama istenirse, PC 10'u çok sayıda delikli kartlarla ve bu arada bir renkli kartla donatmak mümkündür. Bir yazıcının PC 10'a bağlanması son derece kolaydır; çünkü bunun için basit bir standart kablonun bilgisayar ile yazıcı arasına takılması yeterlidir. Biz PC 10'a fiyatı 1400 Mark'ın üzerinde olan Fujitsu DX 2100'ü bağladık. Bu yazıcı PC 10'un ideal bir tamamlayıcısıdır. PC 10'un klavyesi de daha ucuz sistemlerinkinden çok daha iyidir doğal olarak. Bu klavye hızlı ve zahmetsiz çalışmayla ilgili bütün talepleri karşılayabilecek özellikte olup yassı bir forma sahip ayrı bir muhafaza içine yerleştirilmiştir. PC 10 sisteminin bütün fonksiyonlarını bir bir saymak bu yazının sınırlarını aşar, ama 64'er Dergisi'nin büyük bir bölümünün PC 10'a hasredilmiş olması bile bu aygıtın verimlilik kapasitesine tek başına ışık tutabilir. Bu sistemin fiyatı, Microsoft Word da dahil olmak üzere yaklaşık 9320 Mark dolayındadır. Ama etraflıca araştırıldığı takdirde birkaç bin Mark daha ucuzunu bulmak da pek o kadar zor değildir.

Metin işleme sistemini kurmadan önce ihtiyaçlarınızı saptayıp alım gücünüzü belirleyin ve daha sonra seçiminizi yapın.

SESLİ AMA GÜZEL

Size tanıtmak istediğimiz ikinci sistem (resim 2) 1901 monitörlü, 1571 Floppy-Sürücülü ve Uchida DWX 305 papatya-teker yazıcılı C128'dir. 80 simge ile çalışan ve doğru bir yazılımla çok iyi işler çıkarabilecek yeteneğe sahip bir bilgisayardır C128. 1571 Disket-sürücüsü görece hızlı olup 5 1/4 disket üstünde 340 KByte depolayabilmektedir. C128 80 simgeli görüntü için çok uygun düşen 1901 monitörü ile birlikte bölümleri arasında harmoni sağlayan bir sistemdir. C128 ile uyum sağlaması için biraz dikkat gösterilmesi gereken tek aygıt yazıcıdır. Biz C128 için Uchida DWX 305 yazıcısını üç nedenden dolayı seçtik: Birincisi, bu aygıt çok sayıda tekeriyle mükemmel bir yazı kalitesine sahiptir; ikincisi 598 Mark fiyatıyla son derece ucuza gelmektedir ve son olarak üçüncüsü de bu aygıtın merkezcil bir kesim-bölgesine sahip olmasıdır ki, C128 bundan farklı tarzlarda yararlanabilmektedir. Bir papatya-teker yazıcısı hiçbir zaman bütün simgeleri ve bu arada da C128'in grafik simgelerini yazabilecek güçte olmadığından buna bir çözüm bulunması gerekir. Ama bu sorunun çözümü son derece basittir. C128 ile yazıcı arasına basit bir User-Port kablosu çekilir (kablonun fiyatı yaklaşık 50 Mark'tır). Ne yazık ki, bu yöntemin beraberinde getirdiği şöyle bir dezavantaj vardır: User-Port aynı zamanda yazılımla da kullanılmak zorundadır. Bunun yönlendirilmesini bu sisteme ait bir yazılım gereci olan Protext 128 (89 Mark) üstlenmiştir. Bu metin programı hemen her yazıcı için optimal bir uyumu mümkün kılar. Bu durumda Uchida'nın yönlendirilmesinde herhangi bir sorun ortaya çıkmaz. Bu sistemin fiyatı toplam olarak yaklaşık 3830 Mark'tır.

İĞNE VURUŞLARI

Üçüncü sistem (resim 3) bir önce tanıtılmış olandan yalnızca bağlanan yazıcı yüzünden bir farklılık sunar. Bu sistemin referans verdiğimiz yazıcısı Fujitsu DX 2100 evrensel bir



Resim 5: Uchida DWX 305 güzel yazının gerçek bir ustasıdır.



Resim 6: SG-10 ile C64 kendini çok iyi ifade eder.

kullanım sunduğundan bu açıdan hiçbir sorun çıkarmaz. Sivri ve keskin "Near Lether" tarzındaki yazısıyla saniyede 44 simgeyi kağıda vuran bu yazıcı kendi türünde bir mucize yaratır. 9-İğne-matris yazıcı olarak DX 2100 bir teker-yazıcısının çok güzel yazı karakteriyle, bir matris yazıcısının hızını ve esnekliğini bünyesinde toplamıştır. DX 2100'ün bilgisayara bağlanması için biz yine basit bir User-Port kablosunu tavsiye ederiz, çünkü bu kablo mümkün olan en büyük esnekliği koruyabilmekte ve Interface'in yol açabileceği hataların baştan önüne geçebilmektedir. Eğer DX 2100 bir renk donatımıyla zenginleştirilecek olursa, metinlerin çok renkli bir tarzda hazırlanması da mümkün olabilmektedir. Bazı yazılar da böylelikle daha hoş bir görünüm kazanmış olurlar. Bu arada renkli yazının kalitesinin siyahınkiyle aynı oluşu da özellikle dikkat çekicidir.

Bu sistem çok ucuz olmamakla birlikte, sistemin verim kapasitesi çok yüksektir. Fiyatı toplam olarak yaklaşık 5165 Mark'tır.

STANDART

Şimdi ele alacağımız sistem (resim 4) verim kapasitesi yüksek, esnek ve teknik bakımdan yüksek değer taşıyan bir sistem edinmek isteyen bütün C64 sahipleri için düşünülmüştür. Sistem C64'ün yanısıra (500 ilâ 600 Mark) Floppy 1541'den (mümkünse Floppy Speeder ile birlikte), 1702-Monitörden (yaklaşık 700 Mark) ve bizim referansımız olan Epson FX-85 yazıcısından oluşmaktadır. C64'de metin işleme programları açısından ilkece olağanüstü bir çeşitlilik sözkonusudur, ama biz bu seçeneklerden yalnızca üçünün tavsiye edilmeye değer olduklarını düşünüyoruz. Gerek Startexter (resim 12, 64 Mark) ve Textomat Plus (99 Mark) ve gerekse Vizawrite 64 (resim 13, 245 Mark) C64 programlarının rakipsiz üçlüsünü oluşturmaktadır. Her üçünde de yüksek düzeyde bir esneklik ve anlamlı komutlar bakımından büyük bir çeşitlilik göze çarpar. Hattâ Startexter ve Vizawrite 64'ün kendilerine özgü birer merkezcil kesim-bölgeleri de vardır (Protext 128 gibi). Bu da, pahalı donanım-çözümlerini bir yana bırakıp basit bir User-Port kablosuyla yetinmeyi mümkün kılan bir özelliktir. Tabii bu sistem yalnızca metin işleme için kullanılmayacağından, yazıcının metin işleme olmaksızın ve ek bir Interface konulmadan nasıl işletilebileceği sorulabilir. Bu problemin ideal çözümünü 64'er'in 11/1985 sayısında yayımladığımız "Kalp Ameliyatı" başlıklı makalede gözönüne sermiştik. O sayıda çıkan programda yeni bir işletme sistemi ortaya koymuştuk. Bu sistem User-Port'u kendi normal fonksiyonlarının yanısıra varlığını sürekli koruyan bir merkezcil kesim-bölgesine dönüştürmektedir. Bu durumda FX-85 gibi bir yazıcı, Commodore'a sanki doğrudan doğruya bilgisayardan geliyormuşcasına bağlanabilmektedir. Özel bir komutla C64'ün grafik ve yöneltim simgeleri bile doğru olarak verilebilmektedir. Bu çözümün bir sakıncası, C64'ün bir parçasını sizin bizzat değiştirmek zorunda olmanızdır (bunu öylece çekip çıkarmanız ve bir başkasıyla değiştirmeniz yeterlidir aslında). Ama herkes bu önlemi alma cesaretini bulamaz kendinde, özellikle de bilgisayarın garanti süresi henüz dolmamışsa... Bu durumda bir Donanım-Interface'i edinmek en iyisidir. Ama bu konuda da çok değişik modeller, formlar ve verim düzeyleri ile karşılaşıldığından, büyük bir çeşitlilik göze çarpar. Uzun süreli testler sonucunda esas olarak üç tipin kalite olarak diğerlerinin önüne geçtiğini saptamış bulunuyoruz: Bun-



Resim 7: Citizien 120D her sistemi değerlendirir.



Resim 8: Riteman II yerden tasarruf sağlar.

Commodore 64 hafızasının azlığına rağmen metin işleme açısından son derece zengin ve kullanışlı seçeneklere sahip.

lardan ikisi HDS-Interface'i ile Görlitz-Interface'i, üçüncüsü de yüksek düzeyde esnekliği ile dikkat çeken Wiseman-Interface'idir. Bütün Interface'lerin devreye sokulmasında Interface'in kullanımdan önce belli bir duruma getirilmesinin zorunlu olabileceğini unutmamak gerekir. Kural olarak bu tür "dilmaç"ların hoş olmayan bir özellikleri vardır: Bunlar değişik metin işleme programları sırasında iyi niyetle gereğinden çok şey yapmaya kalkışırlar. Kendi yöneltim mantıklarıyla ve hiç de istenmediği halde CBM-Norm'un verilerini ASCII-Norm'unkilere dönüştürürler. Bu durumda, Interface'in bütün fonksiyonlarını durdurmaktan başka çare yoktur. Neyse ki bizim tavsiye ettiğimiz Interface'lerin üçünün de linear birer kanalı vardır. Bu kanalda veriler herhangi bir dönüşüme uğratılmaksızın aktarılabilirler. Bu sistemin fiyatı, kullandığınız programa ve seçmiş olduğunuz Interface'e bağlı olarak 4000 ile 4180 Mark dolayındadır.

Bir işletim sistemi olarak MS-DOS, metin işleme alanında değişik ve avantajlı seçenekleri yazarlara sunmaktadır.

TASARRUF PAKETİ

Şimdi tanıtılacak olan sistem (resim 5) çok ucuz olmasına rağmen yazının kalitesi açısından en pahalı sistemle aynı sonucu verir. Bunun nedeni, sistemin kullandığı yazıcıdır. Çünkü DWX 305 yazıcısı C64'de de elbette kullanılabilmektedir. Burada çoğunlukla gözardı edilen bir gerçek kendini açığa vurur: Yazının kalitesinden yalnızca bilgisayara bağlanmış olan yazıcı sorumludur. Bu sistemle basit bir matris-yazıcısıyla PC-Sistemi'nkinden daha güzel bir yazı elde edilebilir. Bu sistemde renk istenmediğini varsaydığımızdan, BMC BM 12 ES gibi bir monokrom monitörün de burada pekâlâ iyi iş görebileceğini söyleyebiliriz. Metin programları olarak daha önce sözü edilenler bu sistemde de hizmetinizdedir ve kesim-bölgeleri bakımından da aynı çeşitlilik sözkonusudur. Ama giderlerden tasarruf etmek istediğimiz için, Startexter'in ve basit User-Port'un kullanılmasını tavsiye etmek zorundayız. Bu sistem gerçekten bir kelepirdir. Çünkü 2423 Mark karşılığında emsalsiz bir yazıya kavuşmuş olursunuz.

İKİ REFERANS SİSTEMİ

Ancak bilgisayarlarına bağlanmış olan yazıcıları ayırdedebileceğiniz bu iki sistemi (resim 6 ve 7) de fiyatı ucuz sistemler arasına katabilirsiniz. C64'den, 1541-Floppy-Disket sürücüsünden, 1702-monitörden ve biri Citizien 120'den (1000 Mark dolayında), diğeri Star SG 10'dan (1400 Mark dolayında) oluşan bu sistemler çok iyi yazı kaliteleri yanısıra matris-yazıcıların esnekliğini de bünyelerinde toplamışlardır. Ama bu yazıcılarla çok güzel yazı yazmanın dışında nefis desenler ve grafik çizimleri de gerçekleştirilebildiğinden, Interface konusunda ya Kernel-versiyonunu ya da yukarıda sözü edilen donanım kesim bölgelerini tercih etmek gerekmektedir. Bu noktada her durumda şundan emin olabilirsiniz: Yazı ile esnekliğin anlamlı bir bileşimi için harcadığınız tek bir kuruş boşa gitmemiştir. Citizien 120D'de sistemin fiyatı 3142 Mark, Star SG 10'lunun ise 3340 Mark'tır.

YERDEN TASARRUF

Biraz önce söylenenler Riteman II için de (resim 8) büyük ölçüde geçerlidir. Ama bu sistem ayrıca önemli bir avantaja da sahiptir. Çünkü Riteman II, yer konusunda gerçekten tasarruf sağlayan bir sistemdir. Her tür yazı masasına sığdırılabildiği gibi, sumen için de yer bırakır. Bilgisayar masasını bir de kocaman yazıcıyla doldurmak istemeyen herkes için ideal bir sistemdir bu. Yazının



Resim 9: Biraz hile ile MPS 802 ile işler yoluna girer.



Resim 10: Geleceğin Sistemi- Düşten öte bir şey mi?

kalitesine ve esnekliğine gelince, bu küçük yazıcının yeteneği hemen kendini belli eder. NLQ-yazı kalitesine sahip olan Riteman II, grafik çizimine de tam olarak elverişlidir. Fiyatı ise son derece makuldür (1300 Mark). Interface'ler ve metin programları açısından büyük bir uyum yeteneği gösterir, çünkü komut donanımı FX-85 ile hemen hemen özdeştir. Önerdiğimiz metin programlarıyla tam bir uyum içinde çalışır. Fiyatı 3542 Mark'tır.

EV OYUNU

Sondan bir önceki bu sistem (resim 9) bütünüyle Commodore markasını taşıyan bileşenlerden oluşmuştur. Ama bu sistem aynı zamanda küçük bir hile ile metin işleme için uygun bir duruma da getirilebilir. Bizim görüşümüze göre, bir metin sistemi için Almanca'daki "ünlü değişimi" (Umlaut)'a egemen olmak mutlak bir zorunluluktur. MPS 802'yi tanıyanlar, onun bu özelliğe sahip olmadığını bilirler. Buna rağmen MPS 802 çok iyi bir yazıcıdır. MPS'nin bilgisayara bağlanması da hiçbir sorun çıkarmaz. Bir kablo ile Floppy'in arka tarafına takılır ve aynı anda işlemeye hazır bir duruma gelir. Biz şimdi sözünü ettiğimiz hileye gelelim: Tavsiye etmiş olduğumuz iki metin programı MPS 802'ye de Almanca'daki ünlü değişiminin sırrını verebilirler. Bunu MPS 802'nin serbest tanımlanabilen bir grafik simgesiyle başarırlar. Sonucun görece iyi olduğu söylenebilir. Ama en iyisi, dergimizin 5/85 sayısında gösterdiğimiz gibi, MPS 802'nin yapısında bir değişikliğe gidilmesidir. Yazılımla ilgili bu küçük hile gerek Startexter, gerekse Textomat Plus tarafından da gerçekleştirilebilir. Vizawrite 64 ise ne yazık ki özel bir simge tanımı yapabilecek güçte değildir. Bu sistemin toplam olarak fiyatı 2800 Mark'tır. MPS 802'nin yapısında bir değişiklik yapılmasına gelince, bu iş uzmanlara düşmektedir kuşkusuz.

DÜŞLER

Son sistemimizde (resim 10) işin içine biraz hayal karışmaktadır. Şöyle düşünün: Renkli 80-simgeli görüntü, çok iyi bir klavye, yaklaşık 880-KByte depo kapasiteli bir ila iki disket sürücüsü ve kullanımı basit, verim düzeyi yüksek, buna rağmen ucuz bir metin programı... Program ve metinler için elimizin altında yalnızca 100 ya da 200 KByte değil, mümkünse yarım Megabyte olsun. Bunlara ek olarak doğrudan doğruya bilgisayara bağlanabilen ve olağanüstü hızına rağmen mükemmel bir yazı karakteri sunabilen bir yazıcı. Yakın bir zamanda böyle bir şeyin mümkün olamayacağını mı düşünüyorsunuz? Belki de olacaktır. Çünkü bir hayalin arkasında her zaman bir parça gerçek saklıdır. Kendinizi sürprize hazırlamalısınız. Muhtemelen yakın bir gelecekte bu hayalin ardında ne miktarda hakikat yattığını ve bunları mümkün kılan bilgisayarın adını okuyabilirsiniz.

SATIN ALMA KONUSUNDA ÖNERİLER

"Ama bunlar oluyor" diyesi geliyor insanın... Çünkü yukarda betimlenen bütün sistemler sınanmışlardır ve eğer birkaç küçük nokta ihmal edilmezse çalışmaktadırlar da. Bu sistemin kullanımının temel koşulu, hangi fonksiyonların harekete geçirilmek istendiğinin bilinmesidir. Ama bunun için sisteme ilişkin bütün el kitaplarının incelenmesi kaçınılmazdır. Özellikle günümüzdeki sistemlerin yüksek verim düzeyleri için önemli olan şey, bunların doğru kullanımlarıdır. Eğer siz henüz şu ya da bu sistem konusunda kararınızı vermiş değilseniz, size bununla ilgili bir sır verebiliriz: Sisteminizin donanım bölümünü yazılım bölümünden önce satın almayınız. Pazara sunulmuş olan metin programları konusunda, örneğin test raporları yoluyla bilgiler toplayınız ve sizin taleplerinizi en iyi şekilde karşılayabilecek olanı seçiniz. Bir program için kesin kararınızı verdiğiniz takdirde, satıcınızdan buna uygun bir sistem oluşturmasını isteyiniz ve bunun size doğrudan sunulmasını sağlayınız. Eğer satıcı buna hazır değilse, onun ilerdeki servis hizmetlerinde de muhtemelen yeterli olmayacağını unutmayınız. Yalnızca nitelikli uzman firmalardır satıştan önce ayrıntılı ve dürüst bir danışmanlık hizmetini garanti edenler. Ne denli şaşırtıcı gelse de, uzman bir firmada danışmanlık ve daha sonraki hizmet için yapmış olduğunuz her masraf bir yatırımdır. Gerçi aynı malı biraz daha ucuza satan pek çok mağaza bulabilirsiniz, ama çoğunlukla bunun bedelini satıştan sonra onlar tarafından yalnız bırakılmış olmakla ödersiniz. Bir kahve makinesi söz konusu olduğunda bunun bir sakıncası olmayabilir, ama bilgisayar tekniği gibi karmaşık bir alanda durum bütünüyle farklıdır.

Micro Showy

bilgisayar merkezi

TUNALI HİLMİ CADDESİ SEĞMENLER ÇARŞISI NO: 96/59 KAVAKLIDERE-ANKARA TEL: 168 36 47

1987 OYUNLARI 500.-TL.

- DİSK KUTULARI, DİSKETLER, JOYSTİCKLER, DİSK DELİCİLER, LIGHT PEN, KORUYUCU ÖRTÜLER.
- OKUL ÖNCESİ, İLKOKUL, ORTAOKUL, LİSE, ANADOLU LİSELERİ VE ÜNİVERSİTE HAZIRLIK.
- 1 KASETTE 20 SÜPER OYUN: 5.000.-TL. 1 KASETTE 12 SÜPER OYUN: 4.000.-TL.
- 4.000'DEN FAZLA OYUN, İŞ VE EĞİTİM PROGRAMLARI.
- TÜM OYUNLARI VİDEO VE MONİTÖR GÖSTERİLERİMİZDE İZLEYEREK ORİJİNAL RESİMLERİ İLE ALINIZ.
- AYIN SON OYUNLARI LİSTESİ İSTEYENLERE ÜCRETSİZ GÖNDERİLİR.
  - ▶ İSTEKLERİN FAZLA OLMASI NEDENİYLE **90 SAYFALIK** OYUN, İŞ VE EĞİTİM PROGRAMLARI LİSTENİZİ 900.-TL. POSTA PULU KARŞILIĞI GÖNDEREBİLİYORUZ.

BAYİLİK VERMEYE DEVAM EDİYORUZ
31.7.1987 TARİHİNE KADAR UZATILMIŞTIR

İSTEKLERİNİZİ ÖDEMELİ GÖNDEREBİLİRİZ

COMMODORE

Dünyanın en kolay kelime işlem paketi "Easy" şimdi Türkçe

Artık bilgisayarınıza* Türkçe komut verebilirsiniz. Komdata, MikroPro'nun kullanımı en kolay ve en yetenekli kelime işlem paketini Türkçeleştirdi. **Bu paketi kullanmak neden bu kadar kolay?** 1. Hiçbir komutu ezberlemenize gerek bırakmayan menüler içerir. 2. Herhangi bir komutu kullanırken, o komut hakkında ayrıntılı bilgiyi anında görüntüler. 3. Her işlem sırasında sizi adım adım yönlendirir. 4. Yanlışlıkla silinen bir satırı veya paragrafı geri getirebilir. 5. Sınırsız uzunlukta ve 240 sütun genişliğinde dokümanlar yaratmanıza izin verir. 6. Wordstar ve Wordstar 2000 ile yazılan dokümanları kullanabilir. 7. Lotus 123 ve Symphony dosyalarını doğrudan okur. 8. 250'den fazla yazıcının özelliklerini tanır. **Klavyenizle Türkçe komut vermenin rahatlığına biran önce kavuşmak için bizi 1330342'den hemen arayın.**

*Commodore PC ve uyumluları

KOMData

KOMDATA İLETİŞİM SİSTEMLERİ LTD. ŞTİ.



Metin İşlem'e Giriş: Wordstar ve Vizawrite

SAMET KUMAŞ

Bu yazının amacı PC'lerde kullanılan metin-işlemci programlarını birbirleriyle karşılaştırarak, hangi program paketinin sizin amaçlarınıza en uygun olanı olduğuna karar vermenizde yardımcı olmak. Biz size tanıtacağımız bütün programları bir Commodore AT üzerinde denedik. Ancak sözkonusu programlar Commodore AT ile uyarlı diğer bütün makinelerde ve Commodore PC 10 ve PC 20'de de çalışabilen programlar... Bunlar sırasıyla, Visawrite, Wordstar, Word, Moonstar ve Q&A. Bunlardan ilk üçü bugün artık bütün metin-işlem programlarında bulunması beklenen olanakları sunan standart programlar. Diğerleri ise özel bazı ihtiyaçları karşılamak üzere özel yeteneklerle donatılmış, nisbeten uzman programlar...

Ancak gereksiz tekrarlardan kaçınmak için, bir yandan standart diye andığımız programları tanıtırken, bir yandan da bir metin-işlem programı nedir, ne yapar, bütün metin-işlem programlarından beklendiğini söylediğimiz şu genel özellikler nelerdir, onlara bir giriş yapalım, dedik. Bu standart özellikler, elbette aynı zamanda çeşitli programları karşılaştırırken birer ölçüt işlevini görüyorlar. Bu yüzden, bir yandan metin-işlem programlarının genel özelliklerini anlatmak, aynı zamanda standart diye andığımız paket programların, çeşitli ölçütler açısından "ideal metin-işlemci"ye ne ölçüde yaklaştığını gözden geçirmek anlamına gelecek.

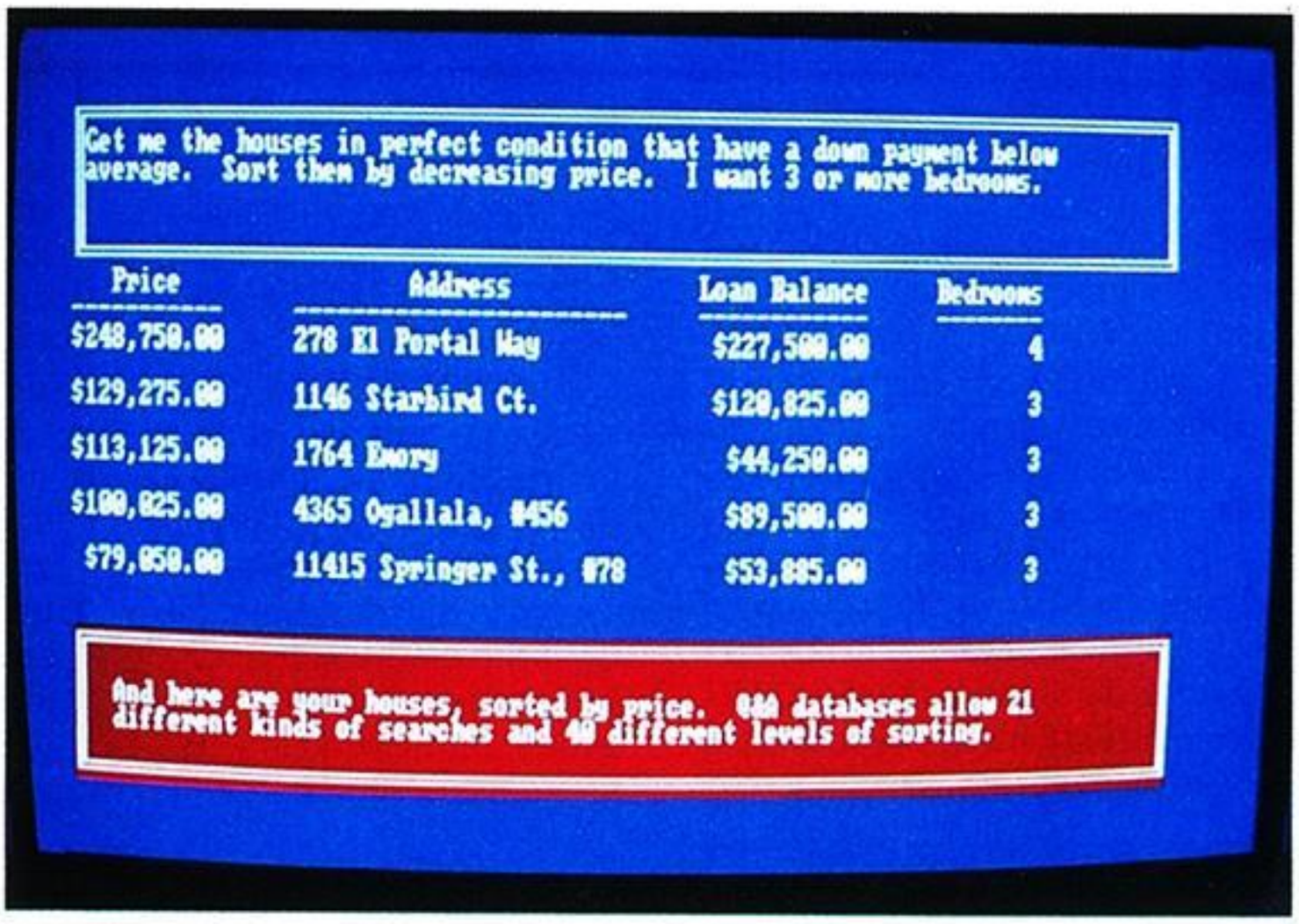
Nedir o halde bir metin-işlemci programı? En kaba tarifiyle metin-işlemci, bilgisayarınızdan bir daktilo gibi yararlanmanızı mümkün kılan bir paket programdır. Ancak metin-işlemci programlarının bugün ulaşmış oldukları gelişkinlik seviyesini gördüğünüzde, böyle bir tarifin ne kadar büyük bir haksızlık olduğunu siz de farkedeceksiniz.

Ancak biz yine de başından başlayalım. Bir yazıyı, mektubu ya da belgeyi daktiloda yazmaktansa bir metin-işlemci programı ile yazmanın ilk akla gelen avantajları nelerdir? Bir kere belgenin manyetik bir ortamda saklanıyor olmasının sağladığı bütün avantajlar - yazı kalitesinden fedakârlık etmeksizin belgeyi istediğiniz kadar çoğaltabilme olanağı; masanızın üstünü, çekmecelerinizi istila etmekle yetinmeyip evinizin, büronuzun her köşesine yayılma istidadı gösteren kâğıt karmaşasından kurtulma olanağı vb. Bir de -belki fazla öznel kaçacak ama- benim teknoloji korkumu yenerek, bir bilgisayar tutkunu olmama yolaçan bir avantajı var metin-işlemcilerin. Eğer siz de benim gibi, yazmaktan hem çok zevk alıyor, ama hem de ak kâğıt önünde ödünüz patlıyor, paralize oluyorsanız, metin-işlemci programları sizin de Frenklerin "yazma blok"u dedikleri hastalıktan kurtulmanızı sağlayabilir. Tuhaf bir şekilde ekran, beyaz kağıdın yarattığı kalıcılık, nihailik, geri dönülmezlik izlenimlerini yaratmıyor; bir metin-işlemci kullanarak yazı yazmak ciddi ve önemli bir iş olmaktan çıkıp giderek bir oyuna benzemeye başlıyor.

Ama artık bu genel lafları bırakıp, bir metin-işlemci programı neler yapar/yapmalıdır, onun ayrıntılarına girmeye başlayalım.

YAZI GİRİŞİ VE DÜZELTMELER

Diyelim ki herhangi bir metin-işlemci programını aldık ve çalıştırdık. Bence ilk yapılması gereken, bu aşamada el kitabı ile fazla uğraşmadan, birtakım yazılar girmeye başlamaktır. Vizawrite programında, programı çalıştırdığınızda doğrudan doğruya yazacağınız "sayfa" ile karşılaşıyorsunuz; dolayısıyla hemen işe koyulabilirsiniz. Wordstar'da ise "ana menü" (Main Menu) denen bir komutlar listesi ile... Şu anda bizi ilgilendiren buradaki D komutunun yanında "open a document file" yazıyor. Bu bağlamda metin-işlemci programlarının genel özelliklerinden ilkini görebiliriz. Bu programlar yazılarınızı, belgelerinizi her birinin kendine özgü bir adı olan, birbirinden bağımsız dosyalar halinde saklar. (Bu dosyalar kendi aralarında ilişki de kurabilir -ama buna daha sonra geleceğiz.) Dolayısıyla, Wordstar'da yazmaya başlamak için D komutunu kullanmanız, yani kendinize bir dos-





ya açmanız gerekiyor. Makine size dosyanızın adını soracak, siz de adını verip Return'e basar basmaz, ekranı değiştirecektir. Artık karşınızda görmekte olduğunuz boş ekranı sayfanız olarak yorumlayabilirsiniz. Sayfanın üzerinde ise yeni bir menü, Ana Menü (Main Menu) diye anılan menüyü göreceksiniz. Artık yazmaya başlayabilirsiniz.

Yazmaya başlar başlamaz, daha ilk satırın sonunda metin-işlemci programlarının genel özelliklerinden birini göreceksiniz. Satırın belli bir noktasına, daha açıkçası sağ marj olarak tespit edilmiş noktasına gelir gelmez, yazı bir alt satırdan devam edecektir- yani, her satırın sonunda Return tuşuna basmanız gerekmiyor. Bu özellik "wordwrap" adıyla biliniyor.

Gözden geçirmekte olduğumuz her iki program da "wordwarp" olanağını sağlıyor. Yine de aralarında bir fark var. Wordstar "wordwarp"ten kurtulmanıza da, yani istediğiniz takdirde sağ marjda siz Return tuşuna basana kadar devam edecek bir satır girmenize olanak tanıyor. Bunun için önce ekranın tepesinde görülen ana menüdeki seçeneklere bir göz atmanız gerekiyor. Orada en sağda, bir dizi diğer menünün adını göreceksiniz. İlk olarak Ekran Düzenlemesi Menüsü diye çevirebileceğimiz "Onscreen Menü"yü seçmeniz gerekiyor. Bunu aynı anda CTRL ve O tuşlarına basarak yapıyorsunuz. (Önemli not: Wordstar'da bütün komutlar CTRL tuşu ile birlikte bir başka tuşa basarak veriliyor.)

Bunu yaptığınız anda sayfanızın tepesindeki menü değişecektir. Yeni menüdeki seçeneklerden biri CTRL W harfleri ile belirtilen Wordwarp Off (ekranda Wrd wrap Now ON şeklinde gösteriliyor). Buna basar basmaz Now ON yazısı Now OFF'a dönüşecek ve satırları istediğiniz yerden bölebilmeye başlayacaksınız.

Maalesef Vizawrite benzer bir olanak sağlamıyor. Bu ilk bakışta önemsiz görünüyorsa da, aşağıda da göreceğimiz gibi yazınıza vermek isteyebileceğiniz sayfa düzenleriyle ilgili önemli bir kısıtlama getiriyor.

SAYFA GEZİNTİLERİ

Bir süre yazdıktan sonra, her kul gibi siz de mutlaka bir hata yapmış olduğunuzu farkedeceksiniz. Şimdi ne olacak?

İlk yapmanız gereken şey hatanın bulunduğu noktaya geri dönmek. Her iki programda da bu amaçla kullanacağınız temel kaynak, klavyenizin sağ yanında diğer tuşlardan ayrı olarak duran, rakam ve ok tuşları. Bu tuşlar sayesinde metin üzerinde karakter karakter (harf harf) sağa sola, ya da satır satır yukarı aşağı gezinebilirsiniz.

Ama binlerce satırlık bir metinde karakter karakter ya da satır satır ilerlemek insanın sabrını tüketebilir. Bu yüzden her iki program da, sayfa gezintilerinizde karakter ya da satırdan daha büyük metin parçalarını da bir birim olarak kabul etmenize olanak tanıyor. Bu daha büyük birimlerden birincisi kelimeler -her iki programda da metin üzerinde kelime ileri ya da geri gitmeniz mümkün. İş daha da büyük birimlere gelince, iki program birbirinden kısmen farklı mantıklar kullanıyorlar. Wordstar'da Ana Menü'den ulaşabileceğiniz CTRL R ve C komutları ile bir ekran dolusu metni tek birim olarak kabul edebiliriz. Yani CTRL R komutuna bastığınızda, hazırlamakta olduğunuz belgede bulunduğunuz yerden bir ekran dolusu metin kadar önceki bir yere geri dönmeniz mümkün.

Vizawrite ise belgenizi, tıpkı sizin de muhtemelen yapacağınız gibi, sayfa sayfa düzenliyor. Böylelikle tek bir komutla bir önceki sayfaya, ek bir komutla da o sayfanın sonuna ulaşabiliyorsunuz. Tabii bunu yapmak için önce sayfa kesim noktalarını tesbit etmiş olmanız gerekir. Bunu da metin girişinin istediğiniz herhangi bir noktasında önce CTRL sonra P tuşlarına basarak yapabiliyorsunuz. Bence Vizawrite'in Wordstar'a üstün olduğu nadir yönlerden biri bu ekran gezintileri mevzuu. Bana sayfa sayfa hareket etmek, bir ekran dolusu metni bir birim olarak kabul etmekten daha mantıki geliyor.

Ama bunu söyledikten sonra da, bu sefer Wordstar'ın bir üstünlüğüne değinmemiz gerekiyor. Vizawrite'de aynı komutu tekrarlamak istediğinizde, aynı işlemi tekrarlamaktan başka bir çareniz yok. Yani diyelim ki, bir metinde 20 sayfa geriye gitmeniz gerekiyor. 20 kere Pg. Up tuşuna basacaksınız demektir. Oysa Wordstar'da bu komutun tekrarlanmasını istediğinizde, komutun kendisini vermeden önce CTRL ve sonra da iki Q tuşuna basmanız yeterli. Bunları yaptıktan sonra vereceğiniz komut, ila-nihaye, yani siz onu durdurana kadar tekrarlanacaktır. Deminki örneğimizde, 20 sayfa geriye gitmeniz gerekiyorsa, CTRL Q Q ve R tuşlarına basmanız yetecektir. Bundan sonra metnin istediğiniz bölümü ekranda belirdiği anda, boşluk çubuğuna basarak komutun işleyişini durdurabilirsiniz.

DÜZELTMELER: INSERT VE DELETE

Bunca lafazanlıktan sonra hâlâ hatırlıyor musunuz bilmem ama sayfa gezintileri hakkındaki açıklamalarımız başlamadan önce bir hata yapmış olduğumuzu farzetmiştik ve bu hatanın nasıl düzeltilebileceğini ince-

lemek üzereydik. Şimdi de artık bunca açıklamadan sonra, hata yapmış olduğumuz yere geri dönebilmiş olduğumuzu farzedeceğiz. Şimdi ne olacak?

Bu noktada bütün metin işlemci programlarında bulunan, iki temel metin (ya da veri) giriş tarzı arasındaki farkı öğrenmeniz gerekiyor. Bu farkı yine bir örnek üzerinde açıklayalım:

Diyelim ki yapmış olduğunuz hata "gitti" yerine "geldi" yazmaktı. Bu takdirde sorun yok; kursoru "geldi" kelimesinin başına getirip "gitti" yazmanız yeterli. Yeni harfler eski harflerin üstüne yazılacaktır.

Ama ya yaptığınız hata bir kelimeyi unutmaktıysa? Yani, örneğin "hemen geldi" yazacağınız yerde "geldi"

Metin-İşlem programlarının en büyük armağanı yazma sürecini özgürleştirmesi, zenginleştirmesi ve zevkli hale getirmesi.

yazmışsanız?.. İşte o zaman "insert mode" denilen metin giriş tarzına ihtiyacınız var demektir. Bu tarzda metin girerken, yeni girdiğiniz metin eskiden orada olanın üzerine yazılmaz, onu zorla sağa doğru iterek kendisine yer açar. Böylelikle girmeyi unutmuş olduğunuz metin parçalarını doğru yerlerine girmenize olanak tanır. Bu yazı giriş tarzının kullanılış kolaylıkları ve olanakları bakımından, benim görebildiğim kadarıyla Wordstar ile Vizawrite arasında, birini ötekine tercih etmenizi gerektirecek önemli bir fark yok.

Buraya kadarki düzeltmeler hep yeni metin girme gerektiren düzeltmelerdi. Ya birşeyin silinmesi gerektiğinde ne olacak?

Klavyenizde bu işlevi yerine getiren iki tuş zaten var: () tuşu ile Del tuşu. Bunlar kursorun halen üzerinde bulunduğu harfin silinmesini sağlarlar. Harf silindikten sonra yerinde bir boşluk kalmaz, eski harfin bir yanındaki karakter silinen harfin yerini alır.

Ama metin-işlem programları bu tuşların sağladığı olanaklara ek bir olanak sağlar. Çok daha büyük metin parçalarını tek (ya da iki) parmak hareketi ile silme olanağı... Bu bakımdan da Wordstar'ın sunduğu olanaklar Vizawrite'inkilere kıyasla daha zengin. Tek komutla bir karakter, bir satır ya da bir blok silmek mümkün. ("Blok" Wordstar'a ait teknik bir terim- metin-işleme giriş konulu bir yazıda açıklamanın yeri yok diye düşündük. Belki ilerki sayılara-). Oysa Vizawrite'da ne kadar bir yeri silmeyi istediğinizi önceden programa bildirmeniz gerekiyor.

VİZAWRİTE VE WORDSTAR

Buraya kadar anlattıklarımız, herhangi bir metin-işlemci programının başında bir saat geçirdiğiniz takdirde, kendiniz uygulamaya başlayabileceğiniz temel özellikler. Aslında bunlara bir metin-işlemci programının sunduğu olanaklar demek bile biraz yanıltıcı. Bu anlattıklarımız sadece, metin-işlemci programı bilgisayarınızı bir daktiloya çevirdiğinde ek olarak neleri mümkün kılıyor? sorusuna cevap veriyor. Yoksa metin-işlemci programlarının gerçekten kendine özgü yönlerinin hiçbirine (sayfa düzenleri, metin parçalarını arama ve yerlerini değiştirme, veri tabanları ile ilişki kurma vb.) değinmedik. Bunlardan dergimizin ilerki sayılarında sözedeceğiz.

Ama şimdi, daha gelişkin özellikler de gözününde bulundurulduğunda, tanıtmakta olduğumuz iki programdan hangisi tercih edilmeli, onun üzerinde duralım.

Daha gelişkin olan programın Wordstar olduğu konusunda sanırım oldukça yaygın bir fikir birliği var. Wordstar bugün en çok kullanılan metin-işlemci programlarından biri... Daha ilerki sayılarda tanıtacağımız ve nisbeten daha uzmanlaşmış Moonstar gibi programlar da, aslında Wordstar'ın geliştirilmiş biçimleri.

Wordstar'ın Vizawrite'a olan üstünlüklerinin bir kısmına metin içersinde değindik. Bunlara ek olarak, Wordstar'ın sunduğu sayfa düzenleme olanaklarının çok daha zengin olduğunu söyleyebiliriz. Ayrıca Wordstar size belli bir metin üzerinde çalışırken başka bir dosyaya ulaşmanız Wordstar'da daha kolay.

Gelelim Vizawrite'ın üstünlüklerine... Pek emin değilim ama bir avantajı, daha kolay öğrenilebiliyor olması olabilir. Vizawrite'in metni sayfalar halinde düzenlenmesini, Wordstar'in düzenine tercih ettiğimi zaten söylemiştim. Ama Vizawrite'ın en önemli üstünlüğü estetikle ilgili. Bir kere benim gördüğüm Wordstar versiyonlarının hepsi, kullandığınız ekran renkli olsa bile, bu olanaktan yararlanmıyor. Sadece monokrom bir görüntü ile çalışmanıza olanak tanımıyor. Vizawrite'ın güzel renklerine ek olarak ikinci bir estetik üstünlüğü ise pencereleri... Programı kullanırken ekranı değişik pencerelere bölmeniz ve böylelikle yazmakta olduğunuz metni değişik boyda pencerelerden seyretmeniz mümkün. Ben bu pencerelerin metin girişi, arama vb. gibi herhangi bir konuda sağladığı herhangi bir pratik avantajı keşfedemedim. Ama, özellikle her pencereye değişik bir renk verdiğinizde ekranın görüntüsü çok hoş oluyor.



İnsan Düşüncesinde

D.HOFSTADTER

Çeviren: Deniz Erksan

İşe matematiksel mantığın 1931 öncesi tarihini kısaca özetlemekle başlayalım. Tüm bu tarih, akıl yürütme süreçlerini mekanikleştirme girişimleriyle başlamıştı. Şimdi, akıl yürütme yeteneğimizin bizi diğer canlılardan ayıran şey olduğu sık sık söylenir; bu yüzden, en insanî özelliğimizin mekanikleştirilmesi ilk bakışta biraz paradoksal görünüyor. Oysa eski Yunanlıların bile bildiği gibi, akıl yürütme belirli bir kalıba göre işleyen bir süreçtir ve -hiç olmazsa kısmen- ifade edilebilir kanunlara bağlıdır. Aristoteles tasımları (syllogism), Euclid de geometriyi bir kanunlar sistemi halinde toplamıştı; fakat, aksiyomatik akıl yürütme çalışmalarında bunları başka gelişmelerin izlemesi için yüzyıllar geçmesi gerekti.

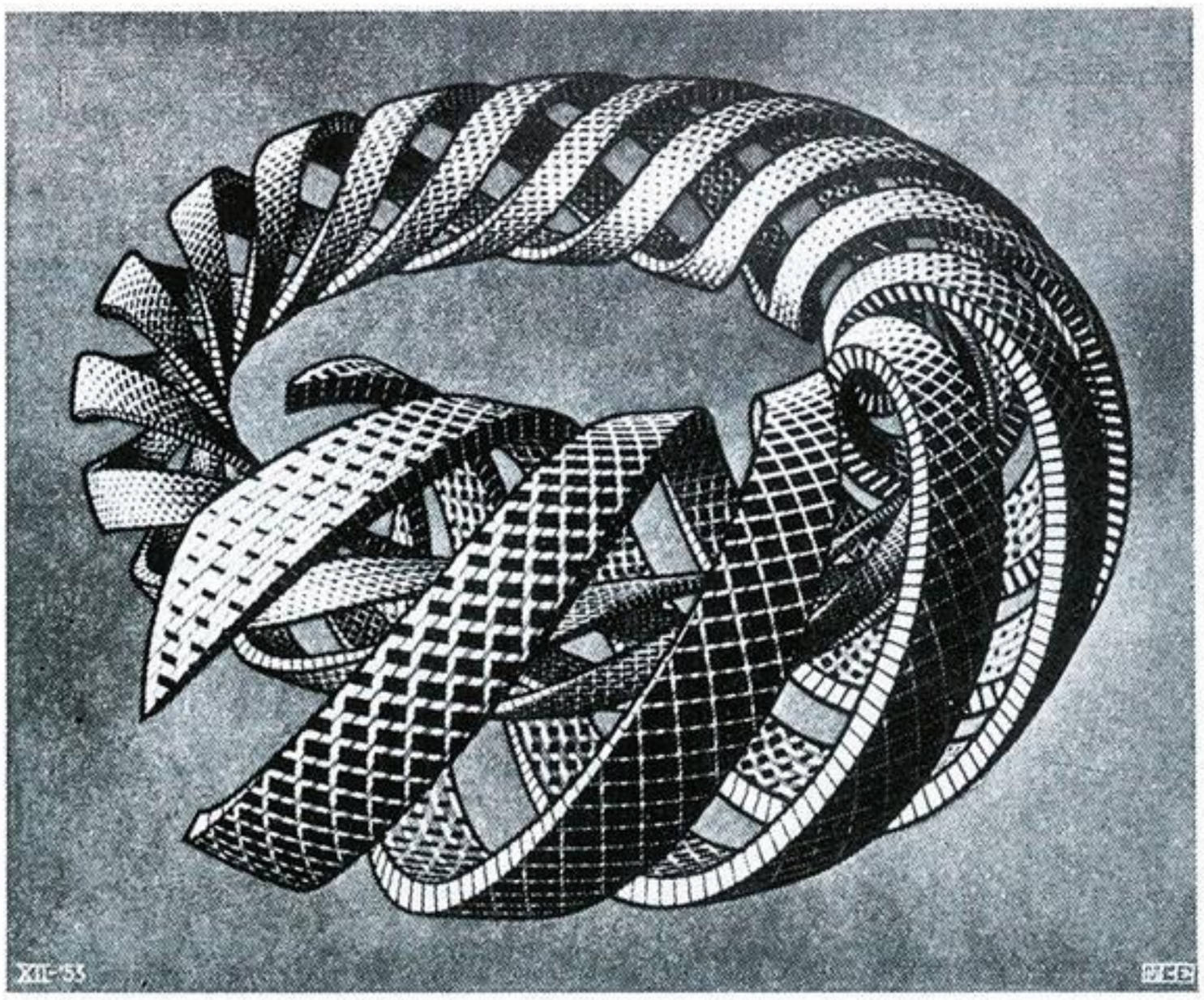
Ondokuzuncu yüzyıl matematiğinin en önemli keşiflerinden biri, aynı derecede geçerli birden fazla geometrinin varlığı oldu. (Burada, "bir geometri" soyut nokta ve çizgilerin özellikleri hakkında bir teori anlamına gelmektedir.) O ana kadar, çok uzun zamandan beri geometrinin Euclid'in sistematikleştirdiği şey olduğu varsayılmış ve Euclid'in sunuş biçiminde küçük kusurlar bulunsa bile bunların önem taşımadığına, geometride gerçek bir ilerlemenin ancak Euclid'i geliştirmekten geçebileceğine inanılmıştı. Hemen hemen aynı anda birkaç kişinin birden Euclid dışı geometrileri keşfetmesi bu inancı paramparça ettiği gibi, matematiğin gerçek dünyayı incelediği fikrine ciddi biçimde gölge düşürdü. Bu, matematikçiler üzerinde de şok etkisi yarattı. Tek bir gerçeklikte nasıl pek çok değişik türde nokta ve çizgi bulunabilirdi? Bugün bu ikilemin çözümü -matematikçi olmayan bazı kişilere bile- aşikâr gelebilir; fakat o sıralarda, matematik çevreleri tam bir yıkıma uğramıştı.

Ondokuzuncu yüzyılın daha ileriki yıllarında, İngiliz mantıkçıları George Boole ve Augustus De Morgan salt tümdengelimsel akıl yürütme kalıplarını sisteme bağlamakta Aristoteles'i bir hayli geride bıraktılar. Hatta Boole kitabına **"Düşünce Yasaları"** adını verdi. Biraz abartmıştı kuşkusuz, ama kitap gerçekten önemli bir katkıydı. Bu mekanikleştirilmiş akıl yürütme yöntemlerine hayran kalan Lewis Carrol, bu yöntemlerle çözülmek üzere pek çok bilmece üretti. Jena'da Gottlob Frege, Torino'da da Giuseppe Peano formel akıl yürütmeyi küme ve sayı çalışmalarıyla bağdaştırmaya uğraşıyorlardı. Göttingen'de David Hilbert ise geometriyi Euclid'e kıyasla daha sıkı ve kesin bir şekilde formelleştirmeye çalışmaktaydı. Tüm bu çabalar aslında, herhangi bir şeyi kanıtlamanın ne anlama geldiğini berraklaştırmaya yönelikti.



Bu arada, klasik matematikte de ilginç gelişmeler cereyan etmekteydi. Georg Cantor 1880'lerde farklı sonsuzluk tipleri üzerine *kümeler (set) teorisi* diye adlandırılan bir kuram geliştirdi. Kuram güçlüydü, güzeldi, fakat sezgilere ters düşüyordu. Çok geçmeden kümeler kuramına bağlı bir sürü paradoks çıkmıştı ortaya.

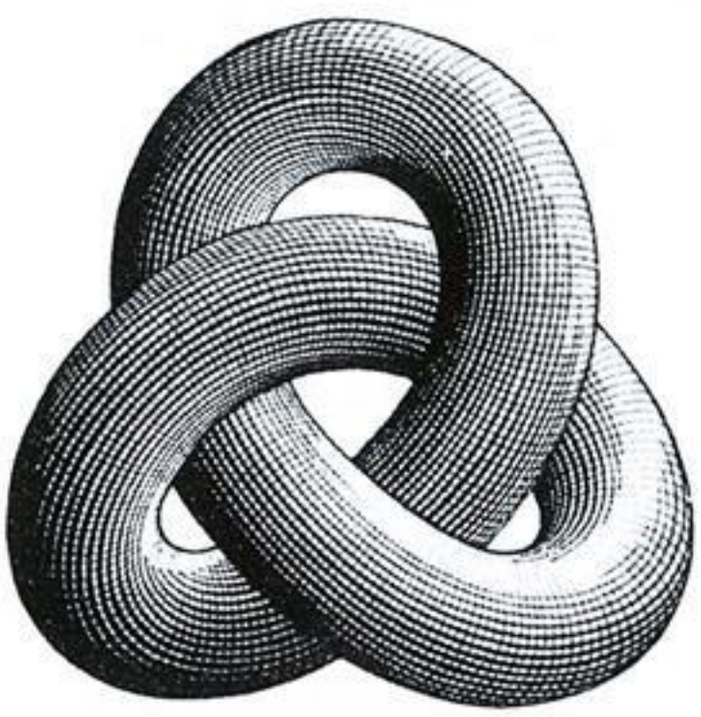
Bunların en tanınmışı Russel'ın paradoksudur. Bildiğiniz gibi, kümelerin çoğu kendi kendisini içermez; yani, köstebekler kümesi (isterseniz "sınıfı" diye de düşünebilirsiniz) bir köstebek; yalnız Jeanne D'Arc'ı içeren küme de Jeanne D'Arc değildir (bir küme bir insan olamaz). Bu açıdan, çoğu küme bir hayli "sıradan"dır. Fakat, kendi kendisinin üyesi olan bir takım "kendini-yutan" kümeler de vardır; örneğin bütün kümeleri içeren küme, yani kümelerin kümesi veya Jeanne D'Arc hariç her şeyi içeren küme gibi... Doğal olarak, herhangi bir küme ya sıradan olabilir, ya da kendisini-yutan türden; hiçbir küme her ikisi birden olamaz. Şimdi oturup R diye bir küme icat edelim: *bütün sıradan kümelerin kümesi.* İlk bakışta, R basbayağı sıradan bir icat gibi görünebilir. Ama



Recursion'ın Yeri

kendinize bir sorun bakalım, "R sıradan bir küme mi, yoksa kendisini yutan türden mi?" Cevabın, "R ne sıradandır, ne de kendisini yutar; çünkü her iki cevap da paradoksa gider" olduğunu göreceksiniz. Deneyin!

Fakat, eğer R ne sıradan, ne de



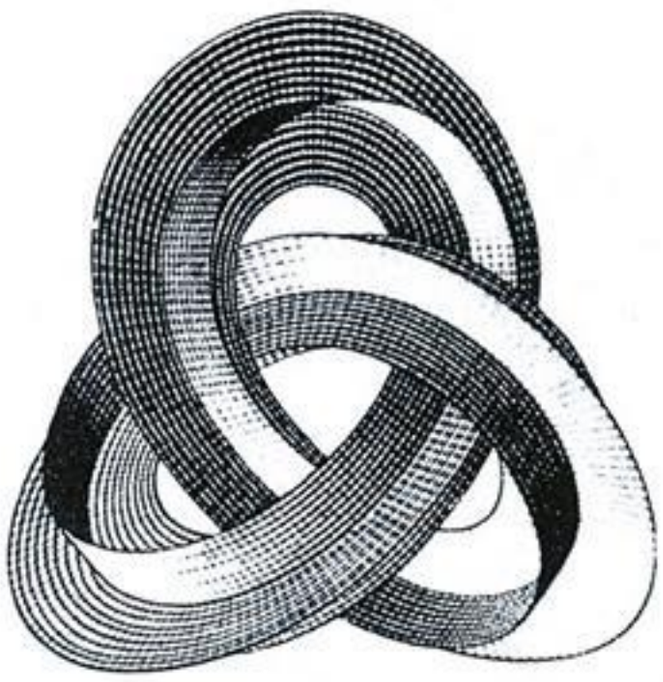
kendisini-yutan türden değilse, nedir? En azından patolojik bir şey. Ama bu tür kaçamak cevaplar kimseyi tatmin etmez, tabiî. Bu yüzden, insanlar kümeler kuramının temellerini daha bir ciddiyetle incelemeye başladılar. Bu incelemede hayati önem taşıyan sorularsa, "Sezgisel 'küme' kavramımızda nasıl bir terslik var? Sezgilerimize gereğince tekabül eden ve paradokslara düşmeyen sağlam bir küme teorisi üretebilir miyiz?" idi. Sayı teorisinde ve geometride olduğu gibi, burada da sorun sezgiyi formelleştirilmiş ya da aksiyomlaştırılmış akıl yürütme sistemleriyle bağdaştırma çabasında yatmaktadır.

Russel'ın paradoksunun şaşırtıcı bir türü de, küme yerine sıfatlar kullanarak yapılan "Grelling paradoksu"dur. Dildeki sıfatları "dörtheceli", "beceriksizli" ve "malumatfuruş" gibi kendini-tanımlayan ve "besleyici", "eksik", "çiftheceli" gibi kendini-tanımlamayanlar olmak üzere iki kategoriye ayıralım. Şimdi, "kendini-tanımlamayan"ın da bir sıfat olduğunu kabul ederek, bu sıfat hangi sınıfa aittir?

Öyle görünüyor ki, bu paradokslarda ortak bir suçlu var, o da kendine-atıf, ya da "kendi üzerine dönme". Yani, bilgisayar programlarından "recursion" diye tanıdığımız özellik. Öyleyse, madem amaç paradoksları tamamen ortadan kaldırmak, neden kendine-atıfı ve buna olanak veren her şeyi menetmiyoruz? Bu öyle göründüğü kadar kolay değil, çünkü kendine-atıfın nerede geçtiğini tam olarak saptamak çok zor olabilir. Mesela, Giritli bir filozofun "bütün Giritli filozoflar yalancıdır" demesini hatırlatan şu iki cümleyi ele alalım:

Aşağıdaki cümle yanlıştır.
Yukarıdaki cümle doğrudur.

Bir arada alındığında, bu cümleler bir paradoks etkisi yaratıyorlar; oysa, ayrı ayrı bakarsak, ikisi de zararsız, hatta potansiyel olarak yararlı cümleler. Bu durumda "suç"u cümlelerden birine yükleyemeyiz. Suç bir-



birlerine "işaret" etme biçimlerinde yatıyor. Aynı şekilde, *İniş Çıkış*'ın her bir bölgesi de pekâlâ doğrudur; imkânsızlığı, bu bölgelerin bütünsel olarak bir araya getiriliş biçimi yaratır. Kendine-atıf yalnız doğrudan değil, dolaylı da olabildiğine göre, her iki türünü birden ortadan kaldırmanın yolunu bulmak gerekiyor - tabiî, eğer kendine-atıfın bütün kötülüklerin anası olduğu düşünülüyorsa.

RECURSION'IN YASAKLANMASI

Russel ile Whitehead böyle düşünüyorlardı; nitekim dev *Principia Mathematica* da kendi kendine atıf yapmayı mantık, küme ve sayı teorisinden kovmaya yönelik bir eserdi. Sistemlerinin ana fikri şuydu: Bütün kümeler bir "tipler hiyerarşisi" oluşturacak şekilde düzenlenmiştir. En düşük "tip"ten bir küme yalnız "nesneler"i içerebilir, kümeleri içeremez. Bunun bir üzerindeki küme yalnızca nesneleri ve en düşük tipten kümeleri içerebilir. Yani, genelde, her küme yalnızca daha düşük tipten kümelerle nesneleri içerebilirdi. Her küme belirli bir düzeydeki bir tipe ait olduğu için, hiçbir küme kendisini içeremez, çünkü bunu yapabilmesi için kendi tipinden daha yüksek bir tipe ait olması gerekir. Böyle bir sistemde yalnızca "sıradan" kümeler bulunur; dahası, bizim eski R -tüm sıradan kümelerin kümesi- sonlu tiplerden hiçbirine ait olmadığından, artık küme olarak kabul edilmemektedir bile. Öyleyse, nereden bakarsanız bakın, "Recursion'ın ilgası kuramı" diye de adlandırabileceğimiz bu *tipler teorisi,* küme teorisini paradokslardan kurtarmakta başarılı olmuştur; fakat, ancak yapay-görünümlü bir hiyerarşi getirmek ve -tüm sıradan kümelerin kümesi gibi- belirli türden kümelerin oluşmasına imkân tanımamak pahasına...

Tipler teorisi Russel'ın paradoksuyla başa çıkmıştı, ama yalancı Giritli filozof paradoksuna ya da yukarıdaki örnekle, birbirine gönderme yapan iki cümleye karşı elinden bir şey gelmiyordu. Küme teorisinden ötesiyle uğraşmayanlar için bu kadarı da yeterliydi; ama, genelde paradoksların elimine edilmesiyle ilgilenenler için, dilde sarmallanmayı yasaklayacak yeni bir "hiyerarşikleştirme" gerekli görünüyordu. Böyle bir hiyerar-

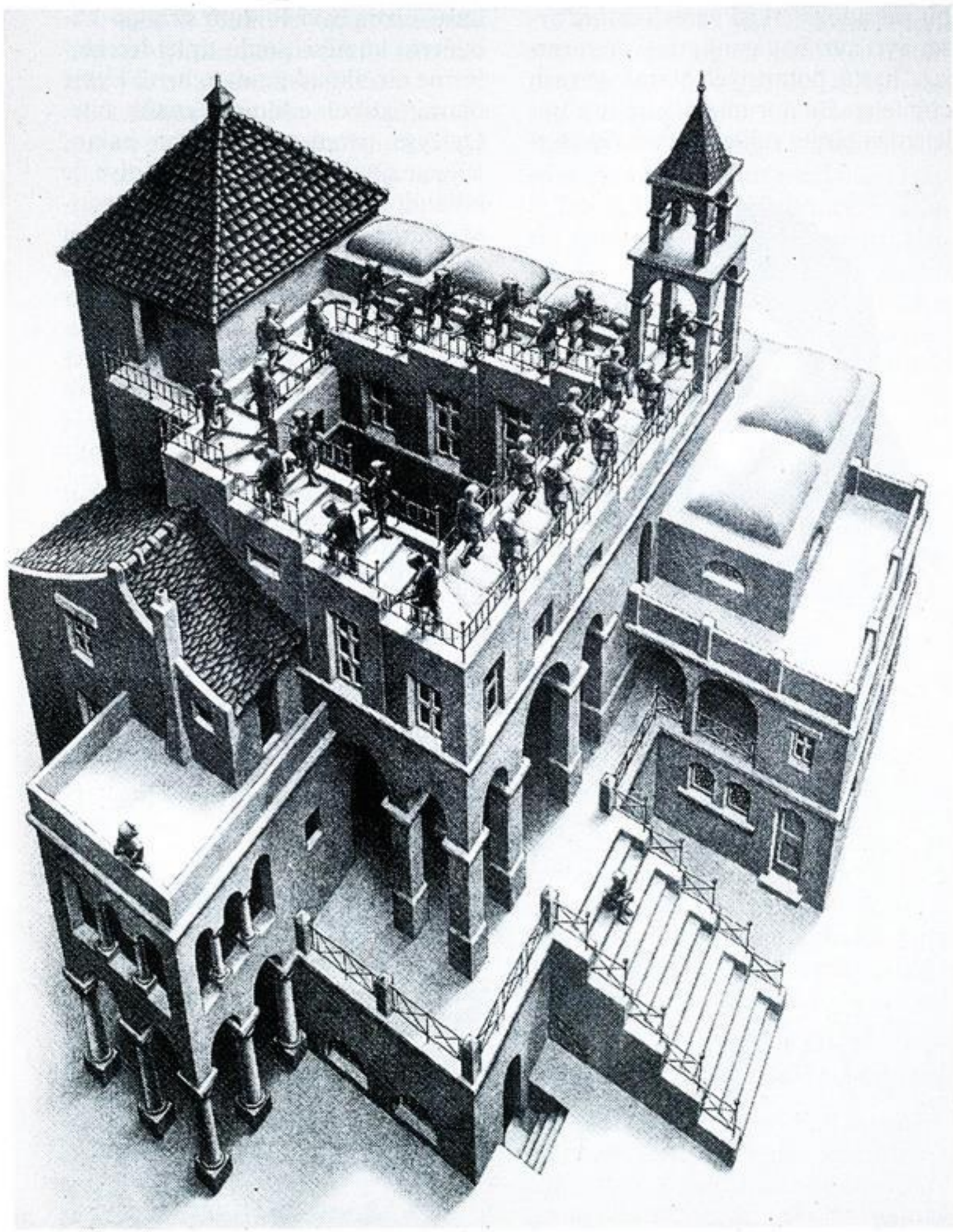


REVİZYON

şinin en altında, yalnızca belirli bir alana atıfta bulunabileceğiniz bir *nesne dili* yer alacaktı. Nesne dilinin kendi özelliklerine (örneğin gramer kurallarına, ya da bu dilde ifade edilen belirli bir cümleye) atıfta bulunmak içinse, bir *üst-dil*'e başvurulacaktı. Yabancı dil öğrenen herkes dilde böyle iki ayrı düzeyin varlığını hissetmiştir. Sonra, üst-dilden sözetmek için bir üst-üst-dil olacaktı, vb. Her cümlenin hiyerarşide belirli bir düzeye ait olması şarttı. Dolayısıyla da, herhangi bir ifade eldeki düzeylerden hiçbirine uymuyorsa, anlamsız addedilecek ve unutulacaktı.

Ama şimdi bizim şu birbirlerine işaret eden iki cümlemizi tahlil etmeye çalışalım. İlk cümle ikincisinden söz ettiğine göre, daha yüksek bir düzeyde bulunması gerekiyor. Ama, aynı şekilde, ikinci cümlenin de ilkinden daha yüksek bir düzeyde bulunması lazım. Bu imkânsız olduğuna göre, her iki cümle de "anlamsız". Daha açık konuşmak gerekirse, katı bir dil hiyerarşisine dayanan bir sistemde bu tür cümleler kesinlikle kurulamaz bile.

Ama dil gibi hayatın tümüne nüfuz eden bir alanda böyle bir hiyerarşi tamamen saçma geliyor. Bir takım şeylerden söz ederken dil hiyerarşisinde bir aşağı bir yukarı sıçrıyormuş gibi görmeyiz kendimizi. "Bu kitapta tipler teorisini eleştiriyorum"gibi gayet basit bir cümle, bu sistemde çifte yasak altındadır. Birincisi, "bu kitap"tan söz etmektedir, ki bunun ancak bir "üst-kitap"ta yapılabilmesi gerekir; ikincisi de, *ben* demektedir, oysa benden söz etmeye hiç mi hiç hakkı yoktur! Bu örnek, tanıdık bir bağlama alındığında tipler teorisinin ne kadar saçma göründüğünü ortaya koymakta. Paradokslara karşı benimsediği çare -her türden kendine-atıfı tamamen yasaklamak- tam bir vur deyince öldürmek vakası sergileyerek, hiçbir kusuru olmayan bir sürü ifadeye anlamsız damgası vuruyor. Bu arada, "anlamsız" sıfatının linguistik tipler teorisiyle ilgili tüm tartışmalar (mesela, şu okuduğunuz paragraf) için de kullanılması lazım, çünkü belli ki bu tartışmalar düzeylerden hiçbirinde -ne nesne dilinde, ne üst-dilde, ne üst-üst-dilde, vb.- geçemezler. O halde, sırf teoriyi tartışma konusu etmek bile, onu düşünülebilecek en ciddi şekilde ihlal etmek olacaktır!



Bu teorilerin, sıradan, günlük dille değil, formel dillerle uğraşmak üzere üretildiği söylenebilir, tabiî. Böyle bir savunu doğru olabilir, ama bu da ancak böyle teorilerin son derece akademik olduğunu ve özel ısmarlama sistemlerde ortaya çıkan paradokslardan gayrısına pek hükümlerinin geçmediğini gösterir. Üstelik, ne pahasına olursa olsun paradoksları elimine etme çabası, hele son derece yapay formalizmler yaratmayı da gerektiriyorsa, tüm ağırlığı donuk bir tutarlılığa vermekte ve hayatı da, matematiği de asıl ilginç kılan dolambaçlı, tuhaf şeyleri çok hafife almaktadır. Tutarlılığı korumak tabiî ki önemli, ama bunun için uğraşırken kendinizi muazzam çirkin bir teorinin içinde bulursanız, bir terslik olduğunu anlamanız gerekir.

İnsanın akıl yürütme yöntemlerini sisteme bağlama konusuna bu yüzyılın başında gösterilen yüksek ilgi, işte matematiğin temelindeki bu tür sorunlardan kaynaklanmıştır. Bu sorunlar özellikle matematiğin mantığın bir dalından ibaret olduğuna (ya da tersine, mantığın matematiğin bir dalından ibaret olduğuna) yürekten inananları etkiliyordu, ki bunların sayısı da hiç az değildi. Nitekim, sırf bu soru -"Matematikle mantık birbirinden ayrık mıdır, yoksa aynı mı?"- bile, yoğun tartışmalar doğurmaktaydı.

Matematiğin bu şekilde incelenmesine üst-matematik -ya da, matematikle mantık böylesine içiçe girmiş olduğuna göre, üst-mantık adı verildi. Üst-matematikçilerin birinci ve en önemli uğraşı, matematiksel akıl yürütmenin gerçek doğasını belirlemek-

ti. Hangi işlem yöntemleri kurallara uygun, hangileri aykırıdır? Matematiksel akıl yürütme her zaman "doğal dil"de (yani Fransızca, Latince ya da normal iletişimde kullanılan herhangi bir dilde) yapılagelmiş olduğundan, hep bir muğlaklık riski taşımıştı. Kelimeler farklı insanlar için farklı anlamlar taşıyor, farklı imgeler çağrıştırıyordu. Her türlü matematiksel çalışmanın yapılabileceği ve herhangi iki matematikçinin önerilen bir kanıtın geçerli olup olmadığı konusundaki anlaşmazlığı çözüme bağlamakta yararlanabilecekleri, düzenli, tek bir notasyon oluşturmak akla uygun ve hatta önemli görünmekteydi. Bunun içinse, evrensel olarak kabul edilebilir nitelikteki akıl yürütme tarzlarının, en azından matematikte geçerli oldukları ölçüde, baştan sona sisteme bağlanması gerekiyordu.

TUTARLILIK VE EKSİKSİZLİK

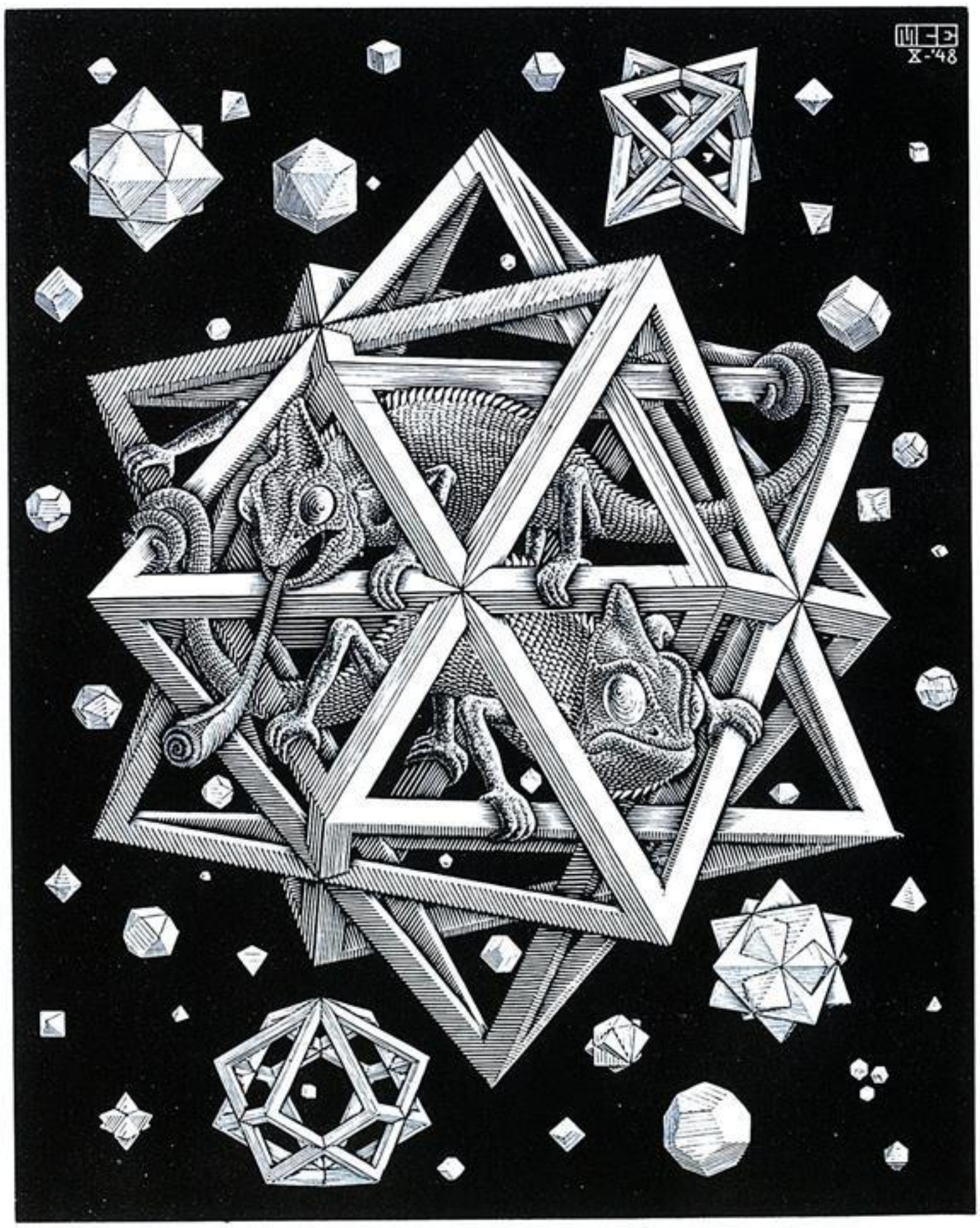
İşte, *Principia Mathematica*'nın hedefi de buydu. Matematiğin tamamını, hem de çelişkisiz olarak, mantıktan türetme iddiasındaki *Principia Mathematica* herkesin hayranlığını kazanmıştı kazanmasına, ama kimse ne (1) Russel'la Whitehead'in belirlediği yöntemlerin gerçekten de matematiğin tamamını kapsadığından, ne de hatta (2) bu yöntemlerin kendi içlerinde tutarlı olduğundan emin olamıyordu. Russel ve Whitehead'in yöntemleriyle, hiçbir matematikçinin *asla* çelişkili sonuçlara varamayacağı her türlü kuşkunun ötesinde kesin miydi?

Bu soruyu özellikle rahatsız edici bulan seçkin Alman matematikçisi (ve üst-matematikçisi) David Hilbert, dünya matematik (ve üst-matematik) çevrelerini, *Principia Mathematica*'da tanımlanan sistemin hem tutarlı (çelişkisiz), hem de *eksiksiz* olduğunu (yani, sayı teorisindeki her bir doğru ifadenin *P.M.*'de çizilmiş olan çerçeveden türetilebileceğini) kesin olarak -hatta belki de Russel'la Whitehead'in belirlediği yöntemleri kullanarak- kanıtlamaya çağırdı. Öyle kolay üstesinden gelinecek bir iş değildi bu; üstelik, bir anlamda döngüsel de sayılabilirdi: akıl yürütme yöntemlerimizi, tam da bu yöntemlere dayanarak nasıl doğrulayabiliriz? İnsanın kendi sırtına tırmanması gibi bir şey bu.

Elbette Hilbert de bu ikilemin tamamen farkındaydı. Bu yüzden, tutarlılığın ya da eksiksizliğin yalnızca 'sonlu' akıl yürütme usüllerine (yani, matematikçilerin çoğunun benimsediği küçük bir akıl yürütme yöntemleri dizisine) dayanarak kanıtlanabileceğini umduğunu belirtmişti. Bu şekilde, matematikçilerin kısmen de olsa kendi sırtlarına tırmanabileceklerini, yalnızca yöntemlerin küçük bir bölümüne başvurarak matematiksel yöntemlerin tamamının sağlıklı olduğunun ispatlanabileceğini umut ediyordu. Bu bize pek dar kapsamlı bir hedef gibi görünebilir; ama, yüzyılımızın ilk otuz yılı boyunca, dünyanın en büyük üst-matematikçilerin birçoğu bu konuyla uğraşıp durdular.

Otuzbirinci yıldaysa Gödel'in makalesi yayınlandı ve Hilbert'in programı bir anlamda tamamen yerle bir oldu. Bu makale yalnızca Russel'la Whitehead'in önerdiği aksiyomatik sistemde onarılması imkansız 'delikler' bulunduğunu açığa çıkarmakla kalmıyor, daha genel olarak, sayı teorisiyle ilgili bütün gerçeklerin de ancak ve ancak tutarsız bir aksiyomatik sistemden türetilebileceğini ortaya koyuyordu. Dahası, makale *P.M.*'de anlatılan türden bir sistemin tutarlılığını kanıtlamaya çalışmanın boşuna olduğunu da göstermekteydi: eğer böyle bir kanıt yalnızca *P.M.*'in içerdiği yöntemler aracılığıyla bulunabilecekse, o zaman -ki bu Gödel'in çalışmasının en garip sonuçlarından biridir- *P.M.*'in kendisinin tutarsız olduğu anlaşılacaktı!

Bütün bunlar yetmezmiş gibi, Gödel Eksiklik Teoremini ispatlarken de recursion'dan kaynaklanan paradoksların her türlü saldırısına karşı kale kadar sağlam olduğu sanılan bir yere, *Principia Mathematica*'nın tam ortasına, Giritli yalancı filozof paradoksunu getirip bırakmıştı! Kısacası, Gödel'in eseri *Principia Mathematica*'yı yıkmadıysa da, Russel ve Whi-



tehead'in başlangıçtaki hedeflerinin hayal ürünü olduğunu ortaya koyarak, üst-matematikçilerin gözünde eserin çekiciliğini son derece azalttı.

BABBAGE, BİLGİSAYARLAR, YAPAY ZEKÂ...

Gödel'in makalesi yayınlandığı sırada, elektronik dijital bilgisayarların icadına da pek bir şey kalmamıştı. Mekanik hesap makineleri yeni bir fikir değildi. Onyedinci yüzyılda, Pascal ve Leibniz sabit işlemleri (toplama ve çarpma) yerine getirecek makineler geliştirmişlerdi. Ne var ki, bu makinelerin hafızası yoktu, yani, günümüz terimleriyle, programlanabilir değillerdi.

Makinelerin hesap alanındaki muazzam potansiyelini ilk farkeden kişi, Londralı Charles Babbage (1792-1871) oldu. Neredeyse Dickens'ın romanlarından fırlamış bir tip olan Babbage, yaşadığı süre içinde asıl ününü Londra'yı 'sokak serserileri'nden temizlemek için yürüttüğü ateşli kampanyaya borçluydu. Bu serserilerin başında da, onu deli etmeye bayıldıkları için gece gündüz kapısının önünde serenad yapan ve adamcağızın her seferinde öfkeyle sokaklara uğramasına sebep olan laternacılar geliyordu. Bugün Babbage'ın çağının yüz yıl ilerisinde bir adam olduğunu görüyoruz: modern bilgisayarların temel ilkelerinin mucidi olduğu gibi, gürültü kirlenmesine karşı da ilk savaş açanlardan biri olmuştu!

İlk icadı olan "Fark Makinesi", "farklar yöntemi"yle çeşitli türlerden matematiksel tablolar üretiyordu. Fakat, daha Fark Makinesi'nin tek bir modelinin olsun üretilmesini beklemeden, Babbage bütün ilgisini çok daha devrimci bir fikir üzerinde yoğunlaştırmıştı bile. Bu fikir, bir "Analitik Makine" tasarımıydı. Bu konuda, pek de mütevazi sayılmayacak bir üslupla, "Herhalde bugüne değin insan zihnini meşgul etmiş en dolambaçlı ve en çapraşık yoldan ulaştım bu fikre" demekteydi. O güne kadar tasarlanmış tüm makinelerden farklı olarak, Analitik Makine'nin hem bir "depo"su (hafızası), hem de bir "değirmen"i (hesap yapma ve karar verme birimi) olacaktı. Bu birimlerse, aklın almayacağı bir karmaşıklıkla birbirine bağlanmış binlerce dişli silindirden oluşacaktı. Delikli kartlarda yazılı bir *program*ın denetiminde akın akın değirmene girip çıkan sayılar canlanmaktaydı Babbage'ın hayalinde. Bu fikrin esin kaynağıysa, kart denetimiyle son derece karmaşık örgüler üreten Jacquard dokuma tezgâhı idi. Babbage'ın zeki ama talihsiz Kontes dostu, (Lord Byron'ın kızı) Lady Ada Lovelace, şiirsel üslubuyla, "Jacquard tezgâhı nasıl çiçekler ve yapraklarla dokuyorsa, Analitik Makine de *cebir örgüleri dokuyor.*" diye yazdı. Ne yazık ki, şimdiki zamanı kullanmakta fazla acele etmişti, çünkü A.M. hiçbir zaman yapılmadı ve Babbage derin bir hayal kırıklığı içinde öldü.

Babbage kadar Lady Lovelace de Analitik Makine'nin icadıyla insanoğlunun mekanik zekâyla oynamaya başlamış olduğunun farkındaydı. Özellikle de, eğer (Babbage'ın, araç kendi depolanmış programını değiştirdiği zaman ortaya çıkan recursion'ı tanımlamak için kullandığı terimle) makine "kendi kuyruğunu yiyebiliyor"sa... 1842 günlüğünde, Lady Lovelace, A.M. "*sayılar*dan başka şeylerle de uğraşabilir," der. Babbage satranç veya üç-taş oynayacak bir otomat yaratmayı hayal ederken, o, Babbage'ın Makine'sinin dönen silindirlerine perde ve armoniler yüklenirse "her kapsam ve karmaşıklıkta, ayrıntılı ve bilimsel müzik eserleri besteleyebileceği"ne işaret etmektedir. Ama, yanlış anlaşılmaya karşı da hemen önlemini alır: "Analitik Makine'nin herhangi bir şey *yaratmak* gibi bir iddiası kesinlikle yoktur. O yalnızca bizim *ona verebildiğimiz emirleri* yerine getirir." Yapay hesabın kudretini gayet iyi anlamakla beraber, Lady Lovelace zekânın yapay olarak üretilmesi konusunda pek iyimser değildi. Fakat, bütün ileri görüşlülüğüne karşın, elektriğin boyunduruk altına alınmasıyla doğacak olan potansiyeli nasıl tahayyül edebilirdi ki?

Bilgisayarların -hem de Pascal, Lebniz, Babbage veya Lady Lovelace'in düşlemeye cesaret bile edemeyeceği türden bilgisayarların- çağı yüzyılımızda başladı. 1930 ve 1940'larda ilk "dev elektronik beyinler" tasarlandı, yapıldı; ve bunların aracılığıyla, daha önce birbiriyle hiç ilişkisi olmamış üç alan, aksiyomatik akıl yürütme, mekanik hesap çalışmaları ve zekâ psikolojisi, bir noktada buluştu.

Aynı yıllar, bilgisayar teorisinin koşar adım gelişmesine de sahne oldu. Bu teoriyse üst-matematikçilerle çok yakından ilişkiliydi. Nitekim, bilgisayar kuramında Gödel'in Teore-



1898'de Hollanda'da Leeuwarden'de doğan Maurits Cornelis Escher, 20. yüzyılın, belki de en "entelektüel" görsel sanatçılarından biridir. Özellikle taşbaskıları ile ünlü Escher'in neredeyse her yapıtı, bizi kendi algılama süreçlerimiz üzerinde düşünmeye, algılarımızın gerçekliği yansıtma yeteneğini sorgulamaya çağırır. Ama bunu söylemek bile Escher'in çok sevdiği ve bu yazının ana konusunu oluşturan paradokslardan birinin içine girmek oluyor. Kendi üzerimizde düşündüğümüzde, düşünen kim oluyor, düşünülen kim? Escher'in neredeyse her yapıtı böylesi bir recursion paradoksu içerir. Kendi üzerine dönmeye çalışan her şey, sonunda hem kendisi hem başkası olmak zorunda kalır. Üst kata çıkan bir merdivenin hareketini üst kattan alt kata inerek tekraramak mümkün değildir - merdiven kendi başlangıç noktasına hem geri dönebilir hem dönemez. Kendi üzerine kapanmaya çalışan bir düğüm ancak imkânsız bir mekânda kendisine kavuşabilir.

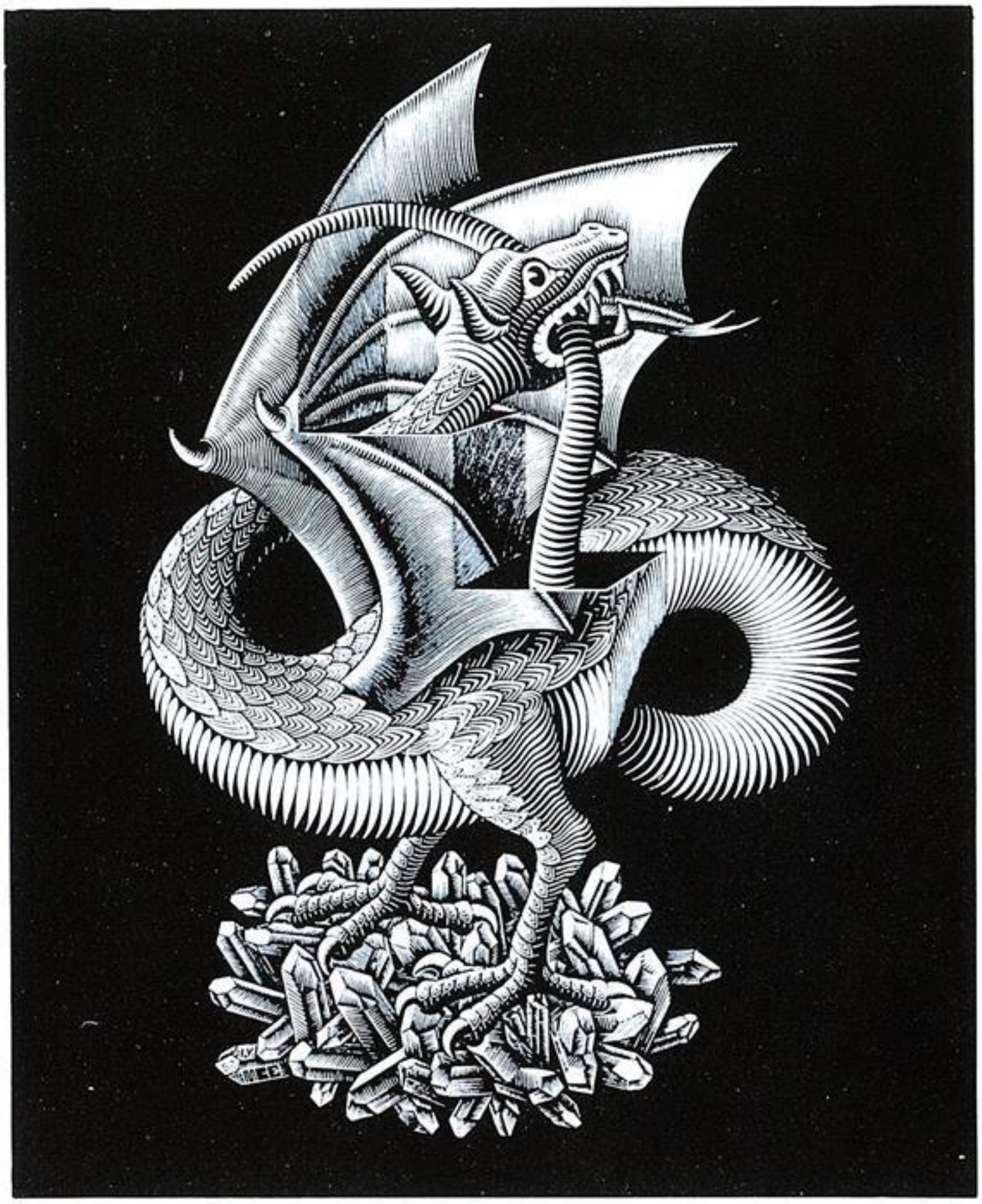
mi'nin Alan Turing tarafından geliştirilmiş olan ve tasavvur edebileceğimiz en güçlü bilgisayarda bile kaçınılamaz "delikler" olduğunu ortaya koyan bir benzeri bulunmaktadır. İşin komiği, tam da bir yandan bu acaip sınırlar keşfedilirken, öte yandan da yaratıcılarının kehanet kudretini gitgide geride bırakacak güçte gerçek bilgisayarlar üretilmekteydi. Vaktiyle, bundan beşyüz yıl sonra tekrar dünyaya gelip yeni çağda rehber eşliğinde üç günlük bir bilim gezisi yapabilmek uğruna ömrünün geri kalan bölümünden seve seve vazgeçeceğini söylemiş olan Babbage, ölümünden yalnızca bir asır sonra gelinen noktayı görebilse, bu yeni makineler ve onların beklenmedik sınırları karşısında herhalde heyecandan dili tutulurdu!

1950'li yıllara gelindiğinde, mekanikleştirilmiş zekâ sanki insanoğlunun parmaklarının hemen ucunda gibiydi; ne ki, aşılan her engelin arkasından, gerçek bir düşünen makinenin fiilen yaratılmasını önleyen yeni bir engel çıkıyordu. Biz ona yaklaştıkça hedefin böyle uzaklaşmasının altında yatan sebep neydi?

Zeki davranışla zeki-olmayan davranış arasında tam bir sınır çizmeye imkân yok; hatta, muhtemelen kesin bir sınırdan söz etmek bile saçma. Fakat, hiç kuşkusuz, şu nitelikler olmadıkça zekâ da olmaz:

- durumlara çok esnek bir şekilde tepki göstermek;
- elverişli koşullardan yararlanabilmek;
- muğlak veya çelişkili mesajlardan bir anlam çıkarabilmek;
- bir durumun değişik unsurlarının birbirlerine oranla önemlerini tespit edebilmek;
- aralarındaki farklara karşın, durumlar arasındaki benzerlikleri görmek;
- eski kavramları yeni şekillerde bir araya getirerek yeni kavram sentezleri oluşturabilmek;
- yeni fikirler üretebilmek.

Görünürde, burada bir paradoks var. Bilgisayarlar doğaları gereği dünyanın en esneklikten uzak, isteksiz ve kuralcı yaratıklarıdır. Bütün hızlarına karşın her biri bir bilinçsizlik anıtıdır. O halde, zeki davranış nasıl programlanabilir? Berbat bir terimlerde çelişme vakası değil mi bu?.. Bunun çelişki falan olmadığı, yapay zekâ programının başlıca tezlerinden birini oluşturuyor. Bu yazının en önemli hedeflerinden biri de, her

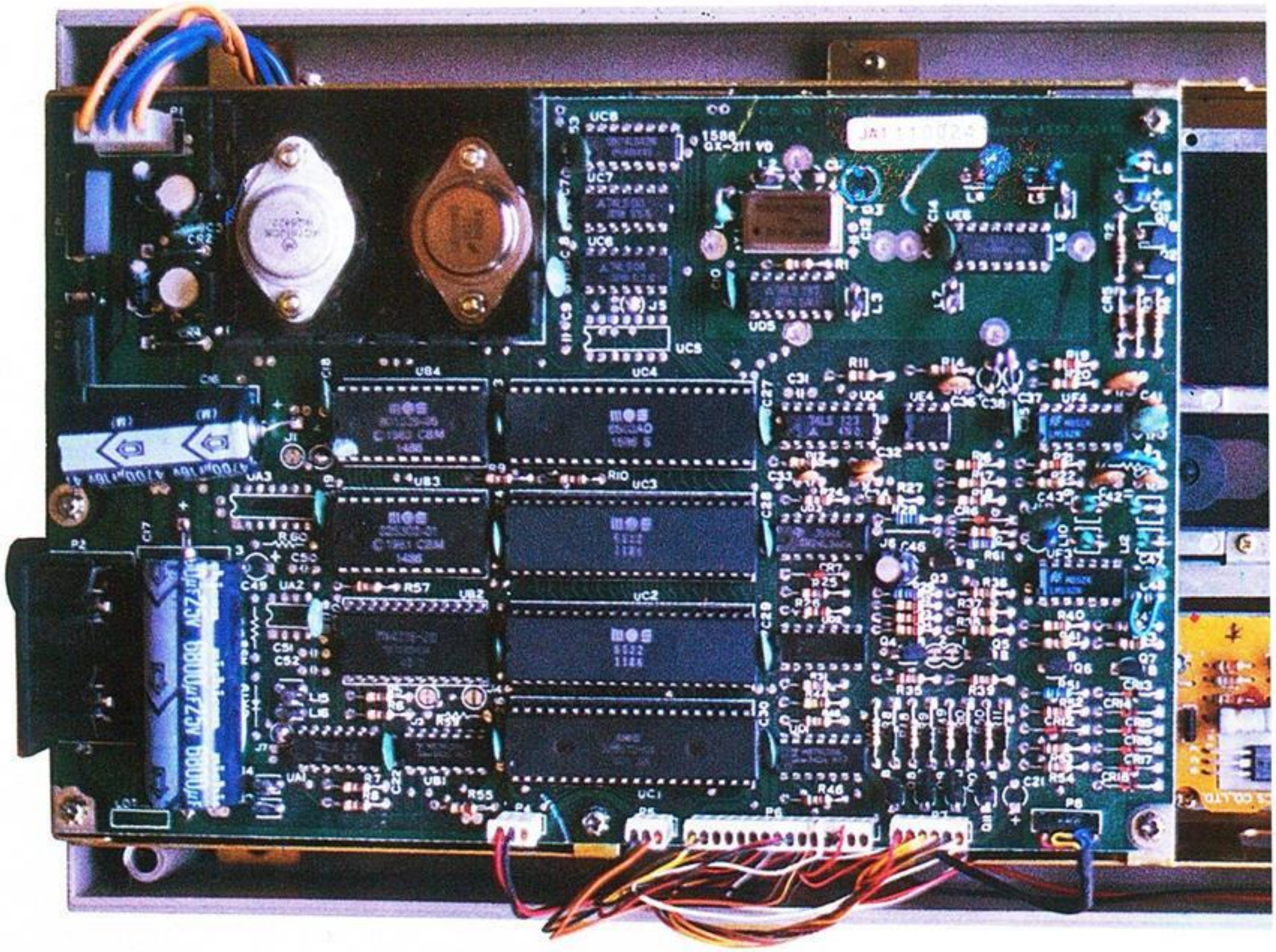


okuyucunun görünürdeki bu çelişkiyi karşısına alıp tadına varmasını, evirip çevirmesini, parçalarına ayırmasını, onun iyice içine girmesini sağlamak; ki okuyucu formelle informel, canlıyla cansız, esnekle esnek olmayan arasındaki sözde kapanmaz uçurum üzerine yepyeni bakış açılarına ulaşabilsin.

Yapay Zekâ (YZ) araştırmaları da işte bundan ibaret. YZ çalışmalarının tuhaf tarafıysa, insanların upuzun kuralları katı bir formel sistem içinde birbiri arkasına dizerek, esneklikten uzak makinelere esnekliği öğretmeye uğraşmaları...

Peki ama, zeki davranış deyince kasdettiğimiz her şeyi ne tür "kurallar" kapsayabilir ki? Hiç kuşkusuz, bütün o değişik düzeylerin her biri için ayrı kurallar gerekiyor. Bir kere, bir sürü "düpedüz" kural olmalı. Sonra, "düpedüz" kurallarda değişiklik yapmak için "üst-kurallar", "üst-kurallar"ı değiştirmek için de "üst-üst-kurallar" olmalı, vb. Zekânın esnekliğini sağlayan, farklı kuralların ve kural düzeylerinin inanılmaz çokluğudur. Bu kadar farklı düzeylerde bunca çok kuralın gerekli olmasının sebebine gelince, yaşamda her canlı birbirinden tamamen farklı milyonlarca durumla karşılaşır. Kimi durumlar stereotipik tepkiler doğurur; bunlar için "düpedüz" kurallar yeterlidir.

Kimi durumlar stereotipik durumların bileşiminden oluşmuştur; bu yüzden, "düpedüz" kurallardan hangisinin uygulanacağına karar vermeyi sağlayacak kurallar gerekir. Kimi durumlarsa sınıflanamaz; bu yüzden, yeni kurallar icat etmek için başvurulabilecek kurallar gereklidir, vb., vb. Hiç kuşkusuz, zekânın özünde doğrudan ya da dolaylı olarak kendi kendisini değiştirebilen kurallar içeren tuhaf recursion'lar yatıyor. Ve bazen zihnimizin karmaşıklığı öylesine başdöndürücü görünüyor ki, insana zekâyı anlamanın hiç yolu yokmuş, yukarıda tanımlanan çok-düzeyli anlamda bile olsa herhangi türden bir kuralın bir yaratığın davranışlarını belirleyebileceğini düşünmek yanlışmış gibi geliyor.



1541'i Anlamak

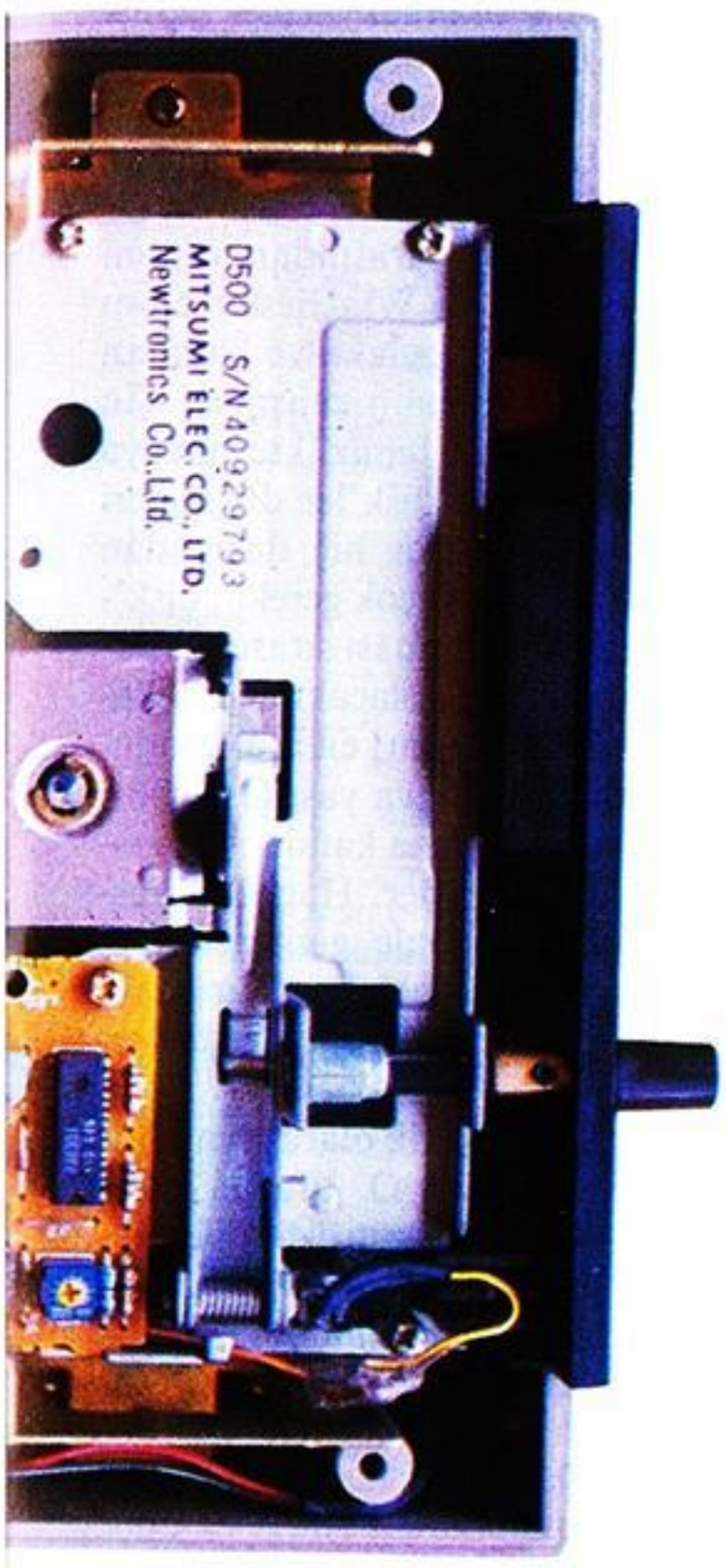
TUNA ERTEMALP

Yeni yazı dizimize hoşgeldiniz. Konumuz, COMMODORE 64'e bağlanan disket ünitesi 1541. Gerçi hepinizin 1541'i olmadığını biliyoruz. Ancak böyle kullanışlı bir alete sahip olan okurlarımızı yalnız bırakamazdık. Hem daha disket kullanımına geçmemiş COMMODORE 64 kullanıcıları, belki bu dizimizden sonra fikirlerini değiştirirler, kim bilir?

Daha önceki sayılarımızda, yeri geldikçe disket ve disket sürücüsü kullanımı hakkında çeşitli programlar ve yazılar sunmuştuk. Eğer onları dikkatlice okumuş ve incelemiş okuyucularımızdansanız, dizimizi izlemek size hiç de zor gelmeyecektir. Gene de herşeyi adım adım ve en basit seviyeden en üst seviyeye doğru çıkararak anlatacağız.

Dizimizin genel bir çerçevesini çizmek gerekirse... İlk bölümde, yani şimdi, kayıt araçlarının geçmişini ve bugününü (biraz da geleceğini), türlerini, yapılarını, farklılıklarını göreceğiz. Bu bize, bir disket ünitesinin ne olduğunu ve nasıl çalıştığını ileride daha rahat anlamak açısından sağlam bir temel oluşturacak. Sonraki bölümleri tamamen VC 1541'e ayıracağız. Disket sürücünüzü kutusundan çıkarmanızdan başlayıp basit kullanımlarıyla devam edeceğiz, LOAD ve SAVE işlemleri gibi. Sonra işin daha incelikli kısımlarına geçip, diskete komut göndermenin yöntemlerini göreceğiz. Sonra sıra gelecek değişik dosya türlerini tanımlayıp kullanmaya. Böylece disket sürücümüzü BASIC yardımı ile nasıl sömürebileceğimizi(!) tamamen görmüş olacağız.

Ondan sonra sıra gelecek bilgilerin disket üzerinde nasıl kaydedildiğini görmeye. TRACK ve SECTOR düzenlerini inceleyeceğiz. Haa, sırası gelmişken, bu iki kavramın ne demek olduğunu, eğer bilmiyorsanız, birazdan göreceksiniz. Eveet, bu inceleme



de bitince, ilk yapacağımız iş bu düzeni BASIC yardımıyla nasıl emellerimize alet edebileceğimizi görmek. Bu arada yaptıklarımızın sonuçlarını daha rahat görmek için bir de "Disket Monitörü" kullanmaya başlayacağız. Disket yapısının ıcığını cıcığını çıkarınca da dizimiz bitecek. Ama ondan sonra da muhtemelen Ayhan Kalaylıoğlu'nun ileri programlama yöntemleri ve makina dili kullanımıyla disket üzerinde neler yapabileceğinize dair yazısı başlayacak. Herhalde çok heyecanlı olacak!..

Artık başlayalım... Bilgisayarların ilk çıkışından bu yana en büyük sorunlardan biri, bilgisayara verilmesi gereken ve bilgisayarın işleyip bize geri sunduğu bilgi yığınlarının nasıl saklanacağıdır. Bunun için ilk kullanılan şeyler kağıt ve kart oldu. Tıpkı bugünkü teleks makinaları gibi, şeritler veya kartlar üzerine delikler delerek bilgisayarla haberleşme yapılıyordu. Bu kartlar günümüzün bilgisayar terminallerinin ve programlarımızı sakladığımız kayıt araçlarının yerini tutuyordu. Ana işlem ünitesine programları göndermek için, program önce özel aletlerle delikli kartlara geçiriliyor ve bu kartlar bilgisayara bağlı bir okuyucu makinaya takılıyordu. Bilgisayar da programı, tıpkı bugün kaset veya disketten yaptığı gibi, kartlardan okuyordu. Ve gene tıpkı bugün kaseti veya disketi sakladığınız gibi, o günlerde de çekmeceler dolusu kart saklanıyordu. Tüm bu zahmetli günlerin üstünden sadece 20 yıl gibi kısa bir süre geçti.

Sonra günün birinde manyetik özelliklerin kullanılması akla geldi. Manyetik bir yüzeyin üzerine gene manyetik bir kafayla bilgiler yazılıp, yüzey üzerindeki parçacıkların manyetik yönünü algılayan bir kafayla da bilgi okunabilirdi. Bunun gerçekleşmesi bir devrim oldu. Artık daha az bir alan üzerine çok daha fazla bilgi saklanabiliyordu. Gerçi bu bilgi miktarı günümüz kayıt araçlarının yanında çok küçük kalıyordu ama gene de o dönem için çok büyük bir gelişmeydi.

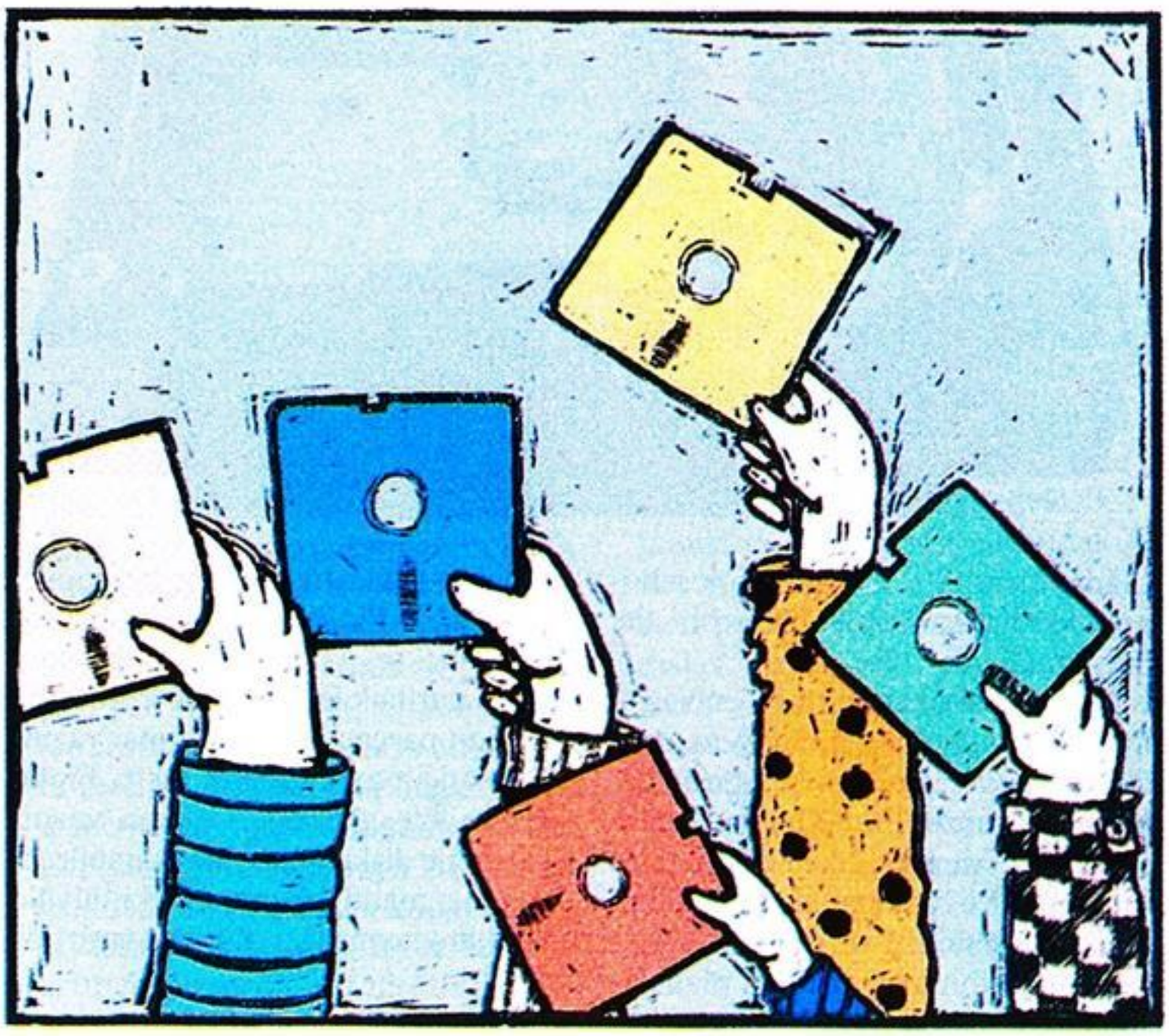
Manyetik araçların ilki bildiğimiz bantlardı. Ama o zamanlar müzik kasetleri değil de çok daha büyük ve kalın makara bantlar kullanılıyordu. Günümüzde de bu bantların daha hızlı ve gelişmiş olanları, bilgisayar sistemlerinin vazgeçilmez parçalarıdır. Bunun nedeni, RAM veya disket üzerinde aynı miktardaki bilgiyi saklamanın bant üzerinde saklamaktan

Manyetik kayıt cihazlarındaki gelişme bilgisayarlara daha ucuz ve kolay bilgi depolanmasını ve bilgiye daha kolay ulaşılmasını sağladı.

çok çok daha pahalıya çıkması.

Bant sistemlerinin başlıca karakteristiği okumanın ve yazmanın aynı sırada yapılması gerekliliği. Bir bilgiyi banda hangi sırada kaydetmişseniz o sırada da okursunuz. Bir dosyanın okunmasını ortada bırakıp başka bir dosyayı okumaya başlamak, sonra eski dosyaya geri dönmek mümkün değildir. Bu tür kayıt biçimine SEQUENTIAL (sıralı) depolama adı verilmektedir.

Bu kayıt biçimini değiştiren araçların ilki bir silindirdi (drum). Çapı 40-80 cm. arasında değişen bir silindirin üstü tamamen manyetik bir madde ile kaplıydı. Silindir kendi ekseni etrafında büyük bir hızla dönüyordu. Silindirin üst yüzeyine çok ya-



kın bir uzaklıkta bir okuma/yazma kafası vardı. Silindir yatay dururken, okuma/yazma kafası sağa sola hareket edebiliyordu. Bu hareketin ve silindirin dönmesinin oluşturduğu bileşke hareket, tüm silindirik alanın kullanılmasına izin veriyordu. Bu da, alan üzerindeki herhangi bir noktaya direkt ulaşılmasını sağlıyordu. Bu tür bir ulaşım ise RANDOM adıyla anılır.

Ne yazık ki, bu araç çok pahalıya maloluyordu. Çünkü, mekanik aksam ile silindir birbirine bağımlıydı. Kaset veya bantta olduğu gibi aynı alete değişik silindirler takılıp çıkartılamıyordu. Üstelik hafıza gereksinimi de artmaktaydı. Yeni bir alet bulunmalıydı.

Bu kayıt aracından türetilen ve günümüzde de kullanılan bu yeni araçlar ise disk'lerdir. Bunları biraz daha ayrıntılı görelim. Yukarıda bahsettiğimiz silindiri dikey olarak tuttuğumuzu ve bir bıçakla yatay dilimlere böldüğümüzü varsayalım. Ortaya üstüste konmuş yuvarlak plakalar çıkacaktır. Şimdi bu plakaların (veya disk'lerin) her iki tarafını manyetik materyalle kaplayalım ve her iki plakanın arasına iki taraflı okuyup yazabilen bir kafa yerleştirelim. Tüm bu kafaları da ortak bir kola bağlayalım. Bu kolu da aletin motoruna. Ortaya çıkan şekli şematik olarak Figür:1.1'de görebilirsiniz.

Bu yöntemle hem kullanacağımız alanı büyüttük, hem de her kafanın sorumlu olduğu alanı küçülttüğümüz için, plakaların üzerindeki bilgilere erişim süresini kısalttık.

Dilerseniz, şimdi de okuma/yazma kafası ile plaka yüzeyi arasında olup bitenleri daha yakın plana alalım. Bu bölgeye yaklaştıkça ilginç bir sonuca varacağız. Bir Jumbo jet için bir dağ ne demekse, küçük bir toz parçası, hattâ parmak izi, daha da kötüsü bir duman parçacığı bile okuma/yazma kafası için aynı anlamı taşır. Figür: 1.2'ye bir gözatın. Okuma/yazma kafası ile disket yüzeyine konabilecek yabancı maddeler arasındaki büyüklük karşılaştırması oldukça şaşırtıcı, değil mi?

Bir uçak dağın üstünden geçip gidebilir, ama okuma/yazma kafası için bu söz konusu değil. Önce bu sorunun disk'lerde nasıl çözüldüğünü görelim. Daha sonra disket yapısına geçeceğiz.

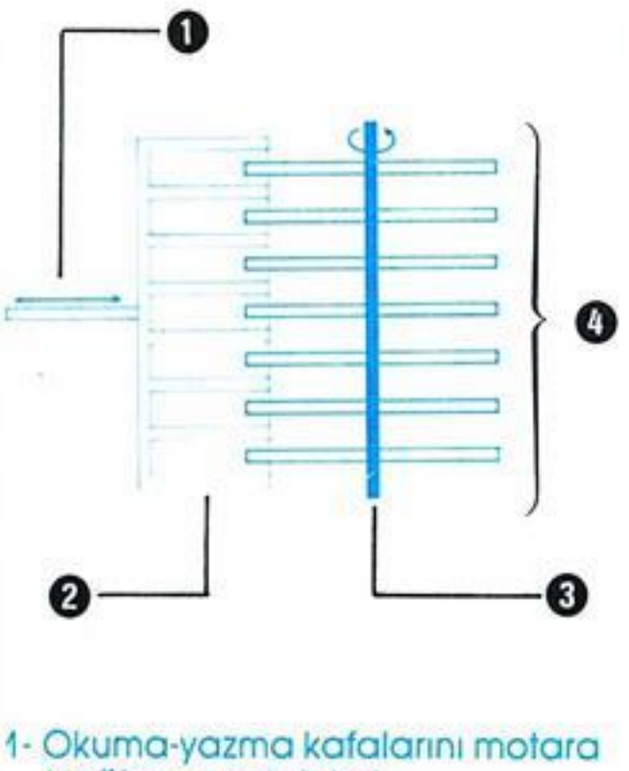
İlk önce IBM tarafından yapımı gerçekleştirilen ve Winchester adını alan, birden fazla plaka ve kafadan oluşan kayıt aracının çalışması için havaya, ama saf temizlikte havaya gerek vardır. Bu disk'ler düğmeleri kapatılıncaya kadar hiç durmadan (hızlı erişim için bu çok gerekli, disk'in durması ve kalkması sırasında boşuna zaman kaybı olacaktır) dakikada 3000 dönüş yaparken aralarındaki hava ince bir hava yastığı oluşturur ve okuma/yazma kafaları bu yastık üzerinde yüzerler. Hattâ bu yastığı daha etkili hale getirmek için Winchester'lerde uçak kanatlarındaki flap'ların benzeri düzenekler de yeralır.

Kaydın manyetik olarak yapıldığını söylemiştik. O ufacık okuma/yazma kafalarının üzerinde daha da küçük yarıklar vardır ve çok ince tellerden oluşan bir bobin yardımıyla bu yarıkta dünyanın manyetik



Bilgi yazılması ve okunması doğal olmayan yollarla yaratılan çok güçlü bir manyetik alan yardımıyla gerçekleştirilir.

alanının 1000 katı kadar güçte bir alan yaratılır. Ancak yarık çok küçük ve ince olduğu için bu alan çok uzağa gidemez. Bu nedenle disk yüzeyinin okuma/yazma kafasına çok yakın olması gerekir. Sonuçta bu alanın gücü değiştirilerek disk üzerindeki manyetik madde etkilenebilir ve bilgi disk yüzeyine yazılabilir. Okuma işlemi ise, disk'teki manyetik alanların okuma/yazma kafası üzerinde bir gerilim yaratması sonucu olur. Bu arada kafanın disket yüzeyinden uzaklığı ortalama olarak bir milimetrenin binde biri kadardır. Kafanın kazara disk yüzeyine değmesi



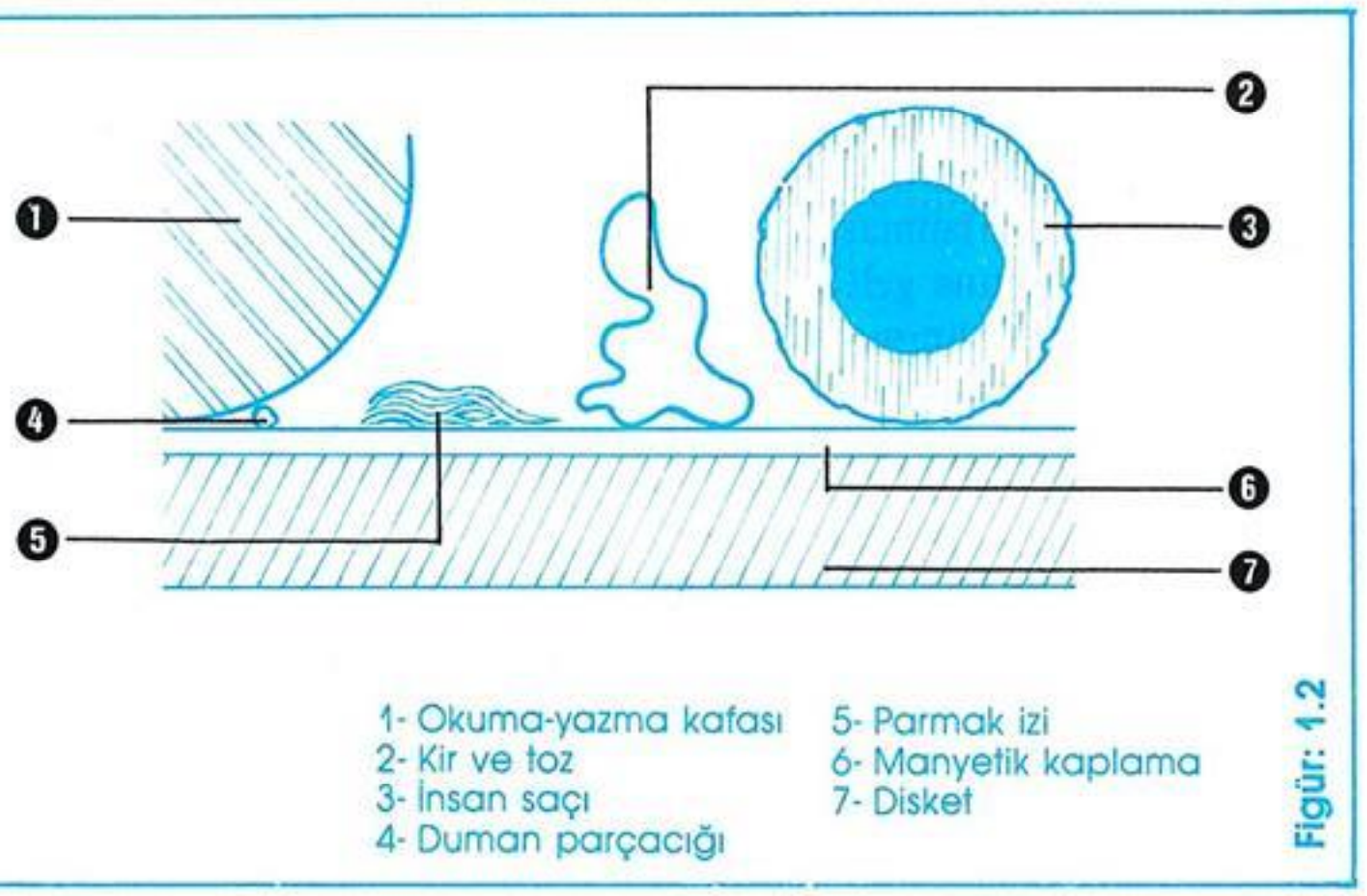
1- Okuma-yazma kafalarını motara bağlayan ortak kol
2- Her bir plaka için okuma-yazma kafası ve ona ait kolu
3- Ortak eksen
4- Manyetik plakalar

Figür: 1.1

disk'ler bir taneden fazladır ve okuma/yazma mekaniği ile bir aradadır. Aynı mekaniği başka disk'ler için kullanamazsınız. Disketlerde ise durum daha farklıdır. Bu kez elinizde bir tek disket vardır. Mekanik ise ayrı bir aletin içindedir. Dolayısıyla aynı aleti değişik disketler için kullanabilirsiniz. Bu, disket kullanımını yaygınlaştırmıştır. Üstelik disketler disk'lere göre daha az duyarlıdır. Daha rahat taşınabilirler. Ancak herşeyin bir fiyatı vardır. Bir Winchester 120 Megabyte (120 milyondan fazla karakter) bilgi kapasitelerine çıkabilirken, bir disketin günümüzde ulaştığı en yüksek bilgi kapasitesi 1.2 Mbyte'dır. Her ne kadar 10MB'lik disketlerden bahsediliyorsa da, daha bunların pratik kullanımı başlamamıştır.

Bu disketlerin kullanımı tek veya iki taraflı olabilir. Bu tamamen disyinde manyetik madde olduğu anlamına gelmez. Her disketin her iki yüzü de manyetik madde ile kaplanmıştır. Ancak SS olanlarda sadece bir tek yüzey test edilmiş ve satışa sunulmuştur. DS olanlar ise iki tarafı da kalite kontrolünden geçmiş olanlardır. Bu nedenle SS veya 1S disketleri de iki taraflı olarak kullanabilirsiniz. Ancak diğer tarafın her an sorun çıkarabileceğini unutmayın.

1D veya SD (Single Density) ise, disketin üzerindeki manyetik kaplamanın tek yoğunluklu olduğunu gösterir. 2D veya DD (Double Density) ile de, kaplamanın çift yoğunluklu olduğunu belirtir. Ev bilgisayarları için genellikle SD disketler kullanılır. DD disketler iş bilgisayarlarının kullanımına ayrılmıştır. Bir de HD (High Density) veya QD (Quad Density) adıyla geçen disketler vardır ki, on-



1- Okuma-yazma kafası
2- Kir ve toz
3- İnsan saçı
4- Duman parçacığı
5- Parmak izi
6- Manyetik kaplama
7- Disket

Figür: 1.2

veya bir toz parçacığını yüzey üzerinde sürüklemesi çok vahim sonuçlara yolaçacaktır.

Temizliğin önemi iyice ortaya çıktı. Bu temizliğin sadece disk yüzeyi ve onu saran hava için değil, Winchester'in yapıldığı tüm materyal için geçerli olması gerekiyor. Aracı oluşturan hiçbir parçanın zaman içinde parçacık, hattâ partiküller halinde bile artık yaratmaması gerekir. Burada en büyük sorunu mekanizmanın hareketliliğini sağlayan yağ oluşturur. Bu yağın disk yüzeyine ulaşmaması gerekir. Bunu uzay araçlarının motorlarında da kullanılan bir yöntemle çözmek mümkün olmuştur. Mıknatıslanabilir yağ kullanılarak ve motor aksamını da hafif mıknatıslayarak, yağın çok yüksek dönüş hızlarında bile motordan kopmaması ve durduğu zaman da yerçekimi ile aşağıya akmaması sağlanmıştır.

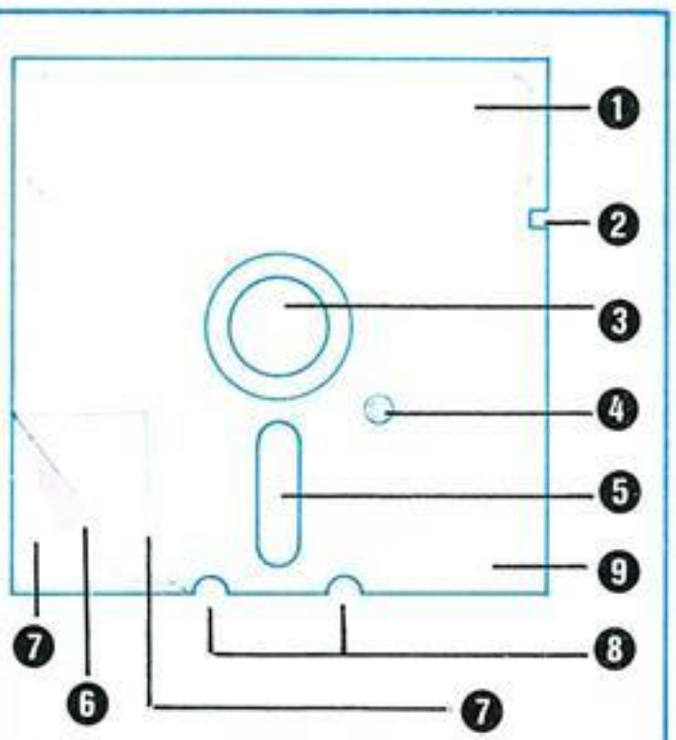
Bir diğer sorun da, yüksek dönüş hızı sonucu aletin içindeki hava hareketinin yaratacağı hava basıncının dışarıdaki hava basıncı ile dengelenmesidir. Bunun için dışarıdan içeriye hava girmesi gerekir. Bu da kirli, yani tehlikeli hava demektir. Bunu önlemek için, aracın tüm hava girişleri sadece saf hava moleküllerini geçirecek kadar küçük delikleri olan bir maddeyle kapatılmıştır.

Winchester'lerin daha da ayrıntısına girmeden esas konumuz disketlere dönelim... Winchester'lerde

ket sürücüsüne bağlıdır. Örneğin 1541'in sadece bir tek okuma/yazma kafası vardır. Bu nedenle aynı anda disketin sadece bir tek tarafını kullanabilir. Ancak, COMMODORE 128'in disket sürücüsü, 1571'de iki okuma/yazma kafası vardır. Bu nedenle önü ve arkası aynı anda kullanılabilmektedir.

Bu arada bir konuya daha değinelim. Disketlerin üzerinde 1S, SS, 2S, DS, 1D, SD, 2D, DD gibi kısaltmalar göreceksiniz. Bunların anlamlarını verelim. 1S veya SS (Single Sided), disketin tek taraflı kullanım için üretildiğini belirtir. 2S veya DS (Double Sided) ise çift taraflı kullanılabilecek disketleri belirler. Bir noktayı belirtmek istiyoruz. Bir disketin SS olması demek, onun sadece bir yüzelar 1.2 MB ve daha yüksek kayıt kapasitelerine ulaşmak için kullanılırlar. Gerek kalite, gerekse fiyat yönünden bize biraz büyük geliyorlar!

Şimdi de dilerseniz bir disketin yapısına gözatalım. Figür: 1.3'de de görebileceğiniz gibi disket değişik katmanlardan oluşur. En dışarıdaki disket muhafazasının iç yüzeyi bir tür özel post ile kaplanmıştır. Bu post, disket yüzeyindeki zamanla birikebilecek yabancı maddeleri toplamaya yarar. Böylece okuma/yazma kafası ile disket yüzeyi arası temiz kalır. Disketin asıl önemli kısmı da plak şeklindeki manyetik parçadır. Üzerindeki bilgilerin okunması için, bilginin olduğu bölgenin disket muhafazasında yer alan pencerenin altına gelmesi gerekir.



1- Etiket
2- Yazma koruması
3- Orta delik ve kuvvetlendirici halka
4- İndeks deliği
5- Okuma penceresi
6- Oksidle kaplı polyester disk
7- Post
8- Sabitleştirme girintileri
9- Disket muhafazası

Figür: 1.3

Figürde de gördüğünüz gibi bir de indeks deliği var. Bu delik, disketin dönüş hızının disket sürücüsü tarafından tam olarak ayarlanmasını sağlar. Tam bu deliğin altına gelecek şekilde disket sürücüsünün mekaniğinde yeralan bir fotosel, bu deliğin dakikada kaç kere açık olduğunu saptar ve olası hız değişimlerinin hemen düzeltilmesini sağlayacak sinyaller üretir. Ancak 1541'de bu delik hiçbir işe yaramamaktadır. Deliğin üstlendiği senkronizasyon işlemi, 1541'de disketin üzerine belli aralıklarla yazılan manyetik kodlar yardımıyla sağlanır. Bunu daha ilerki bölümlerde göreceğiz.

Disketin ortasındaki delik, onun mekanizmanın üstüne tam oturmasını ve dönüş mekanizmasına bağlanabilmesini sağlar. Bu bölgede mekanik sürtünme en üst düzeyde olduğu için, çoğunlukla sert bir kuvvetlendirici halka ile takviye edilmiştir. Eğer disketinizde bu halka yoksa, tehlike sinyalleri çalıyor demektir.

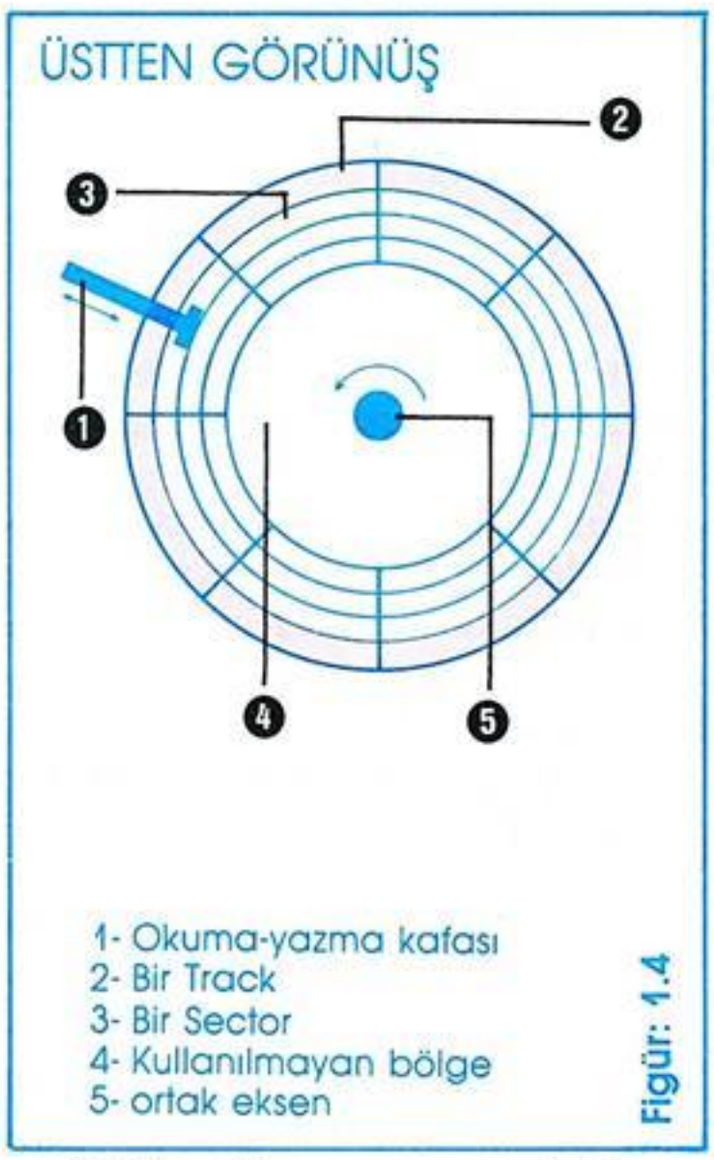
Sıra geldi disketin alt ve yan kenarlarında yeralan girintilere. Alt taraftaki iki küçük girinti, disketin mekanizma içinde yerinden kaymamasını sağlar. Yan taraftaki diğer girinti ise oldukça önemlidir. Bu girinti bir başka maddeyle kapatılmadığı sürece diskete bilgi yazılabilir. Ancak burası kapatıldığında, artık disketiniz yazılmaya karşı korunmuş olacaktır. Bunu da disket sürücüsündeki bir başka fotosel algılıyor. Tek taraflı disketlerin arka yüzünü de kullanmak istediğimizde, diğer kenara da bir delik açmamızın nedeni bu.

Disketinizi korumak için onu direkt güneş ışınlarından, ultraviyole ışınlarından, aşırı kurak veya nemli ortamlardan, aşırı sıcak veya aşırı soğuk çevrelerden, mıknatıslardan ve meraklı ellerden uzak tutmanız gerektiğini de belirterek disketin fiziksel özellikleriyle ilgili bilgilerimizi tamamlayalım. Sıra geldi, bilgilerin disket üzerindeki mantıksal yerleşimine...

Şimdi, anlatacaklarımızı Figür:1.4'ün yardımıyla izlemeye çalışın. Okuma/yazma kafasının sabit durduğunu ve sadece disketin döndüğünü farzedelim. Bu durumda disketin yüzeyinin sadece belli bir kısmı kafanın altından geçecektir. Bu da bir çemberdir. Şimdi kafayı biraz dışarı çekelim. Yeni ve daha büyük bir çemberi okumaya başladık. Böylece devam ederek, tüm disket yüzeyinin içiçe çemberlerden oluştuğunu söyleyebiliriz. Bu çemberlerin her birinin genişliği, okuma/yazma kafasının bir kerede kontrol edebildiği genişliktir. İşte bu çemberlerin her birine TRACK adı verilir. Track'ların uzunluğu dışarıdan içeriye doğru kısalır. Bu son cümle aklınızda kalsın, lazım olacak...

Disketlere ve tabiî disk'lere de, bilgiler bloklar halinde yazılır. Bir blok genellikle 256 veya 512 byte uzunluğundadır. Bu kadar az bir bilgi için bir track'in tamamının kullanılması affedilmez bir savurganlık olurdu. Bu nedenle her bir track, kendi içinde daha küçük parçalara bölünmüştür. Bunlara da SECTOR adı verilir. Böylece disket üzerindeki herhangi bir blok, track ve sector numaraları verilerek adreslenebilir. Şimdi gelelim bir önceki paragrafın son cümlesinin önemine... Figür: 1.4'de de gördüğünüz gibi track'larla beraber sector'lerin de boyu içeriye gittikçe kısalmakta. Dolayısıyla, her sector'deki bilgi miktarının aynı uzunlukta olduğunu düşünürsek, bilgi kayıt yoğunluğu içeriye girdikçe artacaktır; aynı miktar bilgi daha dar bir alana yazılacaktır. Bunu sağlamak için de, daha içerideki track'lara geldikçe disk veya disket sürücüsü hızını yavaşlatacaktır.

Bir başka yöntem ise, ki bu 1541'in izlediği yöntemdir, track'ları ve sector'leri içeriye doğru kısaltmak yerine, track başına düşen sector sayısını değişken kılıp, sector uzunluklarını sabit tutmaktır. Bu durumda, sector'ler Figür: 1.4'deki gibi pasta dilimlerine benzer bir düzen taşımazlar. En dıştaki track'da 30 sector varken en içtekinde sadece 10 sector olabilir.

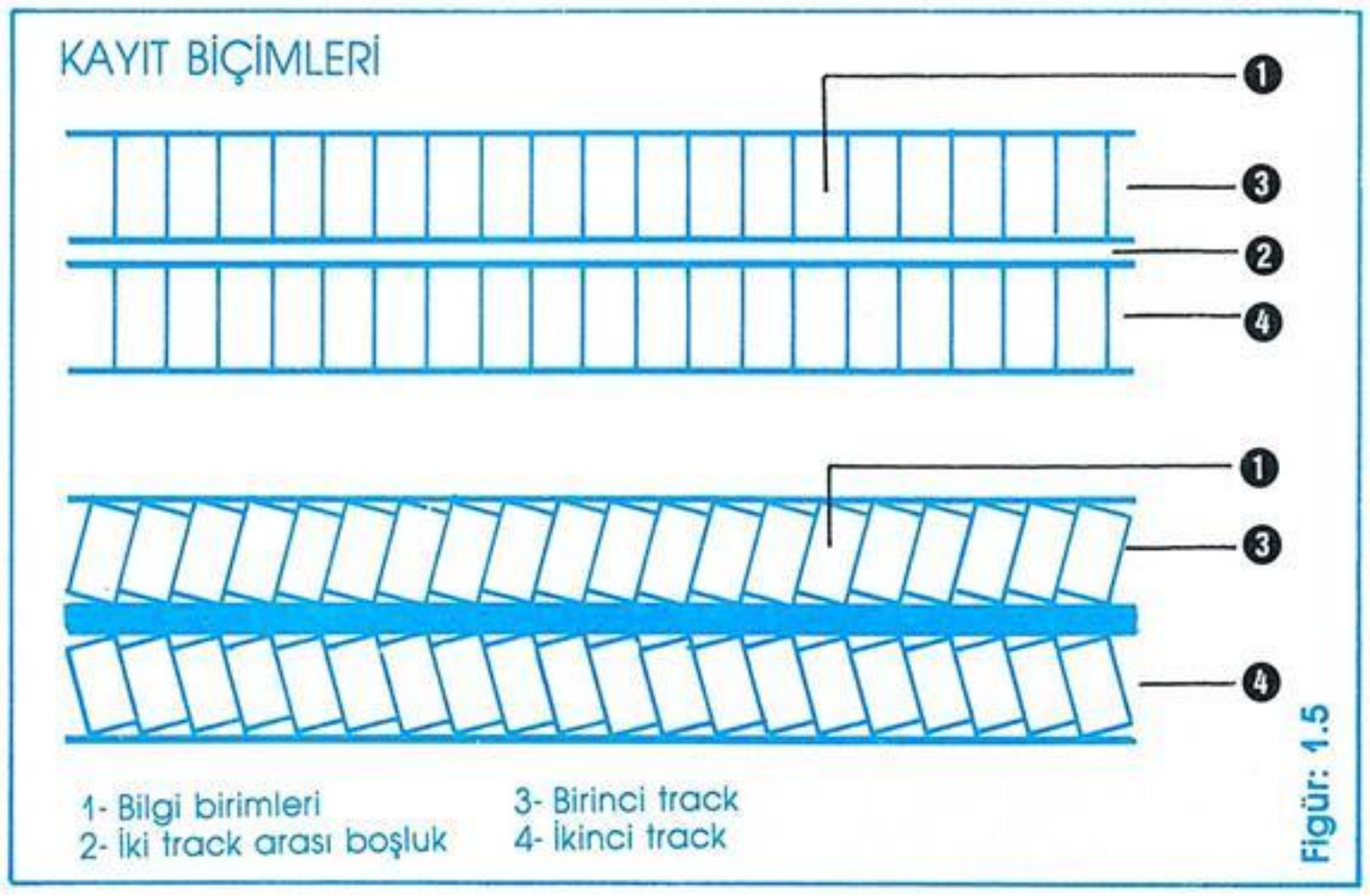


1- Okuma-yazma kafası
2- Bir Track
3- Bir Sector
4- Kullanılmayan bölge
5- ortak eksen

Figür: 1.4

En içteki track ile eksen arasındaki belli genişlikteki bir alan boş bırakılmıştır. Burası artık bilgi yazmaya değmeyecek kadar kısa veya bilgi yazılması riskli bölgelerdir. Bu bölgeler, Winchester'lerde çok daha geniştir. Yüksek kayıt kapasitelerinin kaçınılmaz sonucu olan yüksek kayıt yoğunluğu, bunun nedenidir. Daha iç track'lara yazmak, zaten normalde yüksek olan yazım yoğunluğunu daha da arttıracak ve bilgi saklanmasının riskini arttıracaktır. Ancak daha az bilgi saklayan disketlerde bu tehlike çok küçüktür ve disket yüzeyinin

Manyetik kayıt diskleri fiziksel yapıları gereği ve içindeki bilgiler nedeniyle iyi korunmalı ve saklanmalıdırlar.



Figür: 1.5

daha geniş bir kısmı kayıt için kullanılabilir. İçeride kalan bu alan, Winchester'lerde bir tür park sahası olarak da kullanılır. Winchester'i elektrikten çektiğinizde, kafa kendiliğinden bu kullanılmayan alanın üzerine gelir ve orada kalır. Bunun nedeni, disk sistemlerinde kafa düşmesi (Headcrash) adı ile anılan ve karşılaşan kişi açısından oldukça acıklı sonuçları olan bir olayı önlemektir. Bir disk'in kafası, sert darbeler sonucu manyetik yüzeyin üzerine düşebilir. Bunun sonucu aşikâr... Hiç olmazsa kullanım dışı zamanlarda bu tehlikeyi ortadan kaldırmak için kafa tehlikesiz bölgeler üzerine çekilir.

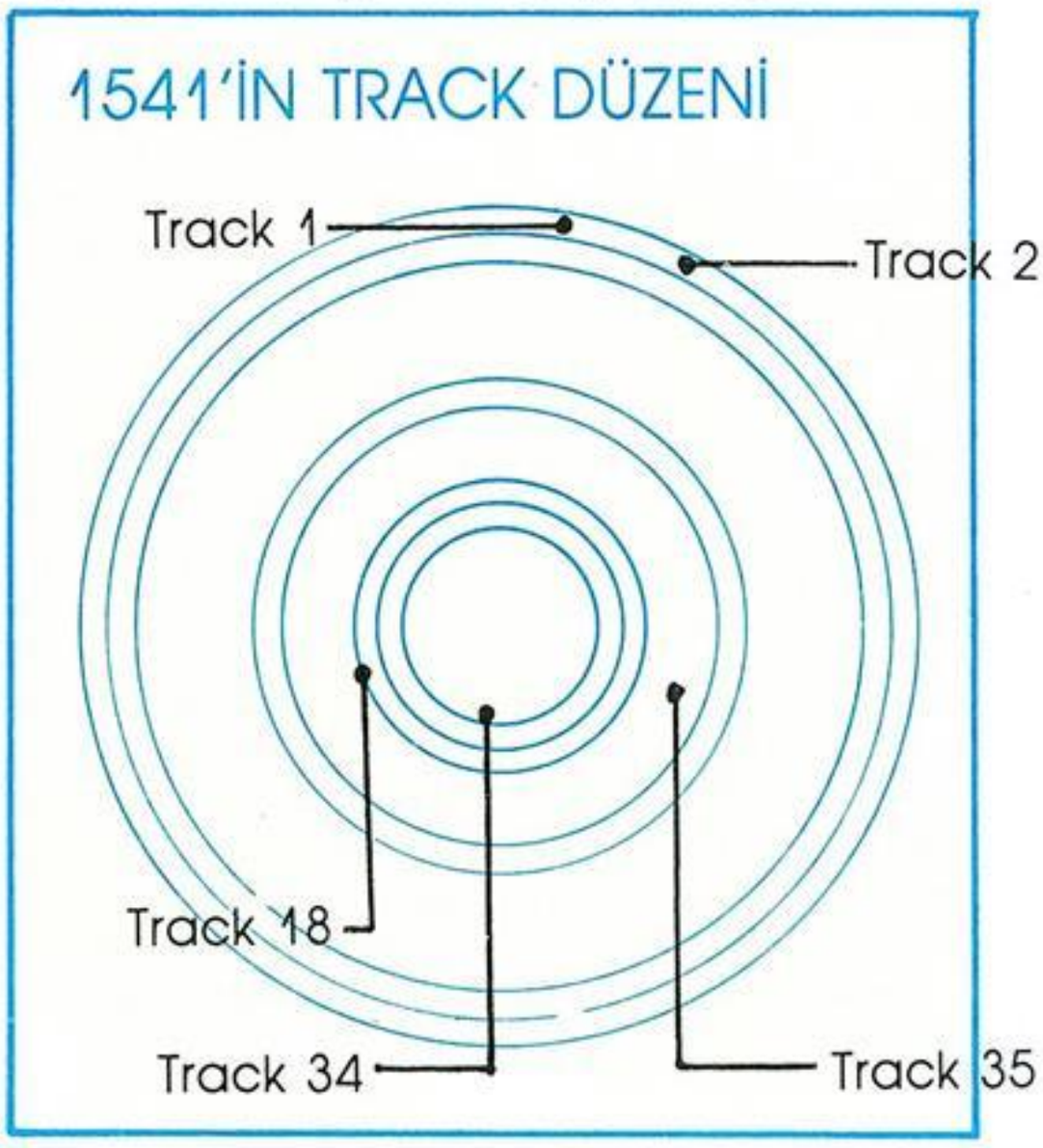
Şimdi de diskete iyice yakınlaşıp bilgi birimleri olan bit'lerin diskete nasıl yerleştirildiğine bakalım. Figür: 1.5, iki tür kayıt biçimini şematik olarak göstermektedir. En sık rastlanan birinci şeklidir. Her bir track, yanındaki track'dan ince bir ara bölge ile ayrılmıştır. Her disket yüzeyi için bir tek okuma/yazma kafası yeterlidir. Ancak aradaki bölge nedeni ile değerli kayıt alanının bir kısmı kaybolur.

İkinci şekil ise, bilgi paketlerinin biraz yatık bir düzende kaydedilmesine dayanır. Birbirine komşu iki track'ın bilgi birimlerinin kayıt edilme eğimi ters yönlerde olacağı için arada boş bir alan bırakmaya gerek kalmayacaktır. Ancak bu sefer de, iki tür eğimi okumak için, her disket yüzeyi için iki değişik okuma/yazma kafası gerekecektir. Eee, herşeyin bir fiyatı var!..

Dilerseniz, biraz da geleceğe uzanalım... Geleceğin kokusu daha bugünden ortaya çıkıyor. Disk, harddisk ve Winchester kullanımı gittikçe modernleşiyor. Hem boyutları küçülüyor, hem kayıt kapasiteleri artıyor, hem de gittikçe hızlanıyorlar. Bizim bugün kullandığımız disketler ise gittikçe tarihe karışmaya başlıyor. En başta 8 inch'lik çapları olan disketler sonra 5 1/4 inch'e dönüşmüştü. Ancak yeni çıkan aletlerde 3.5 inch'lik disketlerin kullanılma oranı gittikçe artıyor. Örneğin Commodore Amiga, Macintosh, IBM'in yeni modelleri gibi. Bu disketlerin en büyük özelliği, oldukça sert bir muhafaza içinde adeta kale gibi korunmaları. Ancak disk sürücüsünün içine soktuğunuzda bir kapak açılıyor ve disket ile kafa arasında bir engel kalmıyor. Çok da küçük oldukları için, kolayca taşınabiliyorlar. En basitlerinin kapasitesi bile bugün kullandığımız disketlerin 2-3 katı kadar. Darbelerden, nemden, ısıdan etkilenmiyorlar.

Bunların dışında, CD-ROM adı altında yeni kayıt birimleri de piyasaya sunuldu. Bunların kapasitesi yüzlerce Megabyte düzeyinde. Genellikle iş bilgisayarlarına bağlanabiliyorlar. Büyüklükleri bir plak kadar. Ancak üstlerine 10 ciltlik bir ansiklopedideki tüm bilgileri kaydetmek mümkün. Daha şimdiden Amerika Birleşik Devletleri başta olmak üzere, dünyanın ileri ülkelerinde ansiklopedik yayınlar kâğıdın yanısıra CD-ROM üzerinde de çıkıyor. Bu plakların özelliği, lazer ışını tarafından okunmaları. Üstlerinde manyetik hiçbir kayıt yok. Yeni çıkan CD plakları gibi çalışıyorlar. Üstlerinde ancak lazerin görebileceği değişik uzunlukta delikler var. Manyetik olmadıkları için tozdan, mıknatısdan, ısıdan, ışınımdan etkilenmiyorlar. Lazerle okunduğu için aşınma da yok. Hattâ suyun altına bile sokabilirsiniz. Tek kötülüğü, bir kere üzerine yazıldıktan sonra silinememesi ve değiştirilememesi.

Bundan sonra neler olabilir?.. Gelişmiş laboratuvarlarda holografi yöntemlerini veya biyolojik oluşumları temel alan, bir ölçüde beyni taklit etmeye çalışan, günümüzde hayal etmesi bile zor kapasite ve hızlarda çalışabilecek kayıt ürünleri ve yöntemleri tasarlanmaktadır. Sonuçlarını görmemiz hiç de uzun sürmeyebilir!..





Memory Nedir?

SALİH EREN

Türkçe karşılığı bellek ya da hafıza olarak çevrilen memory kelimesi, bilgisayarda bilgilerin kullanılabilecek uygun formlarda saklandığı birimler anlamına gelir. Bilgisayarda kullanılan bellekleri ana bellek (prime ya da main memory) ve depo bellek (storage memory) olmak üzere ikiye ayırabiliriz. Örneğin R/WM (Read Write Memory) ve ROM (Read Only Memory) ana bellek grubuna girerler. Bu belleklerde bilgisayarın icra ettiği programlar ve her an kullanabileceği kısa rutinler mevcuttur. Küçük sistemlerde kullanılan bellekler genellikle birkaç kilobaytla sınırlıdır. Kaset teyp, manyetik teyp, floppy disk ve hard disk ise depo belleklere birer örnektir. Bu bellekler, bilgilerin veya programların devamlı saklanması ya da bir yerden başka bir yere nakli esnasında bilgisayar tarafından kullanılır. Kapasiteleri tip ve modellerine bağlı olarak değişir.

Bu yazıda ana bellek grubuna giren temel elemanlardan kısaca bahsedip depo bellek çeşitlerine ise ileriki sayılarda ayrıntılı olarak değineceğiz.

Yaz-oku bellekler daha popüler olarak rastgele erişimli bellek (RAM = Random Access Memory) olarak bilinirler. Bu bellek "uçucu" tipte bir bellektir. Nedeni ise besleme geriliminin kesildiği anda, içindeki bilgilerin kaybolması, başka bir deyişle belleğin sakladığı bilgileri unutmasıdır. Statik ve dinamik olmak üzere iki çeşit yaz-oku bellek tipi vardır.

Statik bellekler, çift kararlı ikili devrelerden (flip-flop) oluşur ve bit-

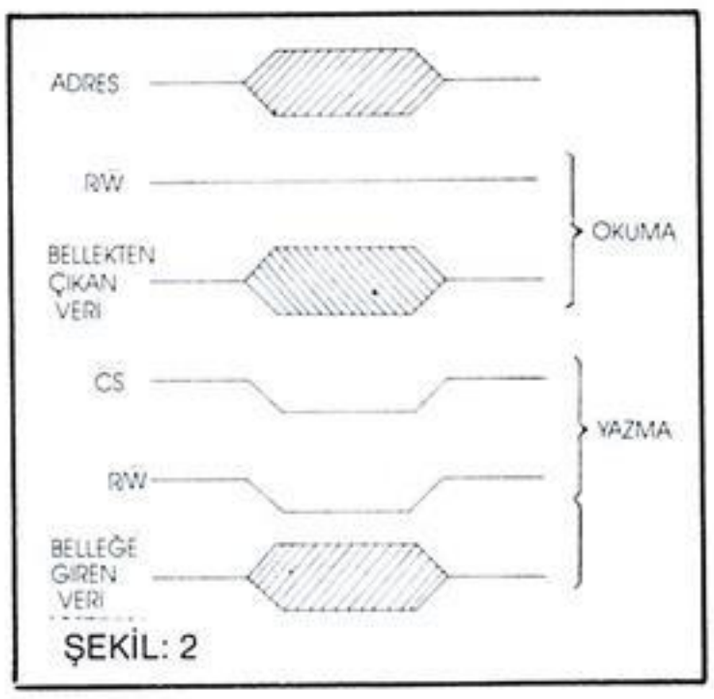
Bellekteki bilginin yazılması ve silinmesi tamamiyle elektrik akımıyla ve çok süratli olarak gerçekleşen bir işlemdir.

ler voltaj değeri olarak saklanır. Bu sayede statik bellekler besleme gerilimi kesilmediği müddetçe herhangi bir ek işleme gerek kalmadan içlerindeki bilgileri saklarlar.

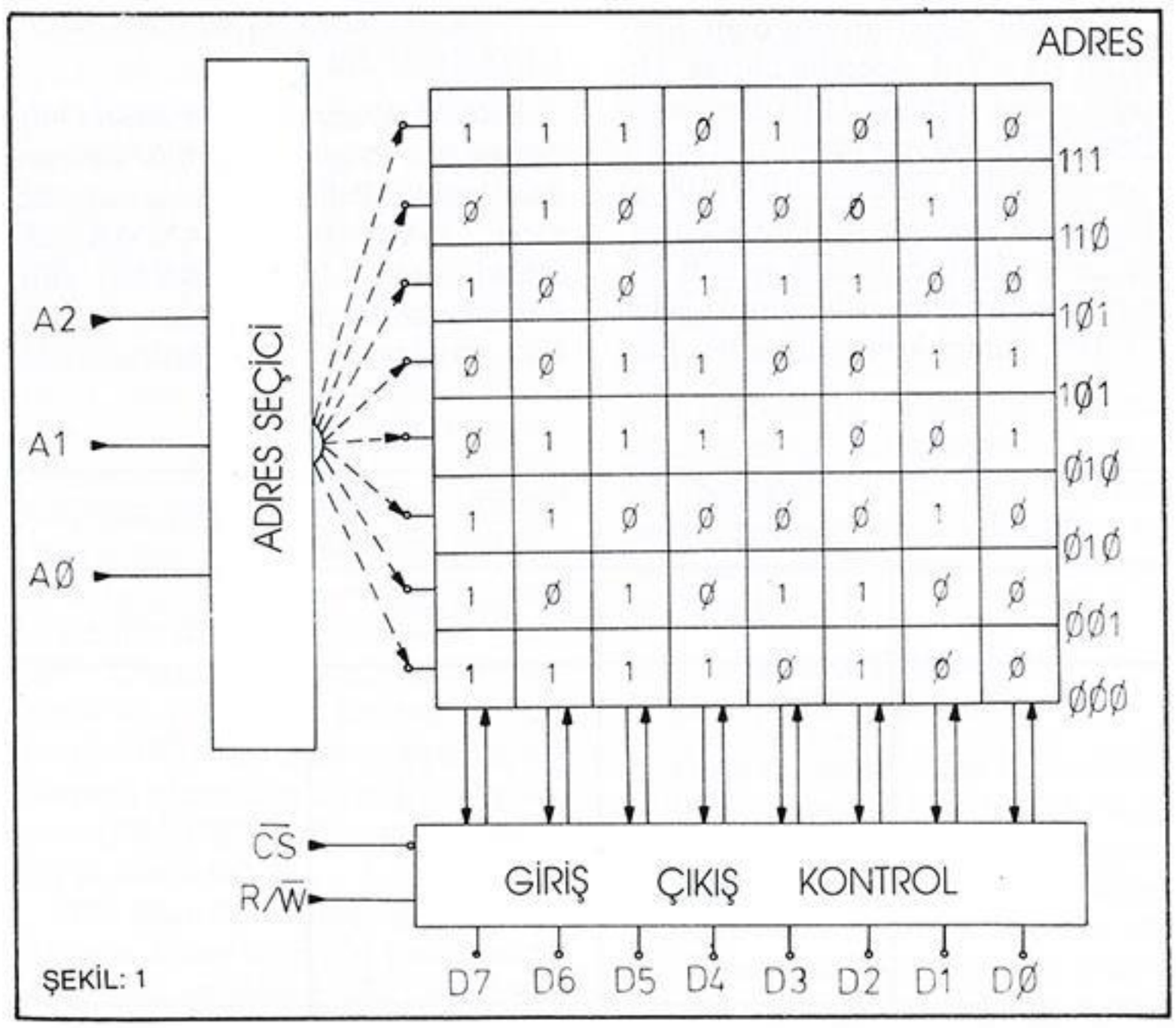
Dinamik bellekler ise MOS transistör kapılardan oluşurlar ve kondansatörün şarjı ilkesine göre çalışırlar.

Bir R/WM belleğin daha iyi anlaşılması için şekil 1'deki 8 baytlık (8*8 bitlik ya da başka bir deyişle 64 adet bellek hücreli) teorik bir bellek diagramı üzerinde bilginin nasıl yazıldığını ve okunduğunu açıklayalım. Teorik belleğimiz 8*8 bit matris düzenine sahip olduğundan tüm kayıt sıralarını seçmek için bize 3 adres hattı yeterlidir. Her hat 0 Volt ya da + 5 Volt olarak maksimum 2 farklı değer alabileceğinden 3 adres hattı toplam 8 farklı değer alabilir (2'nin 3'üncü kuvveti 8'e eşittir). Veri hatları ise seçilmiş herhangi bir bellek hücresinin taşıdığı değerin, çevre birimlere aktarılmasını ya da çevre birimlerden gelen bilginin seçilmiş bellek hücresine aktarılmasını sağlar. Minimum donanımlı bir sistemde bilgi alışverişinin bir ahenk içinde olması için belleğimizde iki tane de kontrol hattımız vardır. Bunlar R/W (Read-Write = oku-yaz) ve CS (Chip Select = çip seç) sinyalleridir. R/W sinyali yüksek iken (+5V) okuma işlemi, alçak iken (0V) yazma işlemi yapabiliriz. CS sinyali ise alçak olmadığı müddetçe bellek aktif değildir. Başka bir deyişle bellekle hiçbir bilgi alışverişi yapılamaz.

Şimdi bir bilgi yazma işleminin nasıl gerçekleştirildiğini adım adım görelim. Mikroişlemcimiz, Commodore 64'ün mikro işlemcisi olan 6510, yazacağımız bilgi de hex EA olsun. Hex EA değeri 6510'un komut kümesinde "hiçbir işlem yapma" anlamına gelen "NOP" (No Operation) komutunun kodudur. Bu kodu mikroişlemcimiz belleğimizin en son adresine (8. bayta) yazması için adres hattına ikili düzende 111 değerini yazması gerekir. Böylece A0, A1 ve A2 adres hatları +5 Volt değerini alır. Ardından D7..D0 veri hatlarına 1110 1010 (=hex EA) değeri konur ve R/W kontrol hattı 0 Volta çekilerek bilginin akış yönü tespit edilir. Bu adımların sonunda NOP komutu belleğin en son adresine yazılmış olur. Okuma işlemi ise biraz daha basittir. Mikroişlemcimiz okumak istediği adresin değerini A0, A1 ve A2 adreslerine yazması ve R/W kontrol hattını +5 Volta çekmesi bilginin D7..D0 veri hatlarında görünmesi için yeterlidir. Tabii tüm bu işlemlerin mümkün olabilmesi için belleğin aktif konumda yani CS sinyalinin alçak seviyede olması gereklidir. Şema 2'ye bakarak yazma ve okuma anında örnek belleğimizin bacaklarındaki sinyallerin konumlarını görebilirsiniz.



ŞEKİL: 2

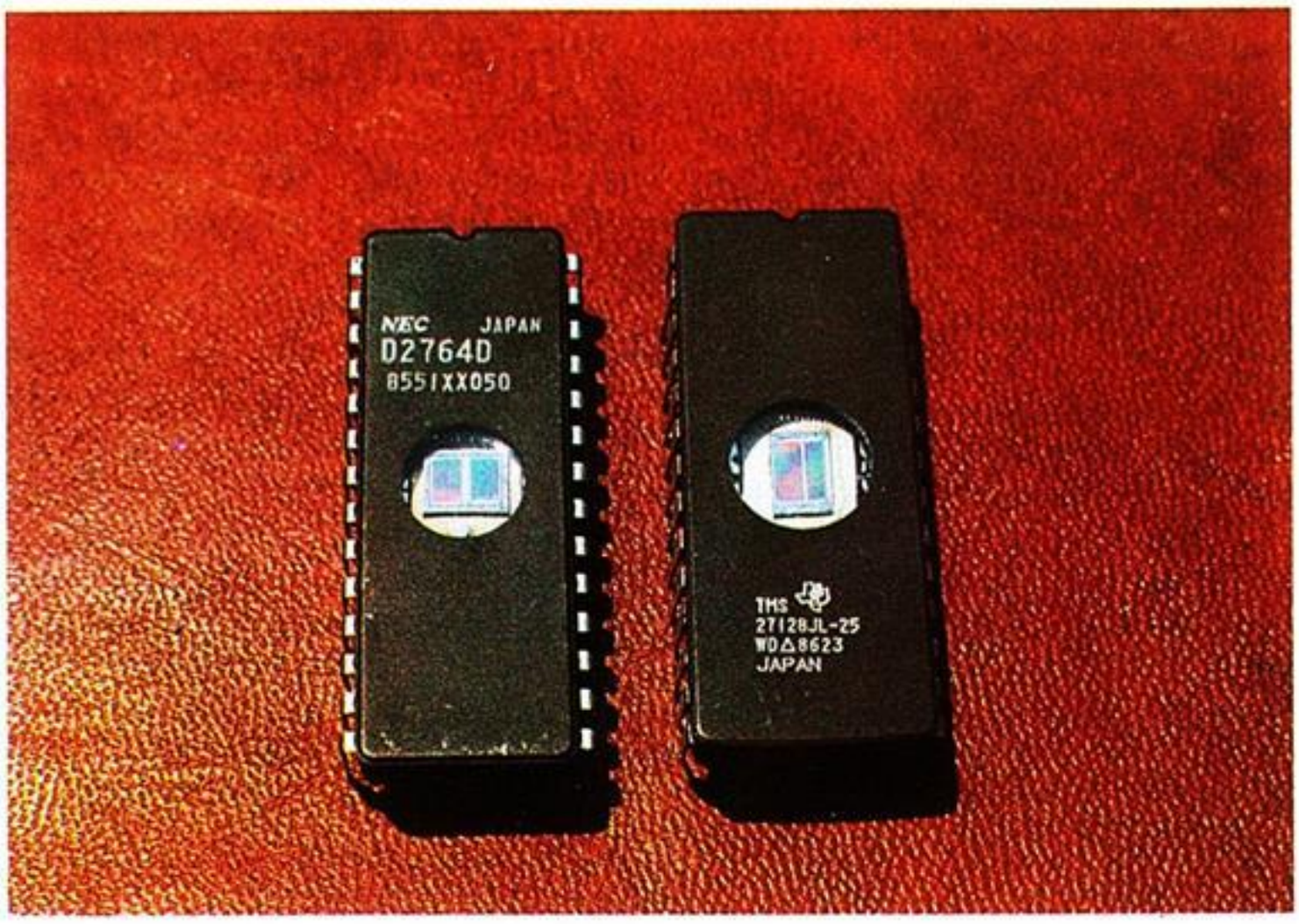


ŞEKİL: 1

DİNAMİK VE STATİK BELLEKLERİN KARŞILAŞTIRILMASI

Dinamik belleklerin, statik tiplere göre avantajı, bir bitlik bilgiyi saklamak için gereken elektronik eleman sayısının daha az olması ve bunun sonucu olarak da daha küçük alana (eşdeğer statik belleğe göre 1/5) daha fazla bilgi hücresi (memory cell) yerleştirilebilmesidir. (Şekil 3'e bakarak çeşitli belleklerde kullanılan devreleri ve eleman sayılarını karşılaştırabilirsiniz.)

Dinamik belleklerin dezavantajı ise, saklama kondansatöründeki yükün kaçak akım sebebiyle azalması ve bilginin kaybolmasıdır. Çünkü lojik bilgi, kondansatörün uçlarındaki gerilimin yüksek veya alçak olmasına göre tespit edilir (Şekil 3c). Bu problemi gidermek için birkaç milisaniyede bir her bilgi hücresi tek tek okunur ve aynı bilgi tekrar yazılır. Bu metoda tazeleme (refreshing) işlemi denir ve ek lojik kontrol devreleri gerektirir.



EPROM'lar ROM ailesi içinde en geniş kullanım alanına sahip ve bu nedenle en çok ilgi çeken bellek birimleridir.

Statik belleklerin bir özellikleri de, dinamik belleklere oranla 20-25 kat daha fazla akım çekmeleridir.

Sonuç olarak dinamik belleklerin ek kontrol devreleri gerektirmesine karşın, fiyatlarının 3-4 kat daha az olması, daha az akım çekmeleri, erişim sürelerinin (Access Time) biraz daha iyi olması sebebiyle büyük bellek gerektiren uygulamalarda dinamik bellekler, küçük bellek gerektiren uygulamalarda ise statik bellekler tercih edilir.

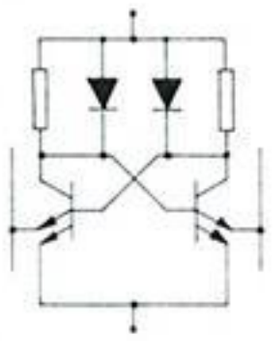
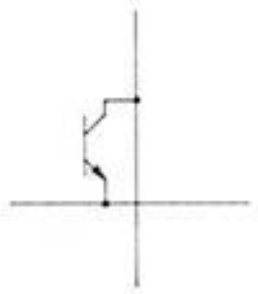
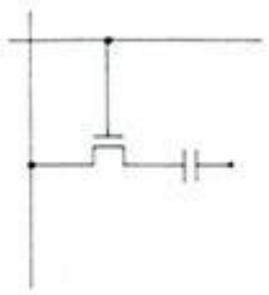
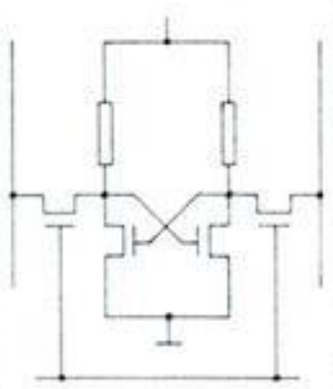
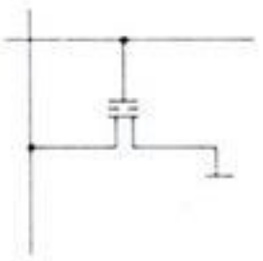
ROM (READ ONLY MEMORY = SADECE OKUNABİLİR BELLEK)

ROM'lar RAM'ların aksine besleme gerilimi kesildiğinde içlerindeki bilgileri unutmazlar, yani "uçucu olmayan" belleklerdir. Şekil 4'te 8 baytlık (8*8 bit ya da başka bir deyişle 64 bitlik) diyotlardan teşkil etmiş, matrix düzeninde örnek bir ROM belleğimiz görülmektedir. Yatay hatlar diyotlarla dikey hatlara bağlıdır ve yatay çizgiler adresleri, dikey çizgiler ise veri hatlarını temsil etmektedir. İkili düzende adres alanı 000'dan 111'e kadar olan yatay sekiz sıranın her biri 1 baytlık değer içerir. Bilgi, diyotların konumlarına bağlıdır. Diyot bağlı olan veri hattında değer, lojik 1 veya +5 Volt, boş olan hatta ise lojik 0 veya 0 Volttur. Bir adres seçildiğinde, seçilen yatay hat ve diyotların bağlı olduğu dikey hatlar +5 Volt değerini, boş olan dikey hatlar ise 0 Volt değerini alırlar. Örneğin adres seçicide 111 adresi seçildiğinde D7..D0 veri hatlarından okunan değer ikili düzende 01001100 ya da onaltılı düzende 4C Hex'e eşittir. Bu değer 6510 mikroişlemcinin komut kümesinde sıçra anlamına gelen "JMP" (jump) komutunun hex karşılığıdır. Mikroişlemcimiz bu komutu ROM'dan okuduğunda takip eden adrese sıçrayarak komutu icra eder.

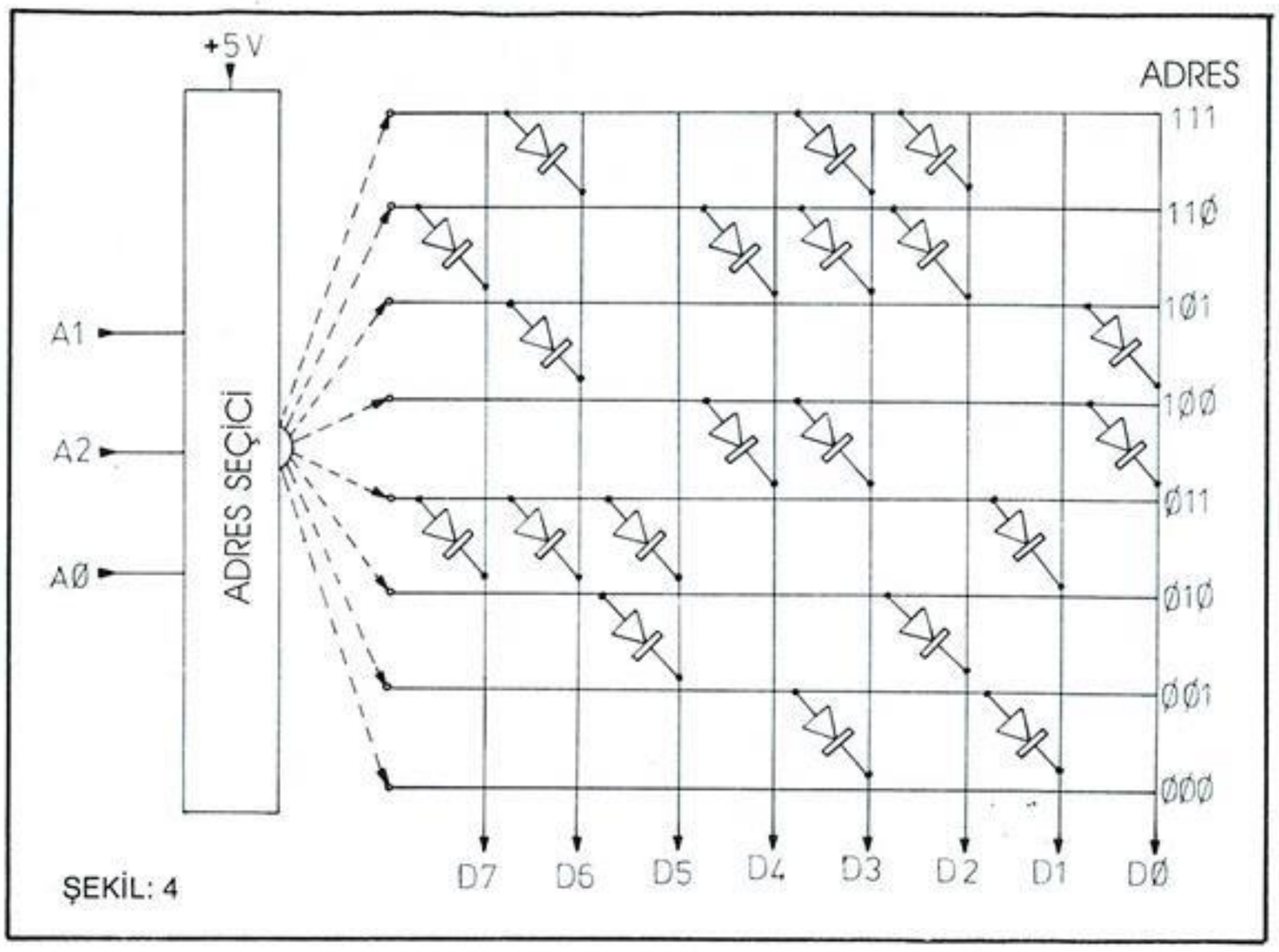
Elektronik malzeme üreticileri artık ROM'larda diyot yerine daha basit ve güncel olan MOSFET bellek hücreleri kullanmaktadırlar. Halen yaygın olarak kullanılan 4 çeşit ROM tipi vardır.

MASKED ROM (Maskelenmiş sadece okunabilir bellek): Bu ROM tipinde bilgiler, özel ve pahalı bir dizi işlemler sonucunda kalıcı olarak kaydedilir. İçindeki bilgiler üretim esnasında maskelendiğinden değiştirilemez. Bilgisayar üreticileri genellikle bu ROM tipini tercih ederler. Fakat pahalıya malolduğundan binleri aşan kullanımlarda ekonomiktir. Örneğin COMMODORE 64'deki basic, kernal ve karakter ROM'lar birer MASKED ROM'dur.

PROM (Programmable Read Only Memory = Programlanabilir-okunabilir bellek): Belleğin diğer bir ismi de ONE TIME PROGRAMMABLE READ ONLY MEMORY'dir (Bir kereye mahsus programlanabilir bellek). Ardından da anlaşıldığı gibi

Sınıfı	RAM	PROM	NMOS Dinamik RAM	NMOS, CMOS Statik RAM	NMOS PROM
Uygulama Alanları	Tampon bellek. Yüksek hızlı bilgisayarlar	Bilgisayar kontrollu devrelerde	Bilgisayarların ana belleklerinde	Bilgisayarların ana belleklerinde	Bilgisayar kontrollu devrelerde
Kullanılan Tipik Bellek Hücresi					

ŞEKİL: 3



ŞEKİL: 4

programlandıktan sonra geriye dönüş imkânı yoktur.

Şekil 6'da görüldüğü gibi bu bellek tipi nicrome veya polysilicon yollardan oluşmuştur ve matrix düzenindedir. Teller titanyum-tungsten karışımı sigorta işlevli tellerle birbirlerine bağlanmıştır. Bu belleği programlamak için özel bir PROM programlayıcı cihazına ihtiyaç vardır. PROM programlayıcı, titanyum-tungsten karışımı sigortaları yakmak suretiyle istenilen bilgiyi kalıcı olarak PROM'a kaydeder.

EPROM (Erasable Programmable Read Only Memory = Silinebilir programlanabilir bellek): EPROM'lar ROM ailesi içinde en çok ilgi çeken ve pek çok kullanım alanı bulan bir bellek çeşididir. En önemli özelliği, bilginin kaydedilip daha sonra bellek kızıl ötesi ışığa maruz bırakılarak silinip tekrar programlanmaya hazır hale getirilebilmesidir. Programlamak için özel EPROM programlama cihazına ihtiyaç vardır. Silmek için ise EPROM'un üzerindeki quartz pencereyi 10 Watlık bir ultraviole lamba altında az bir uzaklıktan ortalama 10-15 dakika tutmak yeterlidir. Üretici firmaların veri raporlarında bir EPROM'a yazıp silme işleminin 100 defaya kadar mümkün olduğunu yazarlarsa da pratik uygulamalarda bu işlem ortalama 10-15 defayla sınırlıdır.

Hali hazırda 1 Kilobayttan 64 Kilobayta kadar EPROM üretilmektedir. Fakat ne yazık ki üreticilerin aralarında bir standart belirlememesi yüzünden EPROM'lar arasında küçük farklılıklar olmaktadır. Bütün bunlara rağmen EPROM'lar araştırma-geliştirme, özel projeler, program kartuşları ve daha birçok uygulamalarda yaygın olarak kullanılmakadır.

İlerki sayılarda EPROM'lara daha ayrıntılı olarak değinecek ve evinizde uygulayabileceğiniz basit EPROM programlayıcı ve silici devreleri vereceğiz.

EEPROM (Electrically Erasable PROM = Elektrikle silinebilir-programlanabilir okunabilir bellek): Fonksiyon olarak EPROM'a benzemesine karşın elektrikle silinebilme avantajına sahiptir. Yani silmek için bir ultraviole ışık kaynağına tutmanız gerekmez. Buna karşın üretim işlemleri biraz komplikedir ve pahalı bir bellek çeşididir. En büyük dezavantajı ise bir baytlık bilgi yazmak için yaklaşık 10 milisaniyeye ihtiyaç duvulmasıdır. Bu nedenle yazma işlemi sıradan bir R/WM bellekten yaklaşık 50 kez daha yavaştır ve ortalama 1000 defa yazma işlemi gerçekleştirilebilir.

BELLEKLERDE YENİ KİLOMETRE TAŞLARI

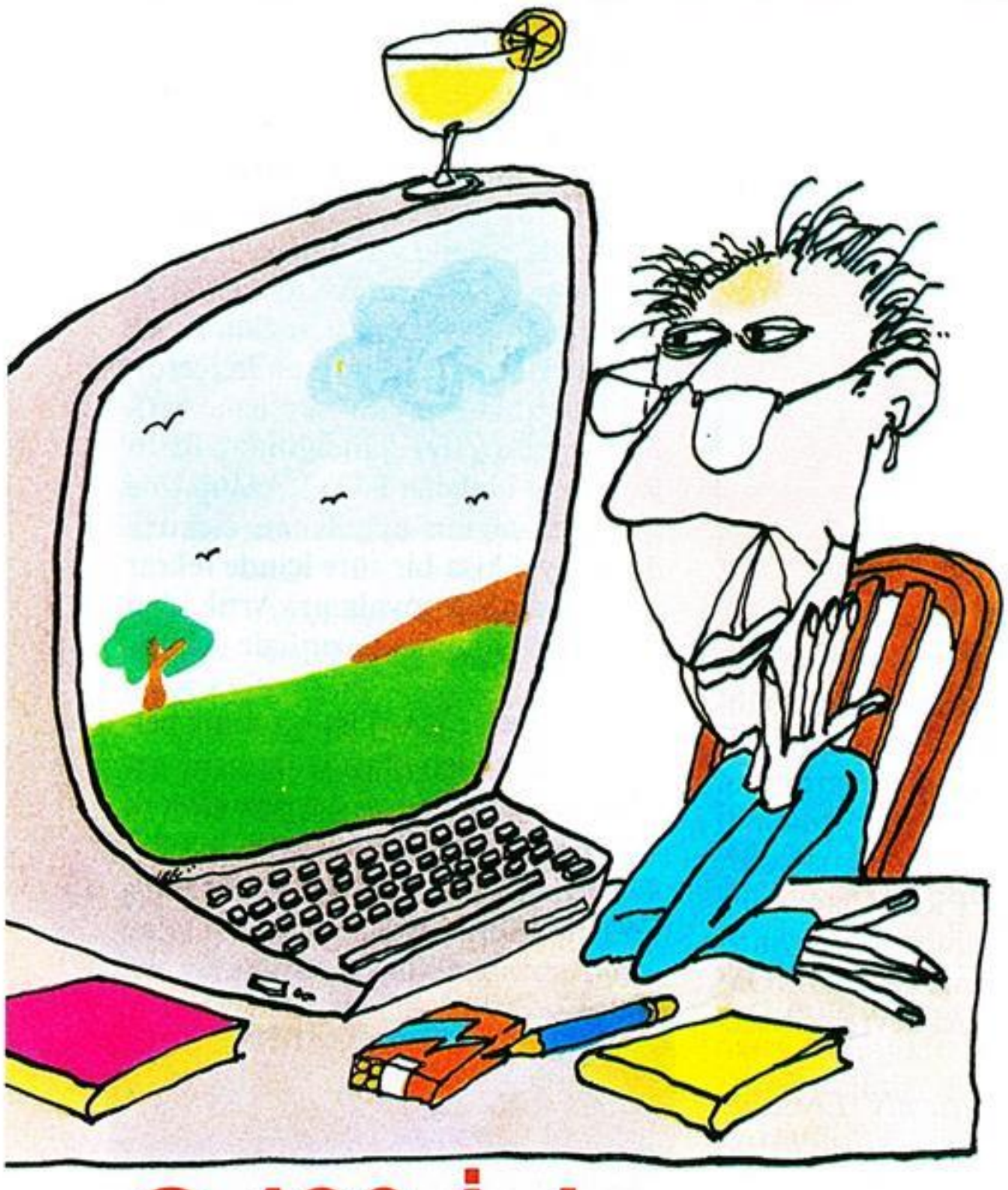
Yukarıda anlatılan bu kadar bellek çeşidine rağmen bilgilerin hızlı transferi ve her türlü şartlara karşı korunması ihtiyacı karşısında yarıiletken malzeme üreticilerini bu ihtiyacı karşılayacak yeni bellek çeşitleri geliştirmeye zorlamıştır. Yeni kuşak bellekler R/WM'lerle ROM'ların birleştirilmiş şekli görünümündedir. Bunlardan ikisi NOVRAM (Non-Volatile Random Access Memory = Uçucu olmayan rastgele erişimli bellek) ve ZERO POWER RAM'dir (Sıfır güç rastgele erişimli bellek).

NOVRAM bir statik bellek ile bir EEPROM'dan oluşmuştur. Besleme gerilimi varken statik bellek çalışır. Besleme gerilimi kesilmesi halinde bilgilerin kaybolmasını önlemek için NOVRAM'ın özel işlevli bacaklarına verilen darbelerle statik bellekteki bilgilerin 10-15 milisaniye içinde EEPROM'a kopyalanması sağlanır. Bu işlemle statik belleğin o anda içerdiği bilgi saklanmış olur. besleme gerilimi tekrar uygulandığında EEPROM'daki bilgiler NOVRAM'ın özel işlevli bacağına uygulanan elektrik darbesiyle kısa bir süre içinde tekrar statik belleğe kopyalanır. Artık bilgi yeniden kullanıma hazır hale gelmiştir.

Zero Power RAM ise bir statik bellek ile bir pilden oluşur. Besleme gerilimi kesildiğinde pil devreye girerek statik belleğin çalışması için gerekli enerjiyi sağlayarak bilginin kaybolması önlenir.







C-128 İçin Püf Noktaları (4)

BURAK KİPER

Yazımıza ilginç bir bilgi ile başlayalım: C128 modunda çalışırken C64 moduna geçmek istediğinizde GO64 komutu vermeniz yeterli olur ama o anda bilgisayarınızda bulunan programınızı da kaybedersiniz. Kaybettiğiniz program bilgisayarın RAM bölgesinde bulunduğundan ve hem C64 modunda iken hem de C128 modunda iken (RAM BANK 0) bilgisayar aslında aynı RAM bölgesini kullandığından geçiş sırasında kaybettiğiniz programı tekrar geri getirmeniz mümkündür. Bu nasıl oluyor? GO64 komutunu verdiğinizde bilgisayar sanki yeni açılmış gibi kendisini C64 modu için ayarlıyor ama C128'de iken çalıştığınız program aslında hafızada kalıyor. Bu programı C64'te nasıl çalışır hale getirebiliriz? Aşağıdaki komutları yazıp LIST komutunu verirseniz C128'deki programınızın geri geldiğini görürsünüz.

POKE 43, 1 : POKE 44, 28

Geri getirdiğiniz programı çalıştırmanız bile mümkün fakat sadece C128'e ait komutların çalışması mümkün değil. Bu metodla C128'e özgü komutları değiştirmek şartı ile programlarınızı C64'e geçirmeniz mümkün olur.

Biraz evvel C128'in RAM BANK 0 ile C64'ün aynı RAM bölgelerini kullandığını söylemiştik. Bunun gerçek olduğunu görmek için aşağıdakileri yazın:

C128 modunda iken:

POKE 8192, 170 :POKE 8193, 85
GO64 (Cevap olarak Y tuşuna basmayı unutmayın)
PRINT PEEK(8192);PEEK(8193)

Bu komutları yazdığınızda C128 modunda POKE ettiğiniz değerleri C64 modunda değişmediğini görebilirsiniz.

Grafik Çizimleri:

Bildiğimiz gibi C128'de grafik çizimler yapabilmemizi sağlayacak yeni komutlar bulunmakta. Halbuki C64'te herhangi bir grafik çizim gerçekleştirebilmek için ya uzun uzun karmaşık POKE komutları ile uğraşmamız gerekecek ya da SIMONS BASIC gibi bünyesine grafik çizmeyi sağlayacak komutlar katmış yardımcı bir program kullanmamız gerekecekti.

Şimdi C128'in kendi BASIC'inde bulunan grafik komutları kullanarak çabuk grafik gösterim sağlayacak bir program görelim. Programın kendisini Program Döküm Eki'nde bulabilirsiniz. Şimdi programımızın açıklamasına geçelim.

Programınızı çalıştırdığınızda ilk olarak sizden grafiğin ismi sorulacak. Sıra kaç değer gireceğinize geliyor. Maksimum sekiz değer seçebilirsiniz, yani ekrana en fazla sekiz çubuk sığdırabiliyoruz. Hemen belirtelim programımızı enteresan hale sokan şey çubukların üç boyutlu olarak çizilmesi. Bu bilgileri girdikten sonra seçtiğiniz sayı kadar her bir çubuk için değer ve çubuğun boyanmasını istediğiniz rengin ko-

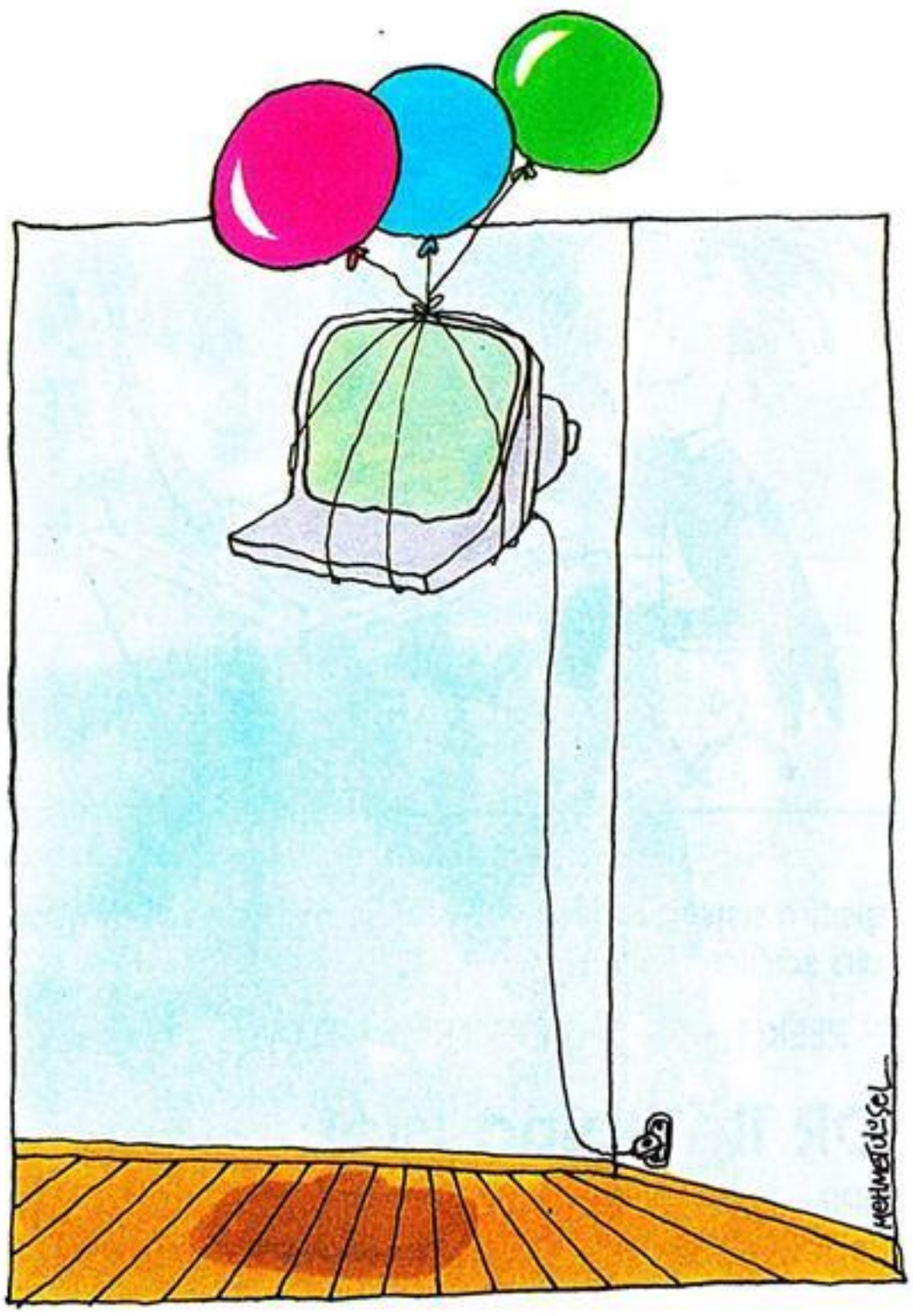
dunu giriyorsunuz. Son olarak çubuğa verilmesi gereken ismi yazmanız gerekiyor. Bu isimler en çok üç karakter olmak zorunda ama eğer daha uzun bir isim girerseniz de programımız girdiğiniz ismi üç karaktere indiriyor. Tüm veri girişleri bittiğinde girilen verilerin doğru olup olmadığı soruluyor, E cevabı halinde çizime geçiliyor, aksi takdirde veriler yeni baştan alınıyor.

Değer için girebileceğiniz en yüksek rakam diye bir sınır yok. Girilen değerlerin en büyüğü yüz kabul edilip diğer değerler buna göre ayarlanıyor, çizimler de buna göre yapılıyor.

Şimdi satır grupları halinde programımızın bazı önemli yerlerini görelim.

100-160: Grafik ekranının renkleri, çubuk isimlerinin yazılması için kolon ve diziler tanımlanıyor.
170-360: Veriler isteniyor.
370-400: Grafik ekranı hazırlanıp grafiğin başlığı yazılıyor.
410-490: Gerekli hesaplamalar yapılarak çubuklar çizilip renklendiriliyor.
500-600: Herhangi bir tuşa bastığınızda karşınıza menü çıkıyor.

Seçeneğinize göre gerekli işlem basamağına atlanıyor.



Dataset:

Sanırım her Commodore kullanıcısı Commodore'unu ilk aldığında veri saklama amacı ile Dataset kullanmıştır. Eğer sabırlı iseniz programlarınızı yükleme ve saklama açısından Dataset bir Disk Drive'a alternatif olabilir.

Yazımızın bu bölümünde Dataset için bazı enteresan bilgiler vereceğiz.

Dataset'i devamlı olarak kullanan kullanıcılar bile genelde teyp motorunun SAVE veya LOAD işlemi sonunda kendiliğinden durduğuna dikkat etmezler. Datasetinizin motoru bilgisayarınızın işletim sistemi tarafından kontrol edilmekte. Bundan başka datasetin moturunu yazacağımız programlarla da kontrol etmek mümkün. Bizim için önemli olan adresler 1 (prosesor portu) ve 192 (teyp motor göstergeci).

Şimdi ufak bir deneme yapalım. Önce datasetinizin PLAY tuşuna basın. Motor çalışır ve dönmeye başlar. Direkt modda aşağıdaki POKE komutlarını yazın.

```
POKE 192,1:POKE1,PEEK(1) OR 32
```

Dikkat ettiniz mi? Siz datasetinize dokunmadan motor durdu. Datasetinize dokunmadan motoru tekrar çalıştırmak isterseniz aşağıdaki POKE komutunu yazın.

```
POKE 1, PEEK(1) AND 39:POKE 192,0
```

Bize datasetimizin motoruna uzaktan kumanda etme fırsatı veren adres 1 numaralı adres. Bu adresin içeriğini BIT seviyesinde inceleyelim:

BIT3 :Bu BIT yanıksa dataset yazma durumundadır.
BIT4 :Datasetin tuşlarının durumu.
BIT5 :Dataset moturunun durumunu gösterir.

Motoru çalıştırabilmek için BIT5'i yakmamız yani bu BIT0in değerinin 1 olması gerekiyor.

Dataset Tuşlarını Okuma:

1 numaralı adresin yaptıkları bununla bitmiyor. BIT4 datasetinizdeki tuşlardan herhangi birinin basılı olup olmadığını kontrol eder. Dataseti kullanırken karşınıza çıkan PRESS PLAY (&RECORD) ON TAPE mesajları işletim sisteminde bu bitin durumuna göre karşımıza çıkar. Ne yazık ki bilgisayarımız datasetin hangi tuşuna bastığımızı fark edemiyor.

Mesela bir programı SAVE edeceksiniz. Normalde PLAY ve RECORD tuşlarını aynı anda basmanız gerekirken diyelim ki kazara sadece PLAY tuşuna bastınız. Bu durumda işletim sistemi hangi tuşlara basıldığını anlamadığından hatanın farkına varamayacaktır.

BIT4'ün dataset tuşlarını kontrolünü şu şekilde sağlayabiliriz:

WAIT 1,16 (Datasetteki STOP tuşu basılana kadar bekle.)
WAIT 1,32,32 (Datasetteki herhangi bir tuşa basılana kadar bekle.)



İşletim sisteminin RUN-STOP tuşunu kontrol etmesini şöyle sağlayabiliriz:

```
IF PEEK(1)=99 THEN PRINT "BASILDI"
```

DDR ile İlginç İşler:

DRR nedir ve ne tür ilginç işler yapar?

Birçok iş yapabilir. Prosesor portu esas olarak CIA'-lar tarafından kontrol edilen Giriş/Çıkış Portlarıdır. Bu portun altı hattı vardır. Bunlardan üçü Dataset için ayrılmıştır. Bu hatlar hem giriş hem de çıkış olarak kullanılır. Hatların hangi yönde kullanıldıkları prosesor portu için DDR (DATA DIRECTION REGISTER "Veri Yönlendirme Sicili") tarafından belirlenir: Adres 0. Adresin içeriğini BIT düzeyinde inceleyelim:

BIT3 :Veri yönü dataset yazma
BIT4 :Veri yönü dataset tuşlarının durumu
BIT5 :Veri yönü dataset motor durumu
Bit değeri
0=GİRİŞ
1=ÇIKIŞ

Dataset yazma ve dataset motor durumu ÇIKIŞ'dır. Bunlar kasete yazma durumunda iken dataseti etkilerler. BIT4 ise daha evvel de belirttiğimiz gibi Datasette bir tuşa basılıp basılmadığını kontrol eder.

Dataset'te Program Kilitleme Teknikleri:

Eğer DDR'deki bitleri değiştirirsek ne olur? Aşağıdaki POKE komutu BIT4'ü bir ÇIKIŞ hattı haline getirir:

```
POKE0,PEEK(0) OR 2  4
```

Bu komut sayesinde ne elde edeceğiz? Üstteki komutu girmemizle birlikte artık BIT4 datasetteki tuşların durumunu anlayamaz hale gelecek. İşletim sistemi PRESS PLAY(&RECORD) ON TAPE mesajı verecek. İstenileni yapsanız bile işletim size OK veya LOAD veya SAVE mesajı göndererek memnuniyetini bildirmeyecek. Aşağıdaki POKE komutunu girerseniz bilgisayarınızı bu ümitsiz durumdan kurtarabilirsiniz.

```
POKE 0, PEEK(0) AND NOT 2  4
```

Başka bir kilitleme yöntemi de BIT3'ü değiştirerek olur:

```
POKE 0, PEEK(0) AND NOT 2  3
```

Bu komuttan sonra her şey normal gidecek çıkmasını beklediğiniz mesajlar çıkacak, datasetteki tuşların durumu kontrol edilecek, SAVE komutu çalışır gözükecek, ama çalışmayacak. Normale dönmek isterseniz:

```
POKE 0, PEEK(0) AND NOT 2  4
```

Bu soruna çare aşağıdaki POKE komutudur.

```
POKE 0, PEEK(0) OR 2  4
```

Müzikseti Olarak Dataset:

Evet yanlış okumadınız, datasetinizi müzik seti olarak kullanabilirsiniz. Nasıl mı? Program döküm ekinde bulacağınız programı yazıp çalıştırın. Datasetinize de sevdiğiniz bir müzik kaseti koyup PLAY tuşuna basın, televizyon veya monitörünüzün sesini açıp keyifle kasedinizi dinleyin. Ama bizden söylemesi: Datasetten elde edeceğiniz müziğin kalitesi için hiç de iyi şeyler söyleyemeyeceğiz, hayal kırıklığına uğramayın.

Verdiğimiz programın yaptığı, prosesor portundaki üç GİRİŞ/ÇIKIŞ hattını kontrol etmekten ibaret. Dataset ile bilgisayarımız arasında kullanılmayan bir hat daha var: Dataset okuma (bu hat Dataset yazma hattına eşdeğerdir). Datasetten gelen bütün veri bu hattan gider. Fakat ufak bir problemimiz var: Bu hat diğer hatlar gibi prosesorde son bulmaz -CIA1'e kadar devam eder (böylece Seri Port'u bizim kullanımımıza açar). Bu yüzden CIA1'i 13. sicilini kurmamız gerekiyor:

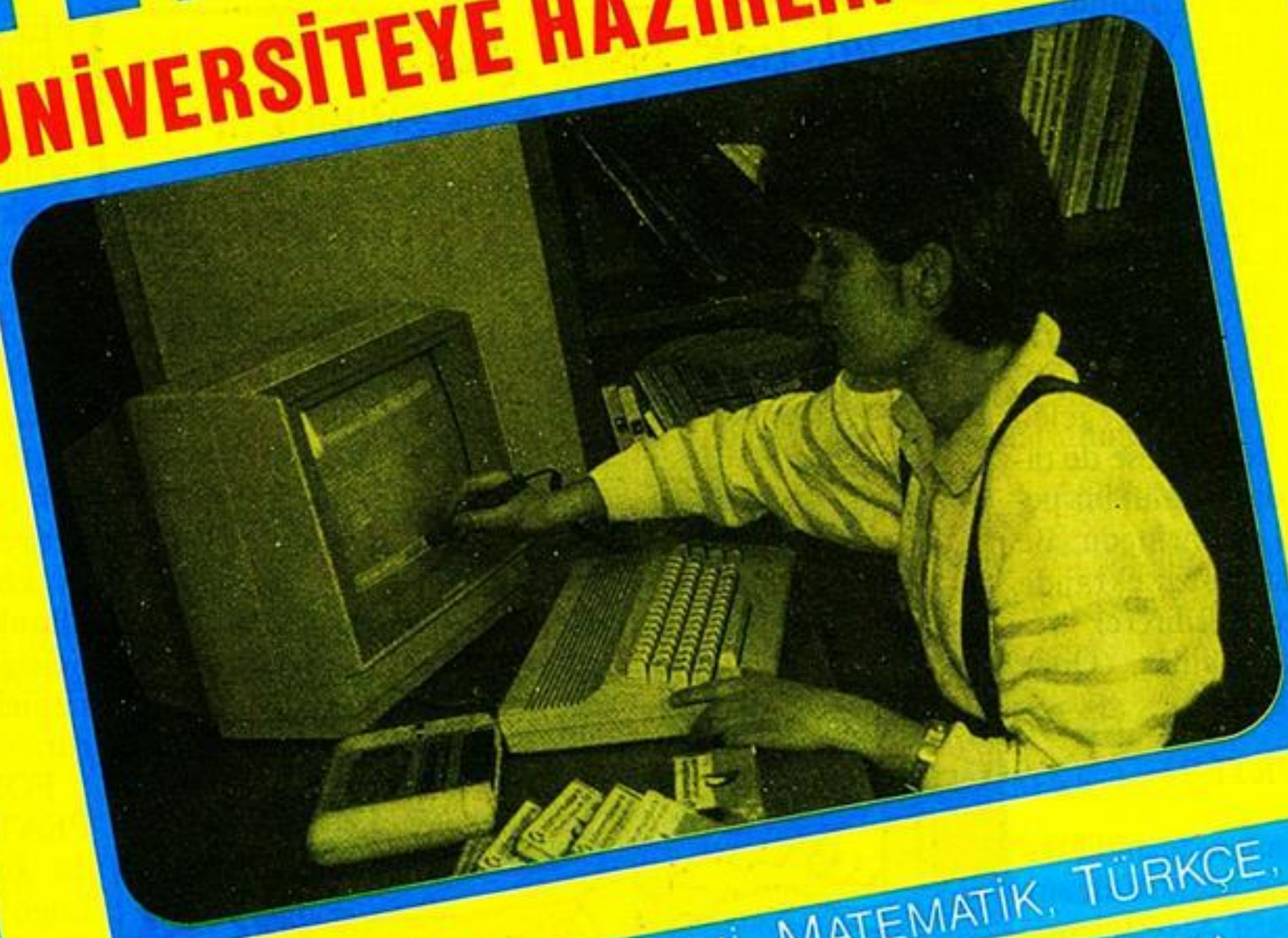
Aşağıdaki Makine Dili ile uğraşanlar için aynı programın Makine Dili'nde yazılmış şeklini bulabilirsiniz.

```
1A00 A0 00      LDY # $00   : SES 0
1A02 AD 0D DC   LDA $DC0D   : CIA1'deki FLAG'ının
                              kontrolü
1A05 C9 00      CMP # $00   : Sinyal yok mu?
1A07 FO 02      BEQ $1A0B   : Evet-devam et
11A09 A0 OF     LDY # $0F   : Hayır-Sesi kapat
1A0B 8C 18 D4   STY $D418   : ve 24 numaralı SID
                              yazmacı
1A0E A5 D5      LDA $D5     : Klavyeyi oku
1A10 C9 58      CMP # $58   : 88=hiçbir tuşa
                              basılmadı
1A12 F0 EC      BEQ $1A00   : Hiçbir tuş-devam et
1A14 60         RTS         : BASIC'e geri dön
```

Bu ufak programımızı SYS DEC ("1A00") komutu ile başlatın. Programımızın çalışma prensibi gayet basit: Sinyal kontrolü yapılır. Eğer sinyal varsa ses açılır, yoksa kapatılır. Sesi açıp kapama Makine Dili'nde mümkün olan en ufak döngü içinde yapılmaya çalışılıyor ama sonuçta korkunç bir ses ortaya çıkıyor.

TELE TEST 64

ÜNİVERSİTEYE HAZIRLIK PAKETİ



FİZİK, KİMYA, BİYOLOJİ, MATEMATİK, TÜRKÇE,
SOSYAL BİLGİLER, İNGİLİZCE, FRANSIZCA
DALLARINDA 1000'E YAKIN SORU

- 6 Kaset
- Işıklı Kalem
- Kullanım Kılavuzu
- Çizim ve puanlama programları

Commodore COMPUTER

Teleteknik yetkili satıcılarında

KDV DAHİL
25.000.- TL.

Bu sayıdan itibaren yeni bir bilgisayar diline başlıyoruz...

Logo Öğreniyoruz

AFİF SAY

İki sayı önce logo dilini tanıtmış, özelliklerinden kısaca bahsetmiştik. Bu dizide ise, adım adım Logo'yla neler yapılabileceğini ve nasıl yapılabileceğini yani Logo'ca düşünmeyi, iletişim kurmayı öğreneceğiz. Önce Logo'nun ABC'sinden, turtle grafiklerinden başlıyoruz. Kaplumbağa anlamına gelen turtle, pek benzemese de tabanı biraz daha kalın bir üçgenle temsil edilmektedir. Bu sevimli dostumuzu ekranda dilediğimizce gezdirerek grafikler çizdirebiliriz. Önce onu görelim:



SHOW TURTLE

İşte ekranın ortasında tepesi yukarıya dönük duruyor. Bu BAŞLANGIÇ pozisyonudur. Nerede olursa olsun "HOME" dediğimizde, bu konuma dönecektir. Bu durumda tepesinin baktığı açı yani yukarısı Ø derece kabul edilir. Şimdi biraz gezinelim.

FORWARD 20

Bizim tosbağa arkasında bir iz bırakıp öne yani ekranın yukarısına doğru yirmi adım attı. Logo'da uzunluklar bir piksel'den oluşan adımlarla ifade edilir ve turtle ilerlerken kuyruğu ile arkasında bir iz bırakır. Çizmeden ilerletmek için kuyruğunu kaldırması gerekir:

PENUP
FORWARD 20
PENDOWN
FORWARD 20
Şimdi biraz çevirelim:
LEFT 90

Gördüğünüz gibi turtle'ın kafası 90 derece sola döndü. LEFT açı ve RIGHT açı komutlarıyla turtle'ın gidiş-dönüş yönünü bulunduğu duruma göre istediğimiz açıya çevirebiliriz.

FORWARD 20

İşte bir öncekine dik bir çizgi çizdirdik.

DRAW

Bütün ekran silindi ve bizimki başlangıçtaki konumuna döndü. Bu, açılıştaki çizim işlemine başlangıç komutudur. Şimdiye kadar öğrendiklerimizle bir kare çizdirebiliriz.

FORWARD 50
LEFT 90
FORWARD 50
LEFT 90
FORWARD 50
LEFT 90
FORWARD 50

Bir kare için bu kadar uğraşılır mı? Bizim turtle böyle tekrarlanan komutları daha kolay da anlayabilir.

REPEAT 4 [FORWARD 50
LEFT 90]

yukarıdakiyle aynı işi görecektir. Tekrar tekrar yapılacak işlemleri bir liste haline getirebilir ve yukarıdaki gibi köşeli parantezler içine alarak istediğimiz sayıda yineletebiliriz. Logo işlemleri, nesneleri hep liste olarak görür.

FORWARD, LEFT, REPEAT gibi sözcükler Logo'da önceden tanımlanmış, Logo'nun bildiği sözcüklerdir. Biz de ona varolanlardan yararlanarak yenilerini öğretebilir, hatta kendi öğrettiklerimizi başkalarını tanımlamada kullanabiliriz; Logo'da gerçek anlamda program yoktur, tıpkı bizdeki gibi kavramlar ve bunları simgeleyen sözcükler vardır. Şimdi KARE'yi görelim.

TO KARE

Ve ortam değişti birden. Koyu renkli bir zeminde TO KARE yazısı yukarda olmak üzere ekran silindi. Altta ise

EDIT: CTRL-C TO DEFINE, CTRL-G TO ABORT

yazısı çıktı. Bu, yeni sözcükler tanımlamada kullanacağımız yazılım modu'dur.

Şimdi karenin nasıl çizileceğini tanımlayalım:

REPEAT 4 [FORWARD 50
LEFT 90]

Tanımlama bittikten sonra yaptığımızı Logo'nun tanıması için Control ve C tuşlarına aynı anda basalım.

KARE DEFINED

Logo KARE sözcüğüne dağarcığına kattığını belirtti. Bundan sonra,

KARE

yazarak istediğimiz an kare çizdirebiliriz.

LEFT 20
KARE

Turtle'ın bu konumuna göre eğri bir kare elde ettik.

REPEAT 18 [LEFT 20 KARE]

Turtle her seferinde 20 derece dönerek onsekiz kare çizdi. Bu arada dikkatinizi çekti mi bilmem, dönüş açısıyla tekrarlama sayılarını çarpımı hep 360 oluyor (kare için 4×90, şimdiki için 20×18). Bu tüm kapalı şekiller için geçerlidir. Tüm poligonlar için geçerli olan kural: Dönme açısı = 360/kenar sayısı'dır.

Tanımlarken

REPEAT kenar sayısı [LEFT 360/kenar sayısı FORWARD adım]

şeklinde ifade edebiliriz.

Buna göre üçgen

TO ÜÇGEN
REPEAT 3 [LEFT 120
FORWARD 50]
END

olabilir. Beşgen, altıgen ve diğerleri için tanımlamayı siz yapın.

Şimdi KARE'yi biraz geliştirelim.

EDIT KARE

Tekrar yazılım moduna geçtik. Şimdi imleçi TO KARE satırının sonuna getirelim ve arada bir boşluk bırakmaya dikkat ederek aşağıdakileri aynı satıra ekleyelim (sonda RETURN'e basmadan):

:KENAR

yazdıktan sonra, REPEAT 4[FORWARD 50 LEFT 50] satırına gelip cursor ve delete tuşlarıyla 50'yi silip yerine :KENAR yazalım. Bitince aşağıdaki gibi olacaktır:

REPEAT 4[FORWARD :KENAR
LEFT 90]

Şimdi yaptığımız değişiklikler Logo'ya anlatmak için Control ve C tuşlarına basıp çıkalım.

KARE

yazdığımızda Logo bize,

KARE NEEDS MORE INPUTS

diye cevap verecektir. Yani daha veri vermemiz gerektiğini söylemektedir.

KARE 70

Yaptığımız kare biraz daha büyüdü. Yeni tanımladığımız bu sözcükle istediğimiz büyüklükte kare çizdirebiliriz. Logo, sözcüğün ardından gelen sayıyı :KENAR nesnesine atamakta ve bu değeri işlem içinde kullanmaktadır.

TO ŞEKİL :KENAR
REPEAT 18 [KARE :KENAR
LEFT 20]
END

Bu tanımlandıktan sonra

ŞEKİL 50

diyerek şekil çizdirebiliriz. Böylece



daha önce öğrettiğimiz bir sözcüğü kullanarak yeni bir sözcük tanımlamış olduk. Logo tanımladığımız her sözcüğü kendisinin önceden bildiklerinden farksız görür. Bunun anlamı tanımladıklarınızı da varolanlar gibi kullanabilmenizdir.

TO ŞEKİL1 :KENAR
REPEAT 18[KARE :KENAR
LEFT 20 MAKE''KENAR
:KENAR +5]
END

Bu yeni tanımlamada MAKE sözcüğü ile KENAR nesnesinin taşıdığı değeri her seferinde artırarak şekil çizdirebiliriz. Dikkat edilecek konu bir nesneye değer atfedileceği zaman '' işaretiyle bunun yeni bir değer olduğunu belirtmemiz gerektiğidir. Varolan değerini kullanırken ise nesnenin adından önce : işaretini koyarız. Logo'da değişkenler tek tiptir, yani sayısal ya da başka tür değişkenler yoktur ve bunlar nesne adını alırlar. Nasıl KARE sözcüğünü yapılacak bir dizi işlemi içeren listeyi ifade etmek için kullanıyorsak, KENAR nesnesini de ona bağlı bir değeri ifade etmek için kullanıyoruz. Okula gitmek bizim için her sabah yaptığımız bir işlemler dizisini ifade eder. Harçlık ise parasal bir değeri ifade eder ve aldığımız miktara göre zihnimizde canlanan, ismi aynı kalmasına rağmen, değişir. Gördüğümüz gibi Logo aslında bizim zihnimizin çalışmasına benzer, ilkelerle çalışır. Bu nedenle BASIC gibi diğer programlama dillerinin mantığını Logo'yu öğrenirken bir kenara bırakmak, doğrudan kendimiz gibi düşünmek zorundayız. Şimdiye kadar yaptıklarımızı kaydetmek için disket sürücüsüne formatlanmış bir disket koyup

SAVE '' ÇALIŞMA1

yazalım. Bu şekilde yaptığımız tüm çalışmalar kaydedildi.

CATALOG

yazdığımızda,

ÇALIŞMA1, LOGO

adını görebiliriz. Kaydettiğimiz, çalışma alanındaki tüm çalışmaları içeren bir listedir.

Başka bir zaman bunları kullanmak için,

READ '' ÇALIŞMA!

dememiz gerekir.

PRINTOUT TITLES

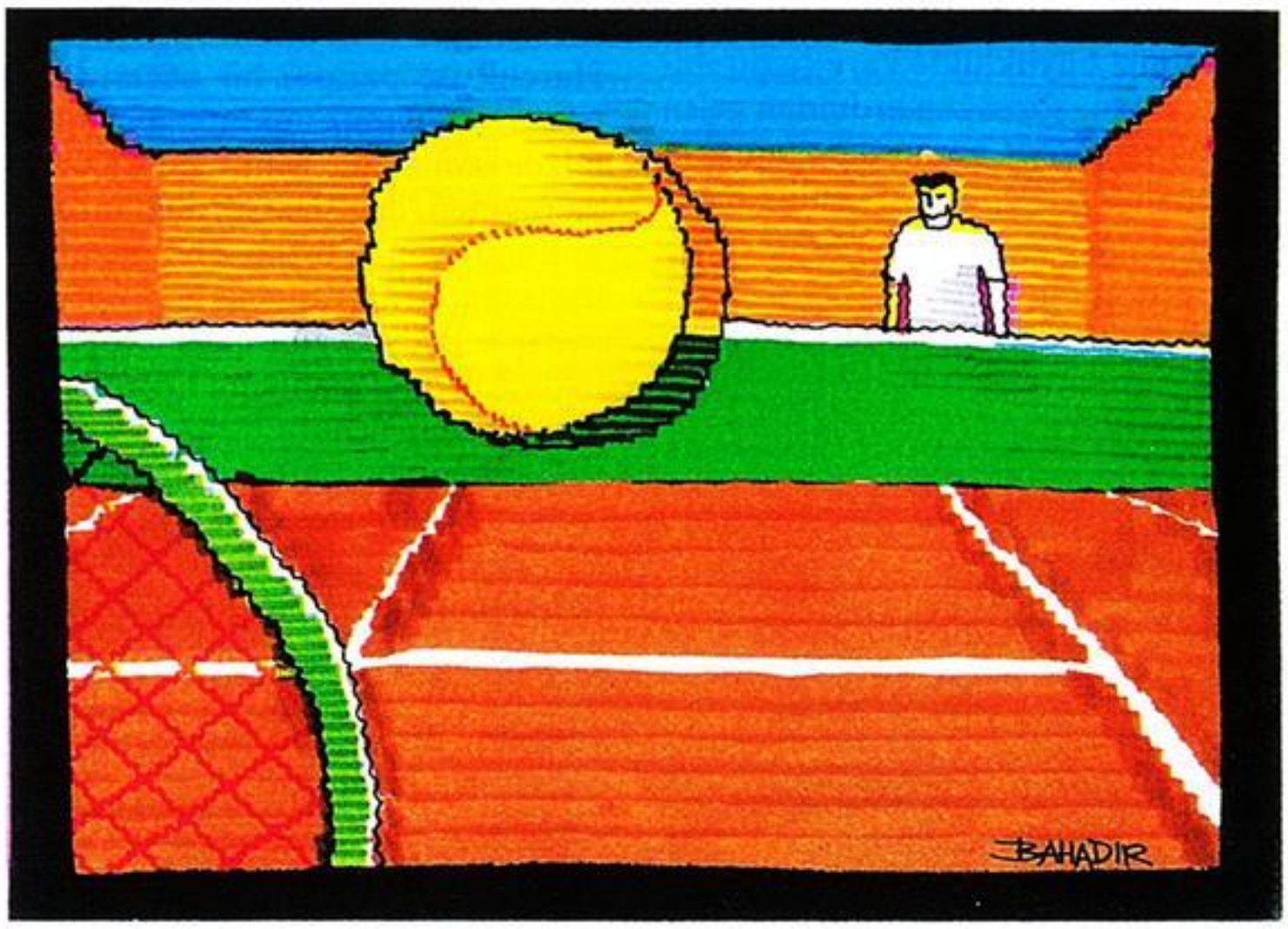
dersek çalışma alanımızda varolan tanımlamaların bir listesini görebiliriz. Şimdi GOODBYE diyerek çalışmayı bitirelim. Bu şekilde çalışma alanı tamamen silinip, Logo ilk açıldığı ana geri döner.

Hoşçakalın... □



Pascal (5)

BURAK KİPER



Geçen bölümde döngülere başlamış ve FOR...DO döngü çeşidini incelemiştik. Bu döngü, eğer tekrarın ne kadar yapılacağını biliyorsak kullanışlı olabilir. Tekrarın sayısını her zaman bilemeyiz ama hangi durumda tekrarın biteceğini bilebiliriz. Başka bir deyişle, döngümüzün belirli bir durumunun doğru olduğu sürece devam etmesini isteyebiliriz. Belirli durumumuz yanlış olduğunda ise döngü sona erer. Belirli durum ne olabilir? Mesela kullanıcıdan alınan cevap "evet" olduğu sürece döngüye devam eder, cevap "hayır" olarak verildiğinde döngüyü sona erdirerek bir sonraki komut gruplarına geçebiliriz.

Döngü çeşitlerinden WHILE komutu döngüyü bu şekilde kontrol etmemizi sağlar. WHILE komutunun genel şekli şöyledir:

WHILE Boolean İfadesi DO Komut

Bilgisayar ilk önce Boolen İfadesi'ne bakar, eğer bu ifadenin değeri doğru (TRUE) ise DO'dan sonra gelen komutu işler. Döngü Boolean İfadesi'nin değeri TRUE olduğu sürece devam eder. Boolean İfadesi'nin değeri yanlış (FALSE) olduğu andan itibaren döngü sona erer ve kontrol bir sonraki komuta geçer. Bu arada Boolean veri tipinin sadece iki durumu olduğunu belirtelim: TRUE veya FALSE. Mesela DEVAM değişkenini VAR bildiriminde Boolean veri tipi olarak bildirdiyseniz, bu değişkeninize ya DEVAM:=TRUE veya DEVAM:=FALSE; şeklinde bir değer verebilirsiniz. Programınızda WHILE DEVAM DO I:=I+1; şeklinde bir komut kullandığınızda DEVAM değişkeninizin değeri TRUE olduğu sürece I değişkenimizin içeriği bir artar.

FOR döngüsünde olduğu gibi, tekrarlanan tek bir komut olabileceği gibi, BEGIN...END bloğu içinde çoklu komutlar da olabilir.

Şimdi WHILE...DO komutu ile ilgili olarak anlattıklarımızı birkaç örnekle inceleyelim. Örneklerimiz bir programın belirli bir bölümünü oluşturmakta. Örneklerde kullanılan değişkenlerin daha evvelden bildirildiğini kabul edeceğiz.

```
I:=0;
WHILE I:<=20 DO
   I:=I+5;
```

Örneğimizde I:=I+5 komutu devamlı tekrarlanmakta ve bu komutun her tekrarlanışında I değişkeninin değeri 5 artmaktadır. Böylece I değişkeni sırayla 0,5,10,15,20 ve 25 değerlerini alır. Komutun her tekrarlanışından evvel I değişkeninin değerinin 20'den küçük olup olmadığı kontrol ediliyor. I<=20 ifadesi bizim Boolean İfademizi oluşturur ve bu ifadenin aldığı değere göre döngünün devam edip etmeyeceği belirlenir. I değişkeninin değeri 0,5,10,15 veya 20 iken ifadenin değeri TRUE olacağı için döngü yapılır, fakat I değişkeninin değeri 25 olduğunda ifade FALSE değeri ile sonuçlanacağından kontrol bir sonraki komuta atlar.

Yukarıdaki örnekte tek bir komutun döngü içinde kullanılmasına örnek verdik. Şimde de döngü içinde bir komut grubunun kullanılışına örnek verelim:

```
I:=0;
WHILE I:<=20 DO
  BEGIN
    WRITE(I:4);
    I:=I+5
  END;
```

Dikkat ettiyseniz WHILE...DO'dan sonra döngüde yer almasını istediğimiz komutları BEGIN...END bloğu içine aldık. Yazımızın ilk başlarında bir PASCAL programının BEGIN ile başlayıp END ile bittiğini söylemiştik. Bundan başka programımızın herhangi bir yerinde üstteki örnekte de olduğu gibi belirli bazı komutların tek bir komut gibi işlem görmesini istersek BEGIN...END bloğu içine alırız.

Programımızı çalıştırdığımızda şöyle bir çıktı alacağız:

```
0 5 10 15 20
```

Örneğimizin başında ı değişkenine sıfır değeri vermiştik. Eğer ı değişkenine 21 değeri verseydik döngü hiç yapılamadan kontrol bir sonraki komuta aktarılırdı. Yani Boolean İfadesi ilk işlendiğinde FALSE sonucu verirse, döngü içindeki komutlar hiç tekrarlanmadan kontrol bir sonraki komuta geçerdi.

İsterseniz şimdi de yeni öğrendiğimiz döngü komutunu bir örnek programda kullanalım. Programımızda kaç gün içinde 1.000.000 TL kazanacağımızı hesaplayacağız. Fakat parayı kazanmamız bir kurala bağlı: İlk gün 1 TL, ikinci gün 2 TL, üçüncü gün 4 TL, dördüncü gün 8 TLyani her gün bir evvelki günün iki katı para kazanacağız. Programımızda döngünün ne kadar süreceğini bilemediğimiz için de, WHILE komutunu kullanacağız. Zaten programımızın amacı da, döngünün ne kadar süreceğini bulmak. Program ve açıklaması için program döküm ekine bakınız.

Döngülerin en çok kullanıldığı yerlerin başında bir seri bilginin okunup işlenmesi gelir. Mesela bir işyerindeki işçilerin maaşlarının, çalışma sürelerinin okunması gibi. Burada karşımıza bilgilerin okunmasına ne zaman son vereceğimiz problemi çıkıyor.

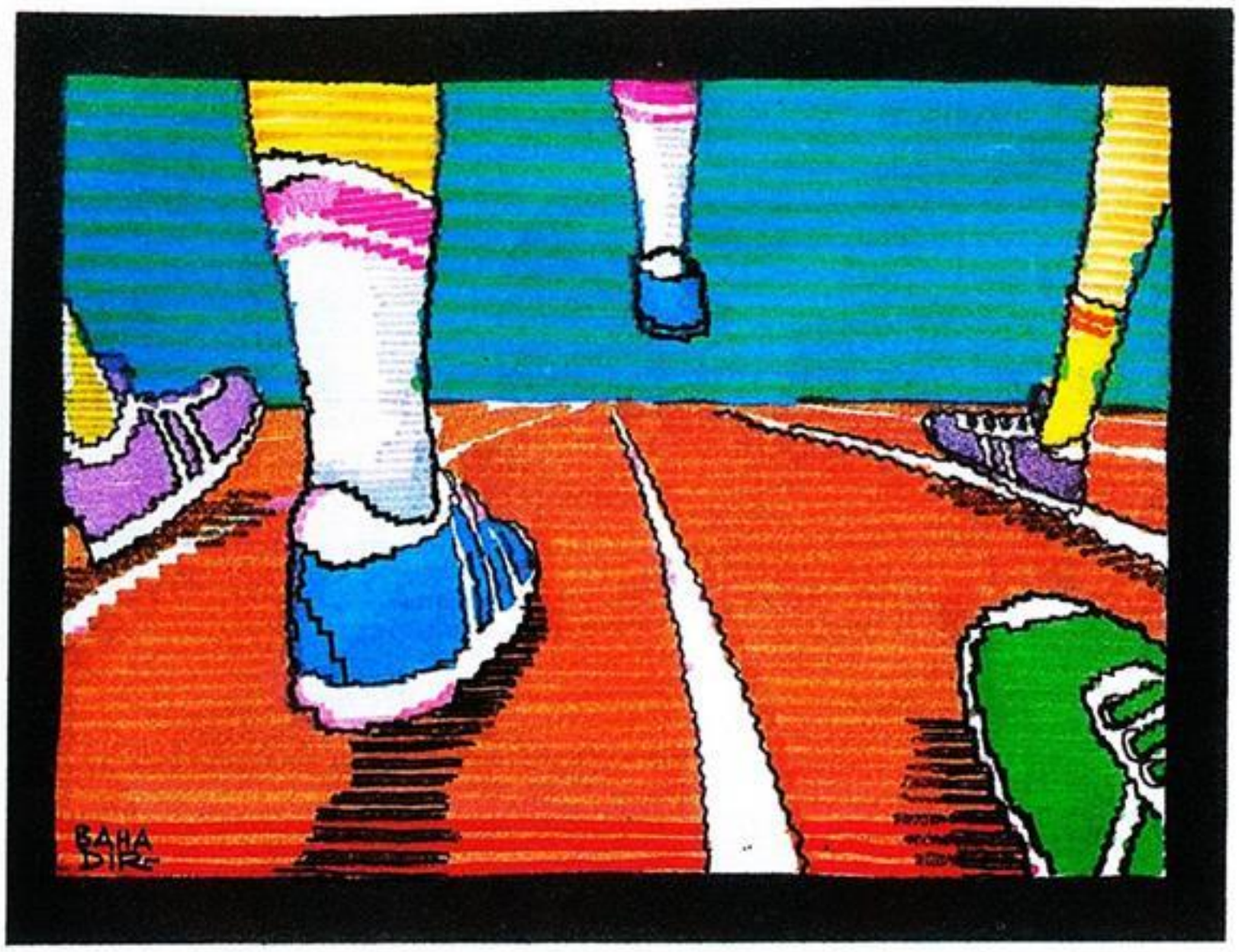
İlk çözüm olarak bilgilerin okunmasına başlamadan evvel, okuyacağımız ilk değeri veri sayısı olarak alıp, bir FOR döngüsü ile verileri okuyabiliriz. Bu tip veri girişine şöyle bir örnek verelim:

```
5 3.87 9.65 5.61 7.76 9.19
```

5, bize okunacak beş veri olduğunu bildiriyor. Bu yöntem programınızın başında kaç veri işleyeceğinizi biliyorsanız işe yarayabilir. Fakat işleyeceğiniz veri sayısını bilmiyorsanız ikinci yöntemimizi uygulamamız gerekecek.

İkinci yöntem olarak istediğimiz özel bir değer kullanacağız. Veri okumamız bu özel veri girilinceye kadar devam edecek. İstediğimiz özel değeri saptarken, bu değerin isteyeceğimiz değerlerle karışmaması için çok dikkatli olup, seçimimizi buna göre yapmalıyız. Mesela 0 ile 100 arasında veriler işleyeceğiz. Bu durumda 999 özel değer olarak çok uygun bir seçim olur.

Özel değer okuyarak veri işlediğimiz durumlarda yapacağımız işlerin



genel hali şöyledir:

İlk veriyi okuyun
WHILE (Okuduğunuz değer özel değere eşit değilse) DO
Verileri işleyen komutlar
Bir sonraki veriyi okuyun

Bu genel duruma bir örnek verelim. Örneğimizde özel değer 999.0 ve kullanılan değişkenlerin daha evvel tanımlandığını kabul ediyoruz;

```
READ(x);
WHILE x<> 99.0 DO
  BEGIN
    y:=x*2;
    READ(x);
  END
```

İlk READ komutu ilk veriyi okuyor. WHILE komutu bunun özel değer olup olmadığını kontrol ediyor. Eğer özel değer değilse okunan değer işlenip yeni bir değer okunuyor, bu işlemler okunan değer özel değere eşit olana kadar devam ediyor. Okunan değer özel değere eşit olduğunda, döngü sona erip kontrol bir sonraki komuta aktarılıyor.

Bu tür kullanımda okunan değerin işlenmeden evvel daima kontrol edildiğine dikkat edelim, böylece eğer okuduğumuz değer özel değerse bunu işleme sokmuyoruz.

Son olarak WHILE..DO komutunu kullanırken nelere dikkat etmemiz gerektiğine değinelim. Döngüye girmeden evvel döngüyü kontrol edecek değişkenin ilk olarak bir değere eşitlenmesi gerekir. Döngü içinde döngüyü kontrol eden değişkenin değerini her seferinde değiştirmezseniz hiçbir zaman döngünün içinden çıkamazsınız.

FOR döngüsünü kullanırken döngüyü kontrol eden değişkenin artışı birden farklı ise bunun yerine WHILE döngüsü kullanabilirsiniz. Örneğin:

```
ı:=0
WHILE ı<=100 DO
  BEGIN
    ı:=ı+5;
    WRITE(ı:3);
  END
```

REPEAT...UNTIL

PASCAL'da döngüleri kontrol etmek için bir başka döngü çeşidimiz daha var. Bu döngü çeşidinin yapısını inceleyelim:

```
REPEAT
  komut;
    "
    "
  komut
UNTIL Boolean İfadesi
```

REPEAT...UNTIL yapısı içinde döngüye giren komutların başına BEGIN...END eklememiz gerekmiyor, çünkü REPEAT...UNTIL yapısı bu görevi yerine getiriyor.

REPEAT...UNTIL arasındaki komutlar işlendikten sonra Boolean İfadesi işlenir. Eğer bu ifade TRUE ve-



rirse döngü sona erer ve programın akışı bir sonraki komuta geçer. Boolean İfadesi FALSE verirse döngü tekrarlanır. Bu yapıyı bir örnekle görelim:

```
I:=0;
REPEAT
  WRITE(I:4);
  I:=I+5
UNTIL I>20
```

Bu işlemin sonucunda şöyle bir çıktı elde ederiz:

```
0 5 10 15 20
```

20 değeri basıldıktan sonra I değişkeni 25 değerine ulaştığından Boolean İfadesi TRUE verir. Böylece işlem sonuçlanır.

REPEAT döngüsünün en önemli özelliği döngü içerisindeki komutların işlenmesinden sonra Boolean ifadesine bakılmasıdır. Bunun anlamı döngü içerisinde bulunan komutların en azından bir kere işleneceğidir.

REPEAT...UNTIL ve WHILE...DO yapılarını kullanarak bir örnek verelim.

```
I:=21;
WHILE I:<=20 DO
  BEGIN
    WRITE(I:4);
    I:=I+5
  END
```

Bu program çalıştığında çıktı alamayız. Programın sonuna I değişkeninin değeri 21 olur.

```
I:=21;
REPEAT
  WRITE(I:4);
  I:=I+5
UNTIL I>20
21
```

Bu program çalıştığında çıktı olarak 21 değerini alıyoruz ve programın sonunda I değişkeninin değeri 26 oluyor.

İki örneğimizdeki fark; birincisinde Boolean İfadesi'nin döngüye başlanmadan evvel kontrol edilmesinden, ikincisinde ise döngünün sonunda kontrol edilmesinden kaynaklanıyor.

REPEAT...UNTIL yapısına örnek program ve açıklaması için program döküm ekine bakınız.

DÖNGÜ ÇEŞİTLERİNİN KARŞILAŞTIRILMASI

PASCAL'da üç çeşit döngü çeşidi bulunuyor: FOR, WHILE, REPEAT. Döngünün ne kadar devam edeceğini biliyorsak FOR döngüsünü kullanabiliriz.

WHILE ve REPEAT döngüleri şarta bağlı döngülerdir. WHILE döngüsü Boolean İfadesi TRUE sonuç verdiği sürece devam eder. REPEAT döngüsü ise Boolean İfadesi FALSE sonuç verdiği sürece devam eder.

Boolean İfadesi'nde değişiklik yaparak REPEAT ile WHILE döngülerini yeniden diğeri şeklinde yazabilirsiniz.

REPEAT döngüsü en azından bir kere çalışır. Buna karşın FOR ve WHILE döngüleri hiç yapılmayabilir.

KONTROL KOMUTLARI

Şimdiye kadar görmüş olduğumuz tüm örneklerdeki komutlar yazıldıkları sıra içinde işlenmekteydiler. Eğer bazı komutların sadece belirli durumlarda çalışmasını istersek ne yapacağız?

Kontrol komutları ile Boolean İfadesi'nin neticesine bağlı olarak bazı komutların çalışmasını, diğerlerinin ise atlanmasını sağlayabiliriz.

TEK YÖNLÜ SEÇİM

Tek yönlü seçimde belirli bir komutu işlemeden evvel Boolean İfadesi kontrol edilir. Boolean İfadesi TRUE verirse komut işlenir, eğer sonuç FALSE olursa kontrol bir sonraki komuta geçer.

PASCAL'da bunu sağlayan IF komutudur. Komutun yapısını görelim:

```
IF Boolean İfadesi THEN
                komut
```

Programın akışı IF komutuna geldiğinde Boolean İfadesi işlenir. Sonuç FALSE ise bir alttaki komuta geçilir, sonuç TRUE ise THEN'den sonraki komut işlenir. THEN'den sonra bir komut işlemek yerine BEGIN...END ile çevrilmiş komutlar grubu da işleyebilirsiniz.

IF komutunu örneklerle görelim:

```
IF sayı=1 THEN
   WRITE('BIR')
IF I<20 THEN
   BEGIN
    I:=I+1;
    WRITE(I:4)
  END
```

İKİ YÖNLÜ SEÇİM

Tek yönlü seçimde Boolean İfadesi'nin neticesine bağlı olarak komut ya işlenir ya da işlenmez. İki yönlü seçimde ise Boolean İfade'nin TRUE olması halinde bir komut, FALSE olması halinde ise diğer komut işlenir. Boolean İfadesi'nin neticesi ne olursa olsun komutlardan birisi muhakkak işlenir.

İki yönlü seçimde IF komutunun yapısını görelim:

```
IF Boolean İfadesi THEN
   komut-1
ELSE
   komut-2
```

Bu yapıya bir örnek verelim:

```
IF sayı1>sayı1 THEN
   büyük: = sayı1
ELSE
   büyük: = sayı2
```

ÇOK YÖNLÜ SEÇİM

İki yönden daha fazlası için çok yönlü seçimi kullanabiliriz. Bunun için IF komutunu kullanabileceğimiz gibi daha sonra göreceğimiz CASE komutunu da kullanabiliriz.

k1,k2,k3,k4 herhangi komutları; b1,b2,b3,b4 ise herhangi Boolean İfadeleri'ni temsil etmekte. Buna göre çok yönlü seçim için IF yapısını görelim:

```
IF b1 THEN
   k1
ELSE
   IF b2 THEN
      k2
   ELSE
      IF b3 THEN
         k3
      ELSE
         k4
```

Hangi durumlarda hangi komutların işleneceğini bir tablo ile görelim. Tabloda Boolean İfadeleri'nin durumuna göre hangi komutun işleneceği gösterilmekte.

b1	b2	b3	İşlenen komut
TRUE	TRUE	TRUE	k1
TRUE	TRUE	FALSE	k1
TRUE	FALSE	TRUE	k1
TRUE	FALSE	FALSE	k1
FALSE	TRUE	TRUE	k2
FALSE	TRUE	FALSE	k2
FALSE	FALSE	TRUE	k3
FALSE	FALSE	FALSE	k4

Şimdi Çoklu Seçimi bir örnekle görelim:

```
IF (a = b) AND (b = c) THEN
      WRITELN('Eşkenar üçgen')
   ELSE
      IF (a = b) OR (a = c) OR (b = c)
         THEN
         WRITELN('İkizkenar üçgen')
   ELSE
   WRITELN ('Çeşitkenar üçgen');
```

Örneğimizde a, b, c değişkenleri üçgenin birer kenarlarını temsil etmekteler. Dikkat ederseniz ilk önce eşkenar olma şartına baktık, çünkü her eşkenar üçgen aynı zamanda ikizkenar üçgen şartını da sağlar. Eğer ilk önce ikizkenar şartına bakarsak hiçbir zaman eşkenar üçgenleri bulamazdık. Çünkü eşkenar üçgenleri de ikizkenar olarak kabul etmiş olurduk.

İÇ İÇE IF KOMUTLARI

IF komutlarını iç içe kullanmamız mümkündür:

```
IF b1 THEN
   IF b2 THEN
      k3
   ELSE
      k4
ELSE
   k2
```

İlk önce b1 Boolean İfadesi'ne bakılır, eğer bunun değeri FALSE olursa k2 komutu işlenir, b1 ifadesinin değeri TRUE olursa THEN'den sonra IF komutu işlenir. Burada b2'nin değeri TRUE ise k3 komutu işlenir, eğer b2'nin değeri FALSE olursa k4 komutu işlenir. Örneğimizin doğruluk tablosunu yapalım:

b1	b2	İşlenen komut
FALSE	FALSE	k2
FALSE	TRUE	k2
TRUE	FALSE	k4
TRUE	TRUE	k3

İkinci bir örnek daha görelim:

```
IF b1 THEN
   k1
ELSE
   IF b2 THEN
      k3
   ELSE
      k4
```

Bu örneğimizin doğruluk tablosunu yapalım:

b1	b2	İşlenen komut
FALSE	FALSE	k4
FALSE	FALSE	k3
TRUE	FALSE	k1
TRUE	TRUE	k1

If komutları ile ilgili örnek programımız için program döküm ekine bakın. Şimdilik hoşçakalın... □



OSAM BİLGİSAYAR

SÜPER OYUNLAR SERİSİ

OSAM

SÜPER OYUNLAR PAKETİNDEN

9 PAKETİ BİRDEN ALANLARA

"LIGHT PEN"

HEDİYE EDİYOR

PAKET PROGRAMLARI	5500 TL.
1 KASETTE 20 OYUN	2200 TL.
1 KASETTE 40 OYUN	3000 TL.
1 KASETTE 1 OYUN	1500 TL.
COMMODORE T-SHIRT	1500 TL.
LIGHT PEN	10.000 TL.

1986 YILININ SÜPER PAKET PROGRAMLARININ ÜRETİMİ DEVAM ETMEKTEDİR.

10 SÜPER PAKETTEN BEŞ PAKETİ TOPTAN ALANA % 20 İNDİRİM UYGULANMAKTADIR.

Süper Paket Programlar serisine devam

PAKET 8

KAFA AYARI PRG.
YIE AR KUNG-FU II.
STRIKE COBRA
TARZAN
AVENGER
FAKLAN'S 82
BASILDON BOND
NEMESİS
WARHAWK II.
PARKPATROL
BURGER TIME
CRACKER REVENGE I
CRACKER REVENGE II
WARIOK
PAPERBOY
FORCE
GREMLİNS
CASİNO ROYALE
GANDALF SORCERER
GALİLAN
STAR PİNG-PONG

PAKET 9

KAFA AYARI PRG.
FOOTBOL OF YEAR
STRET SURFER
SOLDIER
SABRE WULF
BUCK BORGERS
F-FOX
BREAKTHRU
DANTES INFERNO
GREAT PACMAN
CLUMSY COLINS
TWO ON TWO
BAZOOKA BILL
CAPTURED
ARAC
WILL O'THE WIPS
CUTHBERT TOMBS
ORANGE SQUASH
BOZOS NIGHT
HOODOO VOODOO
BREAK FEVER

PAKET 10

KAFA AYARI PRG.
HE-MAN
HE-MAN (ADVENTURE)
STALLONE COBRA
SHAD' LIN'S ROAD
MUTANTS
SOLDİER ONE
IT'S KNOCK OUT
THE YOUNG ONES
SUPER CROSS
MR DO'S CASTLE
FLASH BEEAR
RALLY SPEDWAY (2)
GHOST CHASER
MAD NURSE
FREAK FACTORY
ANTRIAD GAME
LEGENT OF KAGE
SOLDIER FORTUNE
TENTH FRAME
WILLIAM WOBLER

PAKET 11

KAFA AYARI PRG.
SUPER TEST
RED MAX
FLASH GORDON
JOHN REB II
FIRE LORD
PSYCASTRIA
KETTLE
SPELUNCER
MELON MANIA
BISMARC
MAGIC MADNESS
ARD NINJA
KUNUCKLE BUSTERS
CAPTAN KELLY
MUSIC CONSTR SET.
FAIRLIGHT NIWA
EINDELOS
SPACE HUNTER
SKI RACE
COMMANDO LIBYA

PAKET 12

KAFA AYAR RPG
TOP-GUN
FLIGHT 747 BOOT
PIACY
SPY VS SPY IV
TAU CETİ
FUTURE KNIGHT
SKY RUNNER
PRODIGY
KARA KOUT
ENTER THE NİNSA
EQUINOX SPECIAL
HOPELESS
INHERITANÇE
STAR TROOPER
KNUCKER HOLE
TOM
MADONNA
GRAND LARCENY
GILIGIAN'S GOLD
DUMMY RUN

BAYİLİKLER VERİLECEKTİR

Ebuzziya Caddesi Sakızlı Yalı Sokak No.3 Bakırköy-İstanbul Tel: 583 72 46

Makine Diline Merhaba (7)

AYHAN KALAYLIOĞLU

5. ASSEMBLER

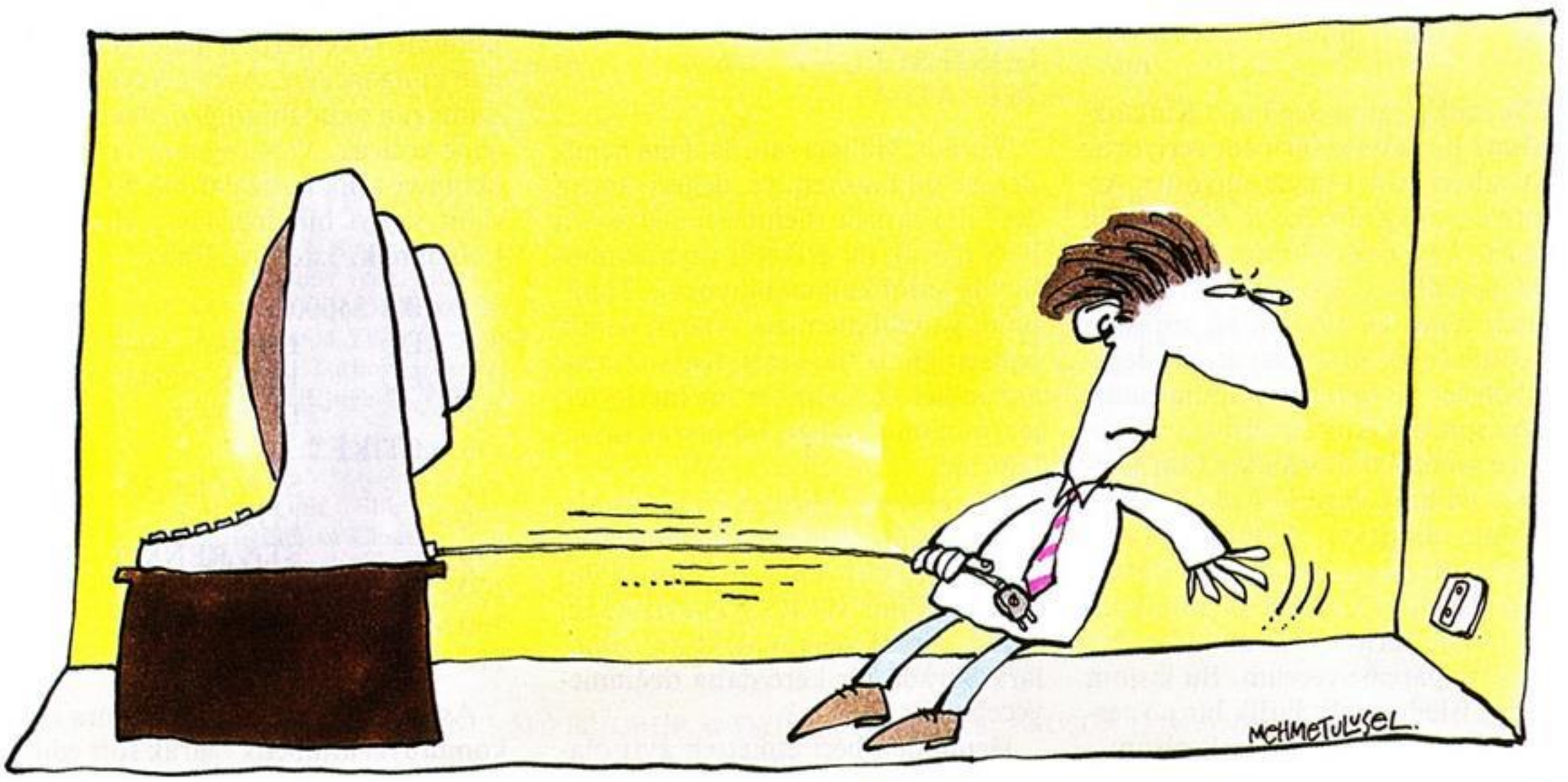
Assembler programı, makine dilinde yazacağımız programları, BASIC'te yazıyormuş rahatlığıyla girmemizi sağlar. Aynen, BASIC'teki gibi satırları listeler, ekranda değiştirebilir, satır ekleyip çıkarabiliriz. Her bir satırın kendisine ait satır numarası vardır. Bu numaranın arkasından assembler komutları gelir. Daha doğrusu, bir satır sırasıyla, satır numarası, bir eksi '-' işareti, bir etiket 'LABEL', komut ve operandını, daha sonra da o satıra ait açıklamaları (noktalı virgülle ayrılmış olarak) içerir. Tuna'nın, Süper Assembler'de de bahsettiği gibi bu açıklama kısmını kullanmayı ihmal etmeyin. Yoksa, birkaç ay sonra kendi programınızdan bir şey anlayamazsınız. Burada kullanılan açıklamalar, üretilen kodu hiçbir türlü etkilemiyor ve sadece dökümantasyon amaçlarıyla kullanalıyor. Şimdi bir örnek satır görelim:

```
100 -TEXT LDA $70,X;
BAŞLANGIÇ DEĞERİNİ ALMAK
```

Şu anda yazmakta olduğunuz programa, 'source' veya kaynak programı denir. Bu programları diğer programlardan kolayca ayırdedebilmek için isimlerinin sonlarına ".SRC" ekleminizi tavsiye ederiz. Assembler'ın, bu kaynak programından ürettiği obje kodunun isiminin sonuna da ".OBJ" eklerseniz, çok genel bir konvansiyona uymuş olursunuz.

Diyelim ki kaynak programınızı tümüyle yazdınız. Şimdi, RUN komutunu kullanarak assembler programının, obje kodunu üretmesini sağlayın. Kaynak programınız içinde hiçbir şey belirtmediyseniz, obje kodu bellekte üretilecektir. Eğer '.OB' komutunu kullandıysanız, obje kodunuz sadece disket sürücüsüne belirttiğiniz ad altında gidecektir. Obje kodu üretilirken istenirse bir de assembler listing'i alınabilir. Bu listing, istenirse ekrana, istenirse bir dosyaya, istenirse yazıcıya çıkarılabilir. Eğer tüm 'assembly' işlemi hatasız olarak biterse, assembler üretilen obje kodunun ilk ve son byte'larının adreslerini ve tüm işlemin ne kadar zaman sürdüğünü anlatan mesajlar verir. Arkasından 'READY.' mesajı ile direkt moda çıkarsınız. Şimdi isterseniz, bir de sembol tablosunu görebilirsiniz. Bunun için '/!' veya '/!!' komutlarını kullanabilirsiniz. Dilerseniz, sembol tablosunun, üretilen assembler listing'inin arkasından otomatik olarak listelenmesini sağlayabilirsiniz.

Bir önceki bölümde gördüğümüz, karakter setini ekranda çıkaran örnek bizi oldukça yormuştu. Bunun nedeni ise gereksiz yere uğraştıran hesaplama ve sayı sistemi çevirme işlemle-



riydi. Şimdi aynı programı, bir de assembler kullanarak yazalım. Bu arada, yeni bir olayla karşılaşacağız. Sözkonusu olay 'LABEL' yani etiket kullanımı. Tuna'nın Süper Assembler yazısını okuyanlara tanıdık gelecek bu terim bizi birçok sıkıcı işlemden kurtarıyor. Etiketler, program içindeki bazı yerlerin adresini assembler programına hesaplatmamızı sağlıyor. Diyelim ki, biz programımızı sabit adresler kullanarak yazdık. Araya bir bölge ekleyeceğimiz zaman programın birçok bölümü ileri veya geri kayacak ve sonuç olarak adresi değişecek, biz ise programın kendi içinde bu adresleri kullanan bütün bölgeleri bulup elden değiştirmek zorunda kalacağız. Fakat, adresleri önemli yerlerin başına birer etiket koyup, programın diğer kısımlarından buraya yapılacak başvuruların bir etiket üzerinden yapılmasını sağlarsak programımız artık her türlü değişikliğe karşı genel bir yapıya sahip olacaktır ve bizi daha az uğraştıracaktır. Tabiî, yukarıda söz ettiğimiz değişikliklere, program içinde başvurulan bir yerin tümüyle silinmesi dahil değil! Sembol ve etiketleri de kullanarak yazacağımız kaynak programımız şöyle görünecektir:




```
10  -          .BA $6000
100 -          LDX # 0
110 - ETİKET   TXA
120 -          STA $0400,X
130 -          LDA # 1
140 -          STA $D900,X
150 -          INX
160 -          BNE ETİKET
170 -          RTS
```

Ne kadar rahat değil mi? Kullanacağımz bir adrese bir isim veriyoruz ve o adresi artık ismiyle anıyoruz. Assembler, programı iki kerede gözden geçirir (Two pass assembler = İki geçişli assembler). İlk geçişte, tüm etiketleri bulur ve bunlara ait adresleri (o sıradaki program sayacının değeri) belirler ve bunları aklında tutar. Daha sonraki ikinci geçişinde ise etiket ve sembol tablosundan tüm sembolize edilmiş değerleri alır ve obje kodunu üretirken kullanır. Yani, böylece bir programın içinde ilerdeki bir etiketi de kullanmak mümkündür. Buna örnek olarak ufak bir program parçası verelim. Bu kısmın yaptığı işlem onaltı bitlik bir göstergecin değerini bir artırmak olsun:

```
.
.
.
230 -          INC GÖSTERGEÇ
240 -          BNE ATLA
250            -          INC
GÖSTERGEÇ+1
260 - ATLA
```

Burada 240'ıncı satırda daha henüz değeri bilinmeyen ve değeri ancak 260'ıncı satırın işlenmesinden sonra belli olacak bir etiketin veya sembolün değerini kullanabiliyoruz. Tabiî, bunu yapabilmemizin sebebi, demin bahsettiğimiz 'iki geçiş' olayı. Zaten, assembler ve compiler'lar (derleyiciler) minimum iki geçişli olarak tasarlanırlar.

Baştan beri, sözünü ettiğimiz sistemli çalışmaya bir örnek daha verip bu bölümü bitirmek istiyorum. Çünkü, Tuna'nın MART 87 sayısındaki yazısı, birçok detayı açıklıyor. Bunlara burada bir kere daha değinmeyeceğim.

Deminden beri etiketten ayrı olarak sembol diye bir şeylerden söz edip durduk. Bunlar gene etiket gibi, assembler'in kullandığı kendine has değişken tiplerinden biri. Biz burada sembolleri, programımızın kaynak kodunu iyice serbest hale getirmek için kullanacağız. Ayrıca kaynak kodumuzun okunabilirliği de önemli ölçüde artacak. Yapacağımız tek işlem, ekran ve renk hafızalarının başı olan sabit sayıyı bir değişkene alıp öyle kullanmak. İşte örneğimiz:

```
10  -.BA $6000
80  -.EQ       EKRAN = 0400
90  -.EQ       RENK = D800
100 -          LDX# 0
110 - ETİKET   TXA
120 -          STA EKRAN,X
130 -          LDA#1
140 -          STA RENK,X
150 -          INX
160 -          BNE ETİKET
170 -          RTS
```

Assembly işlemi bittikten sonra /!! komutuyla alfabetik olarak sort edil-

miş sembol listesini istediğimizle şununla karşılaşırız:

```
EKRAN    =0400
ETİKET   =$6002
RENK     =$D800
```

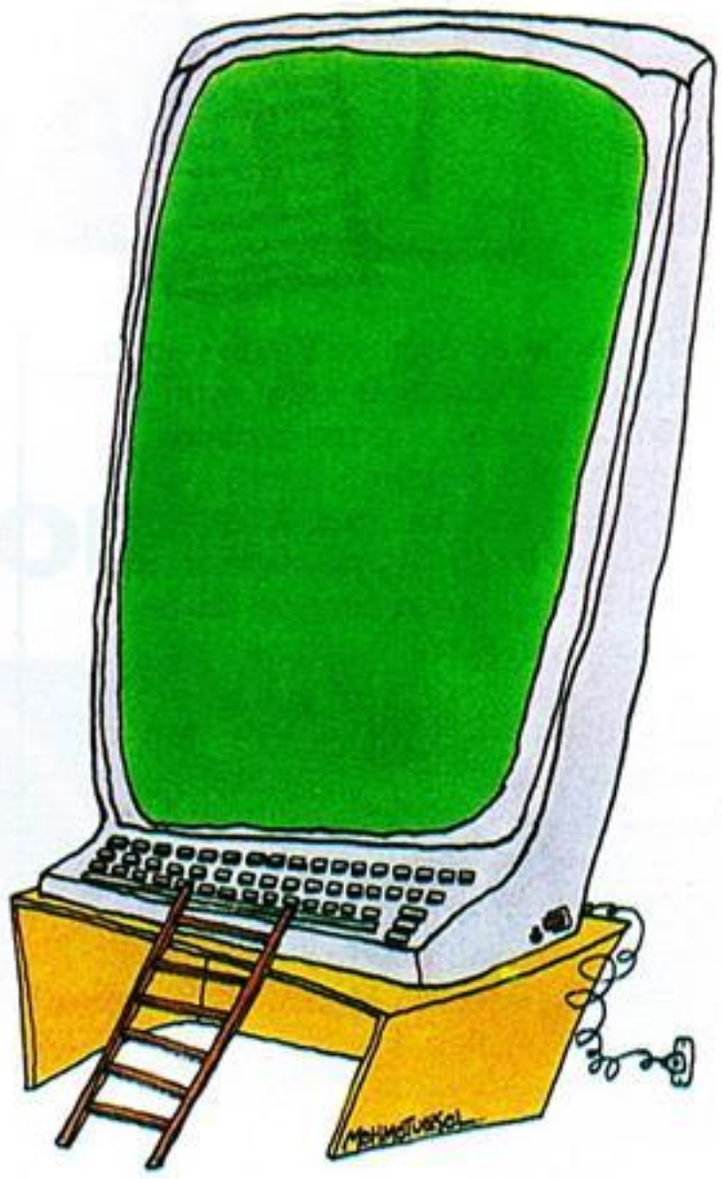
Amaçladığımız şekilde kaynak programımızın okunurluğu ve ilk bakışta anlaşılırlığı büyük ölçüde artmıştır. Artık ekran veya renk ile ilgili adresler kaynak programının birer parametresidir. Diyelim ki ekran hafızasını $2000'a aldınız, (Bunun nasıl yapılacağını, Tuna Ertemalp arkadaşımızın 'COMMODORE 64 ile Grafik' yazısından öğrenebilirsiniz.) karakter setini ekrana çıkarmak için yazacağınız makine dili programının genel görünüşü olduğu gibi yukarıdakiyle aynıdır. Tek yapacağınız:

```
80  -.EQ      EKRAN=0400
```

satırı yerine;

```
80  -EQ       EKRAN$2000
```

satırını koymanız. Tabii, büyük ihtimalle programınız normalde bu kadar kısa olmayacak ve her sembol birden çok yerde kullanılacak. Böyle bir durumda ise, sembol kullanmanın avantajları daha iyi anlaşılacak.



Pratikte, sembol tablosu en çok sistem değişkenlerini ve sistemde yeralan altprogramları adlarıyla çağırabilmeniz için kullanılıyor. Buna en iyi örnek olarak işletim sisteminin KERNAL tablosunu verebiliriz. Bu tablo-da, işletim sisteminden talep edebileceğiniz birçok işlemin (dosya açma, kapama, dosya yükleme, saklama, kullanıcıdan bilgi girişi veya çıkışı gibi) adları ve ROM'da hangi adresten çağırabileceği belirtilmiştir. İleride, sırası gelince bu tabloya da değineceğiz. Sonuç olarak vurgulamak istediğimiz şey, makine dili program yazarken sistemli çalışma prensibini hiçbir zaman elden bırakmamanız.

Assembler kullanımı ile ilgili anlatılması gereken birkaç olay daha var. Mesela makro'lar, rekursif makro'lar, pasudo-op'lar, düşük değerli ve yüksek değerli byte'ların assembler'a hesaplattırılması ' ' ve ' ' operandları) vs. Fakat, Tuna arkadaşımız, verdiği Süper-Assembler adlı yazıda, bütün bunları oldukça detaylı bir şekilde açıklamış ve anlatmış. Ben bunları burada bir kere daha tekrarlama gereksinimi duymuyorum.

Yazımızın, bir sonraki bölümünde beraberce ufak bir grafik yardımcı programını tasarlayıp gerçekleştireceğiz. Bu işlemi yaparken de, yeni programlama tekniklerini deneyip assembler'ımızı kullanacağız. Şimdilik hoşçakalın... □

UFO BİLGİSAYAR

Orhan Altuncu
İstasyon Cad. Pekin Psj. 2.Kat No.35
Akbank Karşısı 572 13 17/Bakırköy/İST.

EN KALİTELİ-EN UCUZ-EN YENİ HARDWARE & SOFTWARE

- FREEZE FRAME MK3.B: 28.000
  En güçlü Copy
  Ek fonksiyonlar, Türkçe broşür
- ICE MACHINE TRISTAR: 27.000
  Türkçe Broşür
  Dünyada Bantlı yükleyen ilk ICE.
- UFO CB: Kilit Dinlemez, TapeTo 12.000
  Tape Copy veya 2 Master Copy
- SAUND CAPSUL: Prg. Başlangıcını 1.500
  bulur, yüklemeyi kolaylaştırır
- EXPERT.9: Tek parça güçlü copy 30.000
  Prg.da değişiklik imkânı, Disk veya
  kaset programı ve broşür.
- EPROM Programlayıcı: Broşürle 30.000
- UFO KA: Rakipsiz Sistem 11.000
  İbreli Kafa Ayar Cihazı
- Kaset ve Diske Özel Çekim 500
- Kasette 1 Program 1.000

TÜRKİYE'DE İLKLER (Başkaları Tarafından Dağıtımı Yasaktır)

- UFO CR-2: Super Teyp Copy Prg. 7.000
  Master Copy verir. Gücü % 65
- UFO HR turbo: Hi-res Grafikle 25.000
  Ve Reklamla Yükleme İmkânı
- UFO GM Turbo: Hi-res Grafikle 100.000
  ve Müzik Eşliğinde Yükleme.
- UFOGRAF: Çok özel oyun doldurma 1.000
- UFO TD-1: Tape To Disc, Gerçek 7.500
  Blokunda tek parça. Gücü % 65
- UFO HR-2 Turbo: Daha Pratik 50.000
  ve Güçlü, Grf. ve Reklamlı yük.
  dilerseniz 246 blok kapasite

UFO CLUB

1- Ödediğiniz Yıllık Ücreti Geri Alacağınız Sistem, Ayrıca Bedava Prg.lar. 8.000
2- Program ve Cihazlarda % 10 indirim. Her ay ücretsiz UFO NEWS
3- UFOGRAF YÜKLEME: Özel Grafik ve isminizle yüklemede % 20 indirim

- ENTEGRE İŞ PROGRAMI (Cari Hesap-Faturalama ve Adres Programı 50.000
- İSTİHKAK PROGRAMI 30.000
- STOK PROGRAMI 20.000
- ADRESLEME PRG. 25.000
- BRUCKNER HESAPLARI (Alan-Hacim) 40.000

• YEPYENİ 3 SÜPER GRUP 20 OYUN... 5.000'er TL. BROŞÜR İSTEYİNİZ.

NOT:
Sipariş bedeline disk ve cihaz için 2.000 TL., kaset için 800 TL. ilave ederek ZİRAAT BANKASI 300/123 Ataköy hesabına yatırınız. Makbuz fotokopisini gönderirseniz, siparişiniz elinize daha çabuk geçecektir.

KÜÇÜK İLANLAR

SATILIK C-16

MURAT KONCAVAR
Tel: 163 19 44

C-64'ü olan arkadaşlarla her türlü program değiş tokuş etmek istiyorum.

UĞUR SUNGUR
Türk Pirelli Lojmanları Köseköy-İzmit,
Tel: 12 110'dan 128.

SATILIK Casio CK-500 Electronic Musical Instrument Stereo Double Casette Recorder Dubbing Kayıt Sistemi. SW-AM-FM-Dalgalı Radyosu, Master-Ton-Effect Controlleri, 12 Ritim ve 12 Ton Fingered Özelliği ve Fill'in Tuşu, Casio Chord System. Accompaniment ve Melody Keyboard. Bu cihazımı 1541 Disk Driver ve MPS-802 DOT MATRIX Printer ile değiştirmek istiyorum.

CÜNEYT KURDOĞLU
Tel: 166 55 51, İstanbul.

1541, C-64, 2 joystick, birkaç oyun satılıktır. 500.000 TL.

ORHAN PEMBEYÜZ
Tel: 368 31 51.

SATILIK C-64, Sam, Datasette, Pascal Simon's Basıc Ex Basic ve 100 oyun dahil.

İLKER KINACI
İzmir Tel: 313542.
İstanbul Adres: Abbasağa Kuyu Sk. No:28 D.4 Beşiktaş-İstanbul.

SATILIK (Commodore 64 veya SX-64 için) Magic Voice Kartuşu Orijinal ambalajında, kullanım kitapçığıyla birlikte 35.000.- TL.

METE KESEN
Tel: 26 26 57-Ankara

87'NİN en son oyunları dahil zengin oyun çeşitleri, ayrıca kasette Simon's Basic, Monitör Prog., Kopya Prog. vs. çok ucuza satabilir veya değişebilirim. Ayrıntılı bilgi ve liste isteyiniz. Oyun alanları sürprizler bekliyor.

Adres: Eski Londra Asfaltı, Villa Ağma 1 Ap. No:48/4 Bahçelievler/İstanbul
Tel: 575 75 47.

ATARI 802 XL teyp ünitesi, 50 oyun, 2 joystick satılıktır. 230.000.- TL.

GÖKHAN BACARLI
Tel: 337 79 14.

BİR kasette 80 oyun ve Simon's Basic 6000.- TL. Bir kasette 5 süper oyun 3000.- TL. (Kaset ve posta giderleri bana ait olmak üzere). Orta 3 Matematik kasetini 2500 TL.'ya satmaktayım. 500'ü aşkın oyunum var. Bunları hem kasette hem de diskte satmaktayım (fiyat listesini isteyiniz).

SELADATA Y. ORHAN
Keçeciler Cad. No: 91/2
Mimar Sinan Mh. Karagümrük/İstanbul

COMMODORE için turboya çevrilmiş Simon's Basic 1, Simon's Basic 2, Graphic Basic, Data Doctor, 10 oyun 10.000.- TL. Ayrıca Amstrad için geniş kapasiteli vergi iadesi programı satmak istiyorum.

YILDIRIM ILICAK
Meriç Tekstil A.Ş. Edirne.

SATILIK Atari 800 XL+Teyp+Joystick+4 oyun 200.000 TL. (16.00-17.00 arası) arayabilirsiniz.)

ÖNER EKEN
Tel: 678809

ARKADAŞLARLA oyun değiştirmek ve yazışmak istiyoruz. Ayrıca çok ucuza oyun satıyorum. İsteğe bağlı olarak eğitim ve iş programları yazılır.

MERT ve GAMZE YILMAZ
Cumhuriyet Cad. No:27/1
Kastamonu. Tel: 1353.

COMMODORE kullanıcıları için 3 ayrı kasette yararlı 3 program.
1) Alman MDE ve BDE'si, Kafa Ayar programları,
2) Locky Jr., Unlocky Jr., Copy 190,
3) Spectrum sim., Monitör Prg. Atari 800 XL sim.
Kaset dahil her biri 3500.- TL. Posta giderleri bana ait olmak üzere ödemeli gönderilir.

BURAK YÖRÜGER
Karasoku Mh. 96 Sk. No:14/C Adana.

AZ kullanılmış ZX Spectrum+bilgisayar, adaptör, TV teyp bağlantı kabloları, orijinal kullanım kitabı, orijinal kasetler (oyun) Tepum'un "Herkes İçin Basic" kitapları ZX 80 ve Basic programlama kitapları, çoğu 86/87 olmak üzere oyun ve eğitim kasetleri hepsi birlikte satılıktır.

ZAFER KARABACAK
Adres: MSB Yük. Öğr. Erkek Öğrenci Yurdu, Sarayburnu/İstanbul.
Tel: 526 27 05.

SAM+Sam's Demo+Simon's Basic+MDE+Koala Painter+İki tür monitör programı+G-Pascal+Forth+Copy 190 ve 220+Esraturbo+Turbo V-3 ve TR-06, 3 çeşit adventure oyun ve 5 çeşit aracade oyun+kafa ayar programı kaset dahil 7500.- TL.

ERKAN BALCI
İvedik Cad. 43. Sk. No:24/11
Demetevler/Ankara Tel: 345 94 87.

COMMODORE'cular yazışalım, bilgilerimizi paylaşalım. Ayrıca her oyun tanesi 500 TL'dan satılık (isteyiniz).

BURÇ ÖZER
Kabataş Erkek Lisesi
2 Fen L Sınıfı
Ortaköy-İstanbul
Tel: 572 98 82 - 571 96 17

1-CBM-64 ve VIC-20 için: İkili kopya cihazımı satıyorum, her tür kopya çekme imkânı. 1000 TL.
2-CBM-64 için: Süper grafik çizim-boyama-doldurma programı Koala Painter ve diğer Utility programları Simon's Basic I, II Sam ve 5 Salt programlarımı satıyorum.

CENGİZ KÜPÇÜ
Cumhuriyet Mah.
Bin Evler Blok E4 No: 7
22030 Edirne

SATILIK 1986'nın en son oyunları. Oyun değişmek isteyenler listeleriyle adresime başvursunlar.

CENGİZ ALİOĞLU
Karadeniz Cad. No: 67/2
Fatih-İstanbul

ELİMDE bulunan 50 oyunu takas etmek istiyorum. Elimdeki oyunların bazıları şöyle:
1- Zorro
2- İnt. Basketbol
3- The Human Race
4- Soccer II
Liste isteyiniz.

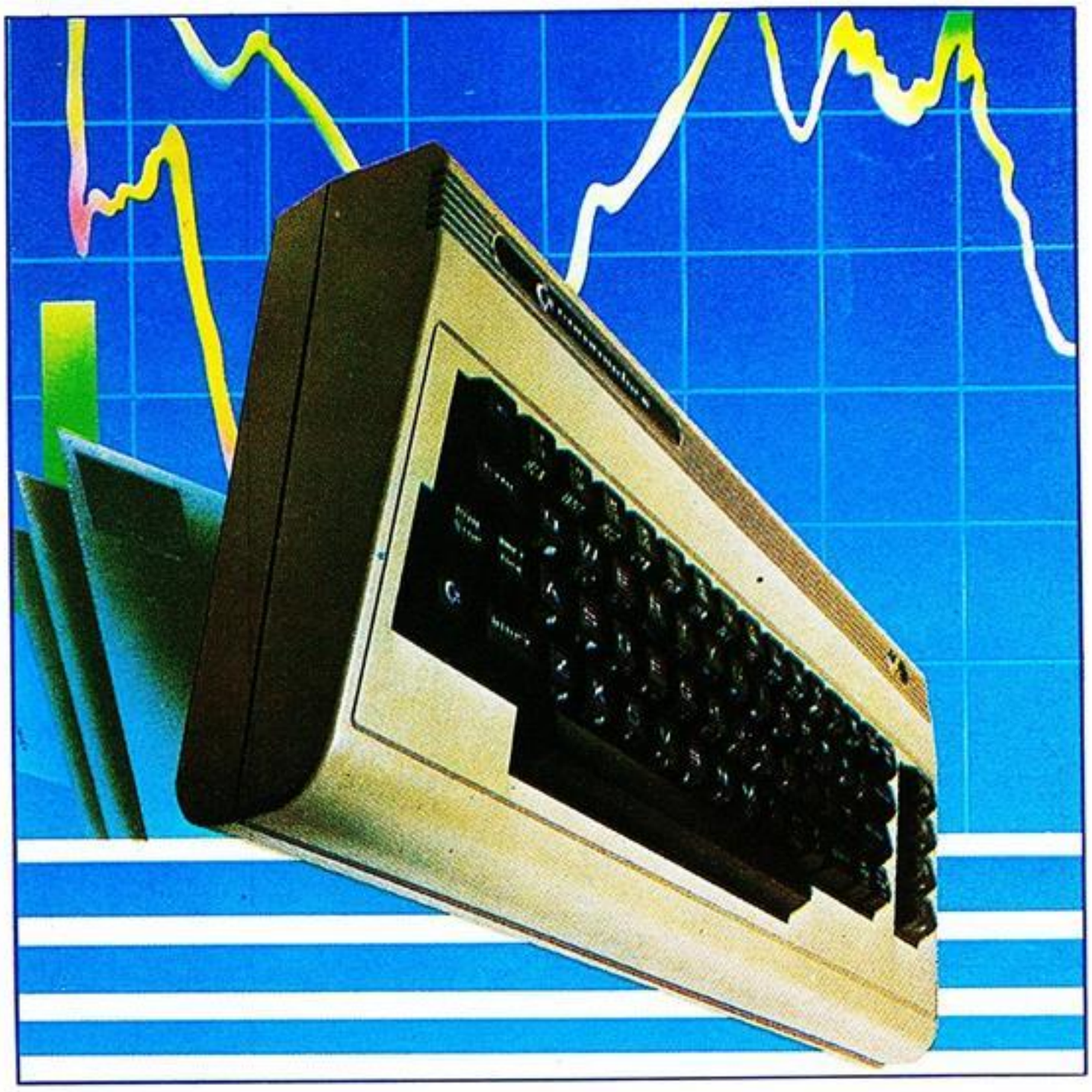
KAAN AKKAN
Üsküdar Cad. Eren Apt. No: 58
Daire: 16 Kartal-İstanbul

AZ kullanılmış, temiz 1 yıl garantili Toshiba RT-SX4 auto reverse sistemli ve ekolayzerli mini setimi C-64 veya Disc Driver ile değiştireceğim.
Not: Talepler çok olursa tercihli seçim yapılır.

ARİF BAYAT
Demirkapı Sk. No: 13/9
Cebeci-Ankara.

2.BASKI

Commodore 64



PROGRAMCININ EL KİTABI

Commodore COMPUTER

BAYİLERDE



Amiga günü.

Gün, Amigaların günü! Çünkü uzun zamandır yalnızca hayal edilebilen olanakları, nihayet, Amiga sunuyor!
Amiga'nın, duyanları olağanüstü heyecanlandıran eşsiz yeteneklerinden bazıları, huzurlarınızda!

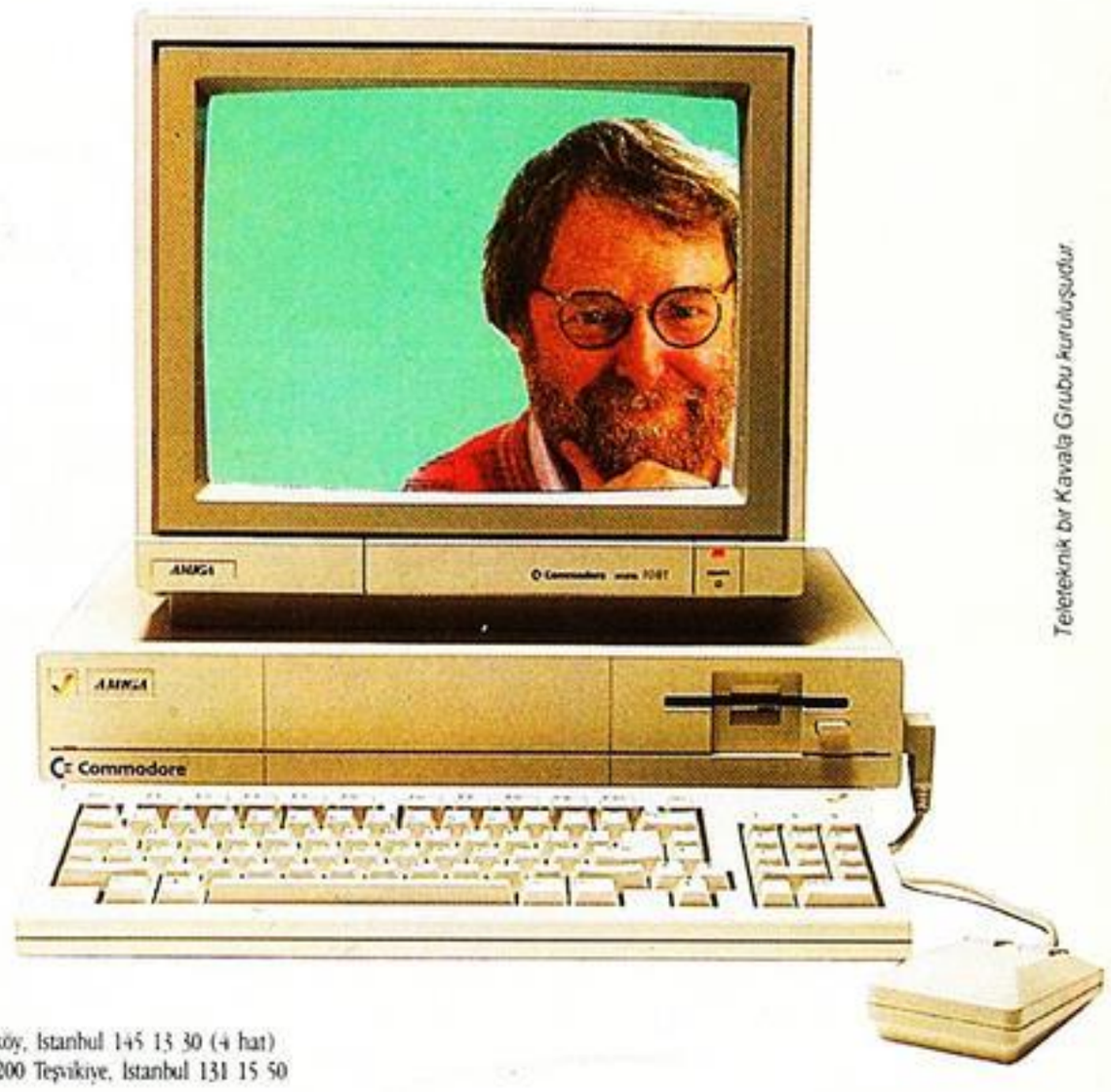
● Durağan görüntüleri hareketli görüntülere Amiga ile dönüştürebiliyorsunuz. ● Banttan ya da kameradan doğrudan görüntü alabiliyor; monitördeki görüntüyü, video kasete kaydedebiliyorsunuz. ● 4 ayrı sesi stereo olarak üretebiliyorsunuz.● İnsan sesi üretebiliyorsunuz.● Sayısız müzik aletini Amiga ile çalabiliyor, dilerseniz orkestra bile kurabiliyorsunuz.● Elde ettiğiniz görüntü ile senkron, müzik yapabiliyorsunuz.● Amiga'daki renk seçeneği sayısı, tam 4096! ● Amiga'da, görüntüleri renkli fotoğraf kalitesinde saklamak mümkün.● Muazzam bellek, 32 Bit hız, Amiga'da! ● Multi tasking işletim sistemi sayesinde, aynı anda çok sayıda programda birden çalışabilme olanağını, size, Amiga sunuyor.

Grafikerler! Mimarlar! Mühendisler! Doktorlar! Tasarımcılar! Ve süper bir bilgisayardan yararlanabilecek herkes!
Gün, Amigaların günü! Yıl, Commodore'un yılı! Yararlanın!

AMIGA

Profesyonellerin Aradığı Üstünlükler
Commodore Amiga'da Standard

Merkezi İşlem Birimi	: MC 68000 (8 MHZ, 16/32 Bit)
Yardımcı İşlem Birimleri	: Custom Animation Custom Graphics Custom Sound/Peripherals
Ana Bellek	: 512 KB Standart (8,5 MB'a Kadar Arttırma İmkanı)
Ara Birimler	: · Mouse, Grafik Tablet, Lightpen, Joystick ve Paddle için 2 Bağlantı Yuvası · Bellek arttırma, harddisk bağlantısı için genişleme yuvası · Programlanabilir centronics paralel port · RS232C (V24) seri port · Stereo ses çıkışı · RGB analog ve digital çıkışlar · 3 adet 3,5 veya 5,25 inch disket sürücü bağlantısı için disket sürücü çıkışı
Klavye	: Düşük yükseklik, eğimi ayarlanabilir, bağımsız 10 fonksiyon, HELP, 4 cursor hareket ve 13 nümerik tuş olmak üzere toplam 89 tuşa sahiptir.
Ekran	: · Renkli grafik 60 veya 80 kolon · 16 renk 640 x 400 yüksek çözümleme gücü · 4096 renk paleti · 8 adet 16 bit genişliğinde yaratık
Yazılım	: · Amiga DOS Multitasking UNIX benzeri, Mouse veya Keyboard kontrolu · Amiga Basic · Amiga Tutorial · Kaleidoscope · Voice Synthesis Library
Ek Birimler	: · Dotmatrix, LQ, renkli bilgiyazarlar · Commodore PC uyumlu ek donanım Sidecar · 10/20 MB harddisk · Video digitizer-Frame grabber, Genlock · MİDİ interface



Teleteknik bir Kavala Grubu kuruluşudur

Teleteknik

Merkez: Rıhtım Cad. Nesli Han 207/4, 80030 Karaköy, İstanbul 145 13 30 (4 hat)
Bilgisayar Danışma Merkezi: Silahhane Cad. 59 80200 Teşvikiye, İstanbul 131 15 50