



7. HAFTA

TBT182

TEMEL BİLGİSAYAR BİLİMLERİ

Yrd. Doç. Dr. Şafak BAYIR

safakbayir@karabuk.edu.tr

KBÜ-UZEM

Karabük Üniversitesi

Uzaktan Eğitim Uygulama ve Araştırma Merkezi

7. Haftanın Konuları (İçerik)

Bilgisayar Donanımı/Donanım

Donanım (Hardware): Donanım "... bir sistemin çalışması, işlemesi için gerekli olan fiziksel araçlar; (program, vb. karşıtı olarak) bilgisayarı oluşturan makineler ..." olarak tanımlanmıştır (Longman-Metro, 1993: 678). Diğer bir ifade ile donanım veya bilgisayar donanımı, bir bilgisayar sisteminin fiziksel öğeleri anlamına gelmektedir (Hardware - Wikipedia, 2011).

Bilgisayar Donanımı: Bilgisayar donanımı, bir bilgisayarı oluşturan fiziksel parçaların genel adıdır ve bu parçalar, kişisel bilgisayarlar, otomobiller, çamaşır makinesi ve benzeri elektrikli ev eşyaları veya çeşitli sanayi u[y]gulamaları gibi birçok alanda kullanılır (Bilgisayar donanımı - Vikipedi, 2011).

Vikipedi'de (Bilgisayar donanımı - Vikipedi, 2011) donanımın "iç" ve "dış" olarak ikiye ayrıldığı belirtilmektedir ve şu şekilde sınıflandırılmaktadır; Bir donanım parçası başka bir donanım parçası (genellikle bilgisayar kasası) içine yerleştiriliyorsa **iç donanım** olarak adlandırılır. Bağımsız kasa, kutu veya kılıf içinde bulunan bilgisayar kasası içinde yer almayan donanımlara **dış donanım** denir. Bunlar

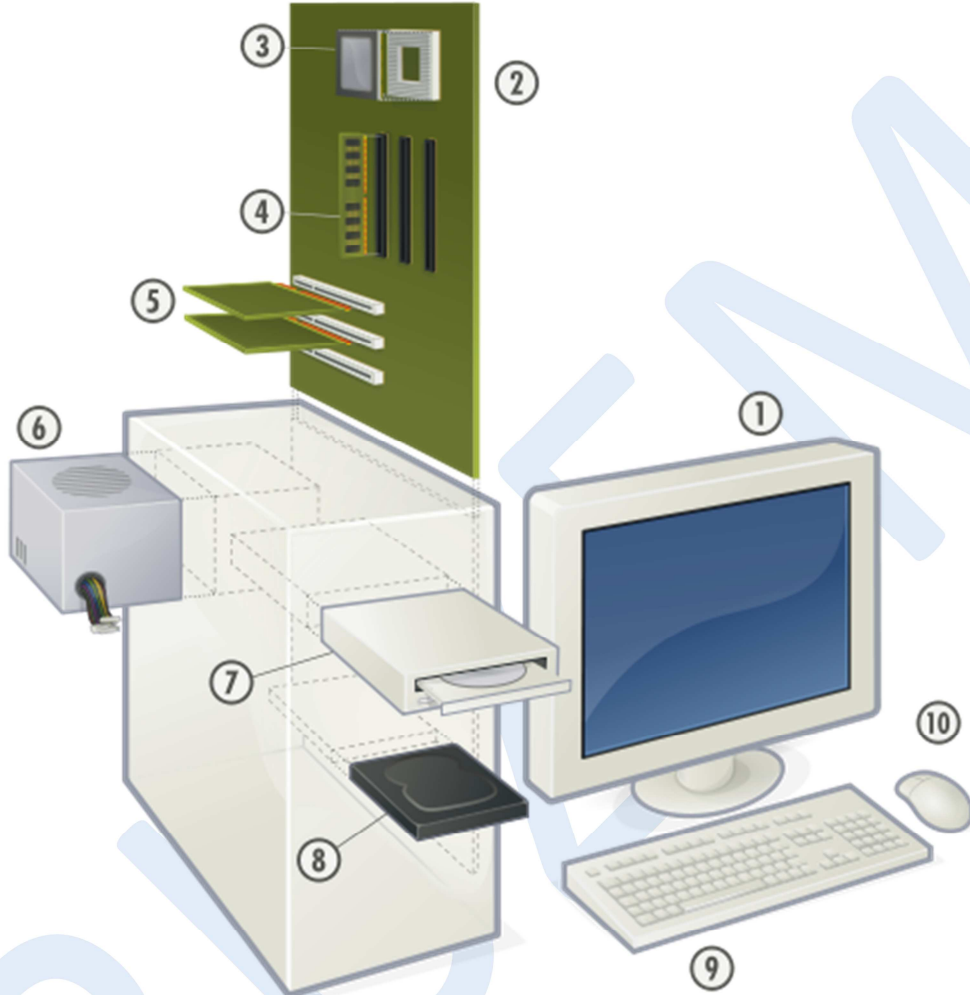
kamera, usb giriři, bluetooth, kızılötesi, tarayıcı, yazıcı v.s. (Bilgisayar donanımı - Vikipedi, 2011).

Kişisel Bilgisayar Donanım Öğeleri

Aşağıda modern bir Kişisel Bilgisayar'a (Personal Computer) ilişkin donanım öğelerine yer verilmektedir (Personal computer hardware - Wikipedia, 2011):

1. Monitör/Ekran (Monitor)
2. Ana kart (Motherboard)
3. Merkezi İşlem Birimi - Mikroişlemci/İşlemci (Central Processing Unit - CPU)
4. Rasgele Erişim Belleği/Bellek (Random Access Memory - RAM)
5. Genişletme kartları (Expansion cards)
6. Güç kaynağı (Power supply)
7. Optik disk sürücü(sü) (Optical disc drive)
8. Sabit disk sürücü(sü)/Sabit disk (Hard disk drive)
9. Klavye (Keyboard)
10. Fare (Mouse)

Kişisel Bilgisayar'a ilişkin donanım öğeleri Şekil 1'de gösterilmektedir (Personal computer hardware - Wikipedia, 2011):



Şekil 1. Kişisel Bilgisayar Donanım Öğeleri

Vikipedi’de (Bilgisayar donanımı - Vikipedi, 2011) veri akış yönüne göre donanımlar şu şekilde tanımlanmaktadır; Bir bilgisayar donanımı dış ortamdan bilgisayara veri aktarmak için kullanılıyorsa **giriş donanımı** veya **giriş birimi**, bilgisayardan dış ortama veri aktarmak için kullanılıyorsa **çıkış birimi** olarak adlandırılır.

Giriş ve Çıkış Çevresel Birimleri (Input and Output Peripherals)

Giriş ve çıkış aygıtları belirgin olarak sistem birimi veya bilgisayar kasasının dışında bulunmaktadır (Personal computer hardware - Wikipedia, 2011). Aşağıda birçok bilgisayar sisteminde standart veya genel olarak bulunan giriş ve çıkış aygıtları yer almaktadır (Personal computer hardware - Wikipedia, 2011):

Giriş Aygıtları (Input Devices)

- Metin giriş aygıtları (Text input devices)
 - Klavye
- İşaret(leme) aygıtları (Pointing devices)
 - Fare
 - Optik fare (Optical mouse)
 - İşaretleme topu (Trackball)
 - Dokunmatik/Duyarlı ekran (Touchscreen)
- Oyun oynama aygıtları (Gaming devices)
 - Oyun oynama kolu (Joystick)
 - Oyun oynama konsolu (Game pad)
 - Oyun yönetme kolu (Game controller)
- Görüntü, video giriş aygıtları (Image, video input devices)
 - Görüntü tarayıcısı (Image scanner)
 - Web kamerası (Web cam)
- Ses giriş aygıtları (Audio input devices)
 - Mikrofon (Microphone)

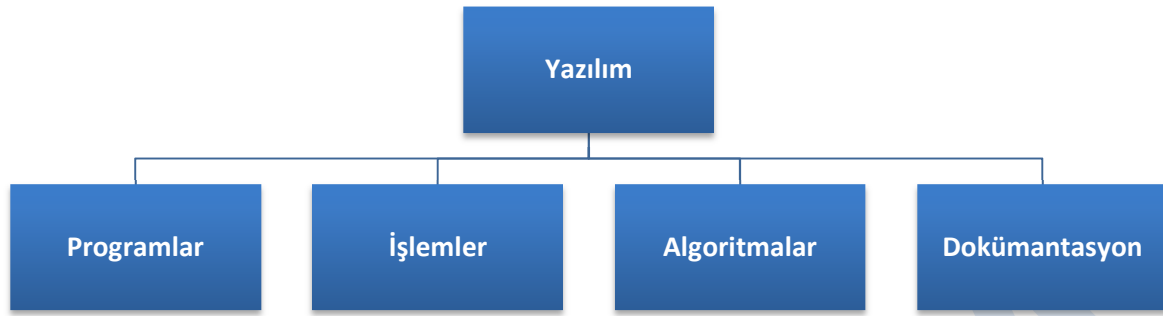
Çıkış Aygıtları (Output Devices)

- Yazıcı (Printer)
- Hoparlör(ler) (Speakers)
 - Kulaklık(lar) (Headphones)
- Monitör
 - Katod Işınlı Tüp (Cathode Ray Tube - CRT)
 - Sıvı Kristal Görüntü Birimi/Ekran (Liquid Crystal Display - LCD)
 - Işık Yayan Diyot (Light-Emitting Diode - LED)
 - Organik Işık Yayan Diyot (Organic Light-Emitting Diode - OLED)

Bilgisayar Yazılımı/Yazılım

Yazılım (Software): Yazılım "... bir bilgisayarın çalışmasını yönlendiren ve manyetik bantlara veya disklere kaydedilmiş olan komutlar, programlar ..." olarak tanımlanmıştır (Longman-Metro, 1993: 1449).

Bilgisayar Yazılımı: Bilgisayar yazılımı veya yazılım bilgisayar programlarının ve ilgili verilerin bir araya getirilerek bilgisayara ne yapacağını ve nasıl yapacağını bildiren yönergeler/bilgiler/direktifler topluluğudur (Computer software - Wikipedia, 2011). Diğer bir ifadeyle, bir yazılım programlar (programlar), işlemler (procedures), algoritmalar (algorithms) ve dokümantasyondan (documentation) oluşmaktadır (Computer software - Wikipedia, 2011) (Şekil 2).



Şekil 2. Yazılımı Oluşturan Öğeler

Vikipedi’de yazılıma ilişkin tanımlar şu şekildedir (Yazılım - Vikipedi, 2011):

Yazılım, değişik ve çeşitli görevler yapma amaçlı tasarlanmış elektronik araçların, birbirleriyle haberleşebilmesini ve uyumunu sağlayarak, görevlerini ya da kullanılabilirliklerini geliştirmeye yarayan makina komutlarıdır.

Yazılım, elektronik aygıtların belirli bir işi yapmasını sağlayan programların tümüne verilen isimdir.

Bir başka deyişle var olan bir problemi çözmek amacıyla bilgisayar dili kullanılarak oluşturulmuş anlamlı anlatımlar bütünüdür.

Yazılım Sistemleri

Yazılım sistemleri üç ana sınıfa ayrılmaktadır (Computer software - Wikipedia, 2011) (Şekil 3):

- **Sistem Yazılımları (System Software)**
 - Aygıt sürücüler (Device drivers)

- İşletim sistemleri (Operating systems)
- Sunucular (Servers)
- Araçlar (Utilities)
- Window sistemleri (Window systems)
- **Programlama Yazılımları (Programming Software)**
 - Derleyiciler (Compilers)
 - Hata ayıklayıcılar (Debuggers)
 - Yorumlayıcılar (Interpreters)
 - Bağlayıcılar (Linkers)
 - Metin editörleri (Text editors)

Tümleşik/Entegre geliştirme ortamı (Integrated development environment - IDE) olarak adlandırılan yazılımlar programlama yazılım araçlarının özelliklerini tek bir pakette toplayan yazılımlardır (Programming tool - Wikipedia, 2011).

- **Uygulama Yazılımları (Application Software)**
 - Endüstriyel otomasyon (Industrial automation)
 - Ticari işlere yönelik yazılımlar (Business software)
 - Video oyunları (Video games)
 - Kuantum kimya ve katı hal fizik yazılımları (quantum chemistry and solid state physics software)
 - Telekomünikasyon/İnternet (Telecommunications (Internet))
 - Veritabanları (Databases)
 - Eğitsel yazılımlar (Educational software)
 - Matematik yazılımları (Mathematical software)
 - Tıbbi yazılımlar (Medical software)
 - Moleküler modelleme yazılımları (Molecular modeling software)
 - Görüntü işleme (Image editing)
 - Elektronik tablo (Spreadsheet)
 - Simülasyon yazılımları (Simulation software)

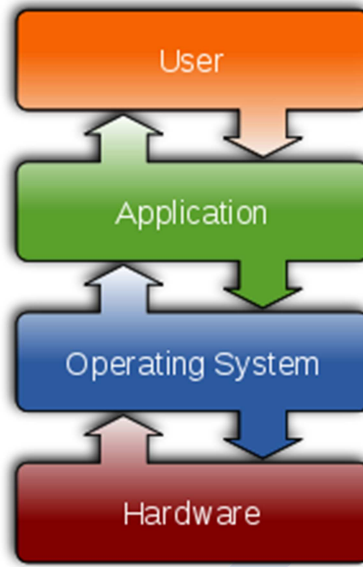
- Kelime işlem(e) (Word processing)
- Karar verme yazılımları (Decision making software)



Şekil 3. Yazılım Sistemleri

Donanım ve Yazılıma İlişkin Katman Yapısı

Kullanıcı (user), uygulama yazılımı (application software), işletim sistemi (operating system) ve donanım (hardware) arasındaki ilişki aşağıdaki gibi belirtilmiştir (Computer software - Wikipedia, 2011) (Şekil 4):



Şekil 4. Donanım ve Yazılıma İlişkin Katman Yapısı

Programlama Dilleri

Wikipedia’da programlama dillerine ilişkin aşağıdaki ifadeler yer almaktadır (Comparison of programming languages - Wikipedia, 2011):

Programlama dilleri bir makinenin (çoğu kez bir bilgisayar) işleyişini kontrol etmek için kullanılmaktadır. Programlama dillerinde doğal diller gibi sözdizimi (syntax) ve anlam (semantics) kurallarına uyulmak zorundadır. Binlerce programlama dili vardır ve her yıl yenileri oluşturulmaktadır.

Program bilgisayar programlama dilleri ile bilgisayarlara verilen istekler (komutlardır) ve bu programlama dilleri üç farklı kategoride sınıflanır (Yazılım (program) - Vikipedi, 2011) (Şekil 5):

- **Alt Seviye Yazılım Dilleri (Low-Level Programming Languages):** Makine koduna olduk[ç]a yakın programlama dilleridir. Makina h[â]kimiyeti olduk[ç]a gelişmiştir. Bu programlama dillerini bilen kişilerin mikro işlemciler hakkında bilgi sahibi olması gereklidir. Assembly programlama dili gibi.
- **Orta Seviye Yazılım Dilleri (Medium-Level Programming Languages):** Olduk[ç]a esnek olan bu diller hem üst hem alt seviye programlama yapabilirler. C programlama dili gibi.
- **Üst Seviye Yazılım Dilleri (High-Level Programming Languages):** Olay tabanlı programlama dilleri olarak da adlandırılırlar, ... [ancak] bu programlama dilleri sadece belirli fonksiyonlar etrafında çalışırlar ve programlama h[â]kim[i]y[e]tini azaltırlar (bu nedenle vir[ü]sler çok nadir olarak üst seviye programlama dili kullanarak yazılırlar) bunun yanında en hızlı ve en etkili programlama dilleri bu kategoridedir. [Ö]rneğin [V]isual [B]asic gibi.



Şekil 5. Programlama/Yazılım Dilleri Kategorileri

Bir programlama dili bir makine tarafından özellikle de bir bilgisayar tarafından gerçekleştirilebilecek işlemleri (hesaplamaları) ifade etmek için tasarlanmış yapay bir dildir (Programming language - Wikipedia, 2011).

Bir programlama dili genellikle sözdizimi/biçim (syntax - form) ve anlam (semantics - meaning) olarak iki ögeye ayrılmaktadır ve birçok programlama dilinde sözdizimi ve/ya anlamı belirtmek için bazı yazılı tanımlamalar (kurallar) bulunmaktadır (Programming language - Wikipedia, 2011) (Şekil 6).



Şekil 6. Programlama Dili Öğeleri

Programlama dilleri çok çeşitli biçimlerde sınıflandırılmakla birlikte genel olarak aşağıda belirtilen tasarlanan/amaçlanan kullanım alanlarına bağlı olarak aşağıdaki programlama paradigmaları bağlamında ele alınabilir (Programming language - Wikipedia, 2011):

- Emir belirten ifadeleri içeren programlama dilleri (Imperative programming languages)
- Bildiren/deklare eden ifadeleri içeren programlama dilleri (Declarative programming languages)
- Prosedür ile ilgili ifadeleri içeren programlama dilleri (Procedural programming languages)

- Nesneye yönelik ifadeleri içeren programlama dilleri (Object-oriented programming languages)
- Fonksiyonel ifadeleri (matematiksel fonksiyonları) içeren programlama dilleri (Functional programming languages)
- Matematiksel mantığa ilişkin ifadeleri içeren programlama dilleri (Logic programming languages)
- İşlemlere yönelik ifadeleri içeren programlama dilleri (Process-oriented programming languages) (Process-oriented programming - Wikipedia, 2011)

Bir programlama dili birden fazla/çoklu paradigmayı (multiple paradigms) destekleyebilir, örnek vermek gerekirse C++ programlama dilinde yazılmış programlar yalnızca prosedürleri içerebileceği gibi sadece nesneye yönelik olabilir veya her iki paradigmanın da öğelerini birlikte kapsayabilir (Programming paradigm - Wikipedia, 2011).

Programlamanın Temelleri¹

Bilgisayarda programlamaya ilişkin yanlış bilinen ifadeler:

- Herkes programlama dilini öğrenemez (Yanlış)
- Herkes bir programlama dili öğrenebilir (**Doğru**)
- Bilgisayar programlama yüksek bir zekâ gerektirir (Yanlış)
- Bilgisayar programlama yüksek bir zekâ gerektirmez (**Doğru**)

¹ Bu altbölüm TC MEB MEGEP (Mesleki Eğitim Ve Öğretim Sisteminin Güçlendirilmesi Projesi) Bilişim Teknolojileri Programlama Temelleri Modülü'nden (Öğretim Materyali/Ders Notu) yararlanılarak hazırlanmıştır.

- Bilgisayar programlama yüksek bir matematik bilgisi gerektirir (Yanlış)
- Bilgisayar programlama yüksek bir matematik bilgisi gerektirmez, sabır ve öğrenme isteği gerektirir (**Doğru**)

Bilgisayarda programlamaya ilişkin öncelikli olarak bilinmesi gerekenler:

- Programlama bir yetenektir. Dolayısıyla bazı insanlar diğerlerinden daha iyidir ama herkes uygulama yaparak daha iyi olabilir.
- Programlama eğlencelidir ama dikkatli ve doğru çalışmayı gerektirir.

İlgili modülün “Giriş” kısmında aşağıdaki bilgiler yer almaktadır (MEGEP-BTPT, 2007, 1):

Programlamada, bir problemin çözüm aşamalarının sıralı bir şekilde yazılmasına “**Algoritma**”, bu aşamaların şekillerle gösterilmesine ise “**Akış Diyagramı**” denir. Algoritmayı, günlük hayattaki bir probleminizi çözerken yapmanız gereken uygulamaları belli bir düzene sokma işlemine benzetebilirsiniz. Algoritma, programlamanın temelidir. Önce çözüm belirlenir sonra kullanılacak programlama diline uygun komutlarla program yazılımı tamamlanır. Algoritma bir binanın temeli gibidir. Temeli sağlamsa bina ayakta durur. Bu yüzden algoritmanın iyi tasarlanması, programın temelinin iyi oluşturulması gerekmektedir.

Bilgisayara yaptırmak istediğiniz şeyi iki şekilde gerçekleştirebilirsiniz (MEGEP-BTPT, 2007, 4):

- Adım adım bir program yazarak
- Uygun bir program satın alarak

Bazı programların giriş ve çıkışları (MEGEP-BTPT, 2007, 4):

Tür	Giriş	İşlem	Çıkış
Kelime İşlemci	Klavyeden karakterler girme	Yazıyı biçimlendirme, yazımını kontrol etme	Yazılanların ekranda görüntülenmesi veya yazıcıdan çıkış/çıkış alınması
Oyun	Fare, klavye ve oyun oynama kolu tuşlarına basma	Ekranda bir animasyon görüntülemeye ilişkin hesaplamaları yapma	Ekranda bir animasyon görüntüleme
Muhasebe	Şu anki ve	Piyasadaki fiyat	Gelecekteki

Programı	geçmiş fiyatları girme	etkilerini tanımlama	ürün fiyatını görüntüleme
Web Tarayıcı	HTML kodları yazma	Kodları resim ve yazıya dönüştürme	Web sayfasını ekranda görüntüleme

İyi bir programın temel özellikleri şunlardır (MEGEP-BTPT, 2007, 4):

- **Doğruluk:** Verilen görevlerin tam olarak yerine getirilmesidir.
- **Dayanıklılık:** Beklenmedik hatalardan dolayı programın çalışması kesilmemelidir.
- **Genişletilebilirlik:** İleri aşamalarda görevlerin değişikliği veya yenilerinin eklenmesi kolay olmalıdır.
- **Basitlik:** Karmaşık tasarımlardan kaçınmak gerekir.
- **Modülerlik:** Program kodları başka programlar içinde de kullanılabilmelidir.
- **Uyumluluk:** Başka bilgisayar ve sistemlerde çalışabilmelidir.
- **Kontrol Edilebilirlik:** Hata olabilecek yerlere açıklayıcı hata mesajları konulmalıdır.
- **Kolay Kullanım:** Kullanıcı arabirimi kolay olmalı ve rahat öğrenilebilmelidir.
- **Parçalanabilirlik:** Problemin küçük parçalara ayrılarak yazılmasıdır.
- **Anlaşılabilirlik:** Başkasının yazdığı program elden geçirilirken rahatça okunabilmelidir.
- **Koruma:** Modüller birbirlerine müdahale etmemelidirler.

Neden Birçok Programlama Dili Vardır?

Her programlama dili özel bir amaca hizmet eder. İnsanlar farklı problemleri çözmek için değişik programlama dilleri yazmışlardır. Temelde bilgisayarlar 1 ve 0'lardan oluşan ikilik sayı sistemindeki dili anlarlar. Buna “**makine dili**” denir.

Aşağıda “TBT182” karakterlerinin ikilik sayı sisteminde temsil edilmesine ilişkin örnek yer almaktadır:

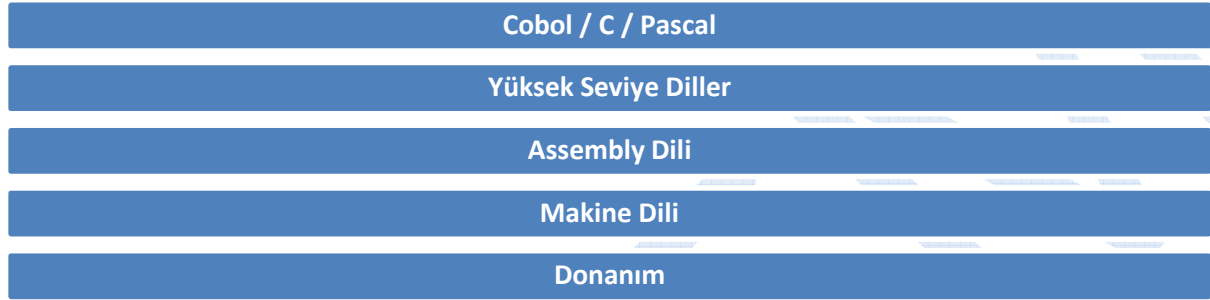
TBT182	01010100 01000010 01010100 00110001 00111000 00110010
--------	----------------------------------------------------------

Makine dilinin dezavantajları, kodları yazarken hata yapılma olasılığının fazla olması ve yazımının uzun sürmesidir. Makine dilinin daha rahat programlanması için 1950’li yıllarda “**Assembly**” dili geliştirilmiştir.

“**Assembly**” dili basit, hatırlanması kolay deyimlerden oluşur.

Programcılar 1 ve 0 ile program yazma yerine “**Assembly**” dilini geliştirmişlerdir, bilgisayarın bu yazılanlarla ilgili hiçbir fikri yoktur. Bu sebeple programcılar “**Assembly**” dil komutlarını “**makine dili**”ne çeviren programlar yazmışlardır. Bu programlara “**makine dili çeviricisi**” (assembler) denir. Böylece “**Assembly**” dili ile yazılmış bir kod, bilgisayarın anlayabileceği “**makine dili**”ne dönüşmüş olur.

Temel olarak makine dilinde; mantıksal işlemler, aritmetik işlemler, dallanma işlemleri ve veri hareket işlemleri yapabiliriz. Bu işlemleri kullanarak ister basit ister çok karmaşık programlar yazabilirsiniz.



Şekil 7. Programlama Dillerinin Genel Görünümleri

Programcılar donanıma erişimi daha kolay olan, okunaklı, yazımı ve düzenlemesi kolay olan bir dile ihtiyaç duymuşlardır. Bunun üzerine C (1972) taşınabilir dili geliştirilmiştir. Cobol (1959) ve Fortran (1957) gibi birçok diller de vardır ama hala günümüzde yaygın olarak kullanılan C dili olmuştur.

C diline ilişkin bir kod örneği aşağıda verilmiştir:

```
main()  
{  
    printf ("Hello, World!\n");
```

```
}
```

Ekrana “Hello, World!” yazan bu örnek İngilizce diline benzer kodlamaya sahiptir. İnsanların konuşma diline yakın olan dillere “**yüksek seviye dil**” denilmiştir.

Bir dilin okunaklı ve kolay yazılabilir olması dışında, bilgisayarın donanımına ulaşması ve başka bilgisayarda da rahatlıkla çalışması gereklidir. Yüksek seviye bir dili “**makine diline**” çeviren programlara “**derleyici**” (compiler) denir. Eğer uygun bir derleyiciniz var ise programınızın kaynak kodunu başka bilgisayarda derleyerek çalıştırabilirsiniz. Örneğin, Macintosh’ta yazılan bir programın kodlarını Windows’ta biraz değiştirerek kullanabilirsiniz.

C programlama dili diğer Cobol ve Fortran gibi dillerden daha basit olduğu için, C dilini makine diline çeviren birçok derleyici yazılmıştır. C dili böylece öyle çok yaygınlaşmıştır ki, bu dile uyumlu olan diller ortaya çıkmıştır. Mesela C++ (C plus plus), Java, Perl, Python ve C# (C Sharp) gibi. Şu anda kullanılan tanınmış, tanınmamış birçok program C veya C++ ile yapılmıştır: Windows, Unix, Microsoft Office gibi.

Programlama dilleri özel amaçlar için yazıldıkları için o dilde her istediğinizi yapmak zordur. Mesela Fortran matematik hesaplamalar için

yapılmıştır, bu dilde bir işletim sistemi yazmak gereksiz yere çok uzun zaman alan bir işlem olur.

Mühendisler diğer insanların rahatça program yazmaları için Basic (Beginners All Purpose Symbolic Instruction Code - 1964) ve Pascal (1971) gibi diller üretmişlerdir. Basit bir dilden yola çıkılarak istenen dile geçiş yapılabilir.

Basic diline ilişkin bir kod örneği aşağıda verilmiştir:

```
PRINT "Hello, World!"
```

Gördüğünüz gibi ekrana bir mesaj yazmak için Basic'te sadece bir satır kod yeterli oluyor. Bu dil sayesinde insanlar kodun yazımı ile uğraşmaya değil, amaçladıkları işe odaklanabiliyorlar.

C ve Basic arasında kalan Pascal ise biraz daha iyi görünümlü, yapısal program yazmayı sağlıyor. Pascal diline ilişkin bir kod örneği aşağıda verilmiştir:

```
Program Message;  
  
Begin  
  
    Writeln ('Hello, World!');
```

End.

Lisp 1950'li yıllarda yapılmış bir derlenebilen mantık programlama dilidir. Prolog ("Programming in Logic" kelimelerinin kısaltması) adındaki 1972 yılında ortaya çıkan dil ise, "mantık programlama dili" olarak tarif edilebilir. Komut dizileri yerine mantık deyimleri ile program yazılır. Genellikle yapay zekâ ile ilgili konularda kullanılır. Yazımı ve kuralları çok basittir.

Yüksek seviyeli dillerin genel özellikleri şöyle özetlenebilir:

- "Makine diline" göre daha fazla kod içerirler ve yavaş kod meydana getirirler.
- Tüm sistem kaynaklarına ulaşamayabilir.
- Bir derleyici gereklidir.
- Kısa zamanda program yazmaya başlanır.
- Öğrenme ve ustalaşma için geçen zaman fazla değildir.
- Yanlışlıkla sistem kaynaklarının bozulmaması için önlemleri vardır.
- Okuması ve değiştirmesi kolaydır.
- Başka bilgisayar çeşitlerinde de çalışabilirler, yani taşınabilirlerdir.

Referanslar

Bilgisayar donanımı - Vikipedi. (2011). Bilgisayar donanımı - Vikipedi.

Çarşamba, 9 Şubat 2011 16:04:53 +0200 Tarihinde

http://tr.wikipedia.org/wiki/Bilgisayar_donan%C4%B1m%C4%B1

Web Adresinden Alınmıştır.

Comparison of programming languages - Wikipedia. (2011). Comparison

of programming languages - Wikipedia. Perşembe, 10 Şubat 2011
19:13:37 +0200 Tarihinde

http://en.wikipedia.org/wiki/Comparison_of_programming_languages Web Adresinden Alınmıştır.

Computer software - Wikipedia. (2011). Computer software - Wikipedia.

Pazartesi, 7 Şubat 2011 10:46:13 +0200 Tarihinde

<http://en.wikipedia.org/wiki/Software> Web Adresinden Alınmıştır.

Hardware - Wikipedia. (2011). Hardware-Wikipedia. Pazartesi, 7 Şubat

2011 10:45:01 +0200 Tarihinde

<http://en.wikipedia.org/wiki/Hardware> Web Adresinden Alınmıştır.

Longman-Metro. (1993). Longman-Metro: Büyük İngilizce Türkçe-

Türkçe Sözlük. İstanbul: Longman Group UK Limited ve Metro
Kitap Yayın Pazarlama A. Ş.

MEGEP-BTPT. (2007). TC MEB MEGEP (Mesleki eğitim ve öğretim

sisteminin güçlendirilmesi projesi) Bilişim teknolojileri

programlama temelleri. Salı, 22 Şubat 2011 02:48:58 +0200 Tarihinde

<http://www.cizgi->

tagem.org/resource/vfiles/tagem/dms_file/3037/programlama%20temelleri.pdf Web Adresinden Alınmıştır.

Personal computer hardware - Wikipedia. (2011). Personal computer hardware - Wikipedia. Pazartesi, 7 Şubat 2011 10:52:02 +0200 Tarihinde

http://en.wikipedia.org/wiki/Personal_computer_hardware Web Adresinden Alınmıştır.

Process-oriented programming - Wikipedia. (2011). Process-oriented programming - Wikipedia. Pazar, 20 Şubat 2011 05:31:31 +0200 Tarihinde http://en.wikipedia.org/wiki/Process-oriented_programming Web Adresinden Alınmıştır.

Programming language - Wikipedia. (2011). Programming language - Wikipedia. Perşembe, 10 Şubat 2011 19:14:08 +0200 Tarihinde http://en.wikipedia.org/wiki/Programming_language Web Adresinden Alınmıştır.

Programming paradigm - Wikipedia. (2011). Programming paradigm - Wikipedia. Pazar, 20 Şubat 2011 05:09:23 +0200 Tarihinde http://en.wikipedia.org/wiki/Programming_paradigm Web Adresinden Alınmıştır.

Programming tool - Wikipedia. (2011). Programming tool - Wikipedia. Perşembe, 10 Şubat 2011 19:47:16 +0200 Tarihinde http://en.wikipedia.org/wiki/Programming_software Web Adresinden Alınmıştır.

Yazılım - Vikipedi. (2011). Yazılım - Vikipedi. Çarşamba, 9 Şubat 2011

16:06:09 +0200 Tarihinde

<http://tr.wikipedia.org/wiki/Yaz%C4%B1l%C4%B1m> Web

Adresinden Alınmıştır.

Yazılım (program) - Vikipedi. (2011). Yazılım (program) - Vikipedi.

Çarşamba, 9 Şubat 2011 16:06:29 +0200 Tarihinde

<http://tr.wikipedia.org/wiki/Program> Web Adresinden Alınmıştır.