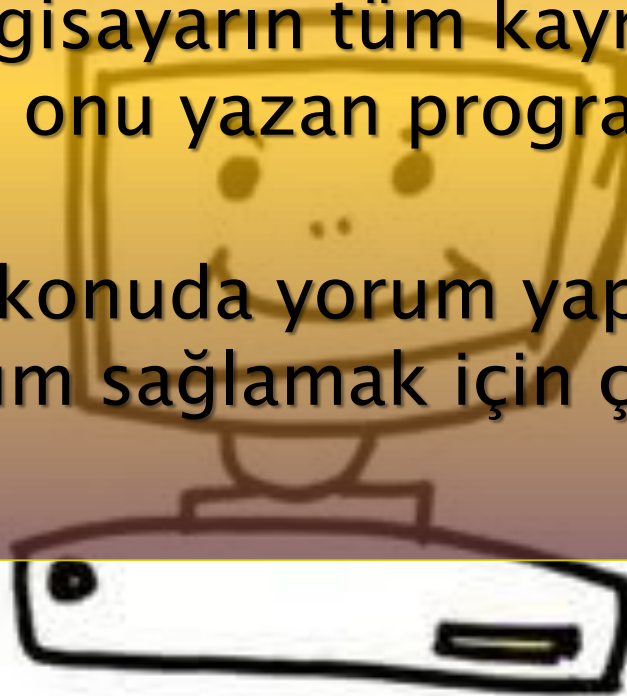


# PROGRAMLAMA TEMELLERİ

Programlama Dilleri

# Programlama

- ▶ Bilgisayar öğrendiğini unutmaz, eğer iyi programlarsanız kusursuz olarak işlemleri yapar
- ▶ Yorulmadan hep aynı işlemi tekrar yapabilir.
- ▶ Programlar bilgisayarın tüm kaynaklarına erişebilir, tabii onu yazan programcı izin vermişse.
- ▶ Bilgisayar, bir konuda yorum yapamaz, yeni durumlara uyum sağlamak için çaba sarf etmez.



# İnsan vs. Bilgisayar

## Human & Computer

- ▶ **İnsan**
  - Unutkandır,
  - Hata yapabilir,
  - Yorulur,
  - Beyninin tamamını kullanamaz,
  - Moral durumu değişebilir,
  - Duygusal olarak etkilenir.
- ▶ **İnsanların en büyük avantajı,**
  - Yeni durumlar karşısında bocalasa bile zamanla uyum sağlayabilir.
  - Başına gelen olaylardan ders çıkartabilir.
  - Tek başına birçok sorunu çözebilir.
- ▶ **Bilgisayar ise**
  - Her zaman aynı tepkiyi verir,
  - Kendini geliştiremez
  - İnsan tarafından kontrol edilmedikçe etkinliklerini değiştirmezler.

I rely on electricity

I'm good  
at computing  
and logics

But I'm better at  
multitasking

I'm a  
good learner

# Bilgisayar Kullanım Alanı

## ► Bilgisayar

- ekonomi,
- bilim,
- mühendislik,
- eğitim,
- askeri alanlarda yardımcı olması için üretilmiştir.
- Çok karmaşık formüllerin sonucunu kısa zamanda elde etmek için programlar yazılmıştır.
- Özellikle İkinci Dünya Savaşı veri şifreleme,
- silahların hedefi daha doğru bulması gibi konular sebebi ile bilgisayarın gelişimi hızlanmıştır



# Programlama Dilleri

- ▶ Bir programlama dilini neden öğreniriz?
  - eğlence için,
  - bir ihtiyacı gidermek için,
  - kariyer için
  - Zekânızı kanıtlamak için
  - Para kazanmak için (her ne kadar birincil hedef gibi görünse de, eğer işinizi severek yapmıyor iseniz, ne kadar kazandığınızın pek önemi olmaz.)

Visual Basic Simula Visual C++  
APC HTML Cobol PL/I Algol  
Fortran Basic  
awk ADA Icon  
Lisp Pascal  
Smalltalk  
Modula-2 perl  
Scheme Delphi JAVA



# Programlama Dilleri

- ▶ Eğer bilgisayara ne yapması gerektiğini söylemezseniz, hiçbir şey icra etmez.
- ▶ Bilgisayara istediğiniz şeyi iki şekilde yaptırabilirsiniz:
  - Adım adım bir program yazarak
  - Uygun bir program satın alarak

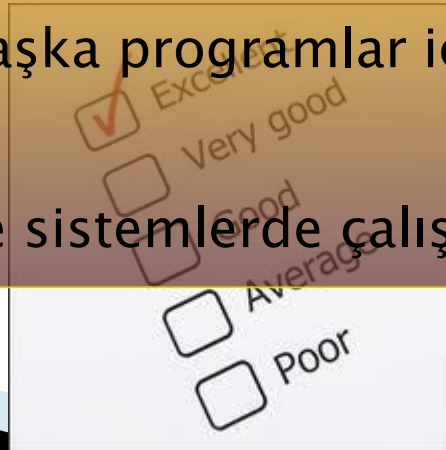


# Bazı Programlar

Tür	Giriş	İşlem	Çıkış
Kelime İşlemci	Klavyeden girilen karakterler	Yazıyı biçimlendirir, yazımını kontrol eder	Düzgünce ekrandan veya yazıcıdan çıkış
Oyun	Fare, klavye ve oyun çubuğu tuş basımları	Ekranda hızlı bir şekilde bir animasyon hesaplama	Ekrandaki figürler
Muhasebe Programı	Şu anki ve geçmiş fiyatlar	Piyasadaki fiyat etkilerini tanımlama	Gelecekteki ürün fiyatı
Web Tarayıcı	HTML kodları	Kodları resim ve yazıya dönüştürme	Web sayfasını ekranda gösterme

# İyi Bir Program

- ▶ **Doğruluk**
  - Verilen görevlerin tam olarak yerine getirilmesidir.
- ▶ **Dayanıklılık**
  - Beklenmedik hatalardan dolayı programın çalışması kesilmemelidir.
- ▶ **Genişletilebilme**
  - 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.





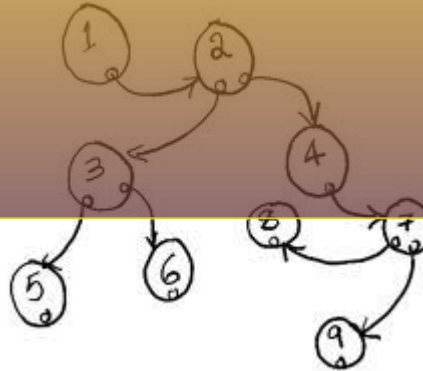
# İyi Bir Program

- ▶ **Kontrol edilebilirlik**
  - Hata olabilecek yerlere açıklayıcı hata mesajları konulmalıdır.
- ▶ **Kolay kullanım**
  - Kullanıcılara birimi kolay olmalı ve rahat öğrenilebilmelidir.
- ▶ **Parçalanabilirlik**
  - Problemin küçük parçalara ayrılarak yazılmasıdır.
- ▶ **Anlaşılabilirlik**
  - Başkasının yazdığı program elden geçirilirken rahatça okunabilmelidir.
- ▶ **Koruma**
  - Modüller birbirlerine müdahale etmemelidirler.

```
WHILE Y < 10:  
  PRINT Y  
  Y=Y+1  
  YOU'RE STUPID!  
  IF Y=10:  
    PRINT "GAME OVER"  
  Y=Y-1  
  Y=Y+1  
TIMER=0  
WHILE TIMER < 10000  
  TIMER=TIMER+1
```

# Algoritma

- ▶ Temel olarak bir algoritma, bilgisayara belli bir problemin çözümünü anlatmaktır.
- ▶ Dünya problemlerden oluştuğuna göre, pratikte insanların yazabileceği program sayısı ve çeşitliliğinin bitmesi çok zordur.
- ▶ Büyük bir problemin bilgisayara yaptırılması için, genellikle küçük parçalara bölmek gereklidir.



Ares: RawMat drone leaving with 35.00 raw materials  
SOUNDSYS> Dynamic source index 27 released.  
SOUNDSYS> Dynamic source index 27 released.  
Viewing from Carrier Epsilon  
Viewing from Manta2  
- Game -

[Vulcan] Manta2  
Sp:0.08 X:-49959.51 Y:1193.00  
P:-9.40 R:3.70 H:316.30  
State:[Manual control] Order:[None]

# Algoritma

Emitters: 0 Camera: [Vulcan] X:-49877 Z:50391 Y:145 H:136  
Particles: 0  
Projectiles: 0

Rnd: 0  
Coin: 0

- ▶ Örneğin bir oyunda şu adımları çözmek gerekebilir:
  - □ Kullanıcının bir nesneyi (araba, adam, uzay gemisi...) nasıl hareket ettirebileceği
  - □ Nesnenin duvar, uçurum veya çevredeki başka nesnelere göre çarpmadan, düşmeden nasıl hareket edeceği
  - □ Gerçekçi olarak çevre tasarımı yapmak
  - □ Gelen merminin nereye çarptığını bulmak ve
  - □ Oyuncunun sağlık durumunu ekrana yazmak

Game speed x0.06  
Game speed x0.13  
Game speed x0.25  
Game speed x0.13  
Game speed x0.06  
- Game speed -

[Mecastor] Manta1 FPS: 130  
Sp:0.08 X:-50794.73 Y:224.71 Z:50653.53  
P:-8.40 R:3.70 H:310.50  
State:[Autopilot] Order:[Undefined]

# Algoritma

Emitters: 0  
Particles: 0  
Projectiles: 0

Camera: [Mecastor] X:-50795 Z:50654 Y:225 Ht:5

Rndr: 6.716  
CPU : 0.728  
Coin: 0.084

- ▶ Programlama genellikle zor değildir, ama zaman harcayan bir iştir.
- ▶ Bir pinpon oyununu yazmak, bir savaş uçağının simülasyonunu yapmaktan daha kolaydır.
- ▶ Eğer adım adım bir yeri tarif edebiliyorsanız, program yazabilirsiniz.
- ▶ Bir bilgisayar aptal olduğuna göre ona ne yapacağını adım adım anlatmanız gereklidir.

Health Spd Fuel Alt

Manta1  
Island: Mecastor  
State: Autopilot  
Autopilot: [Autopilot off]  
Fuel: 86%  
Load: 37.5Kg  
FuelCons: 0.00016

Target: Manta3  
Range: 50.04  
Health: 100%  
[Autopilot]

Assassin AAM [2]

# Algoritma

- ▶ Arkadaşınıza evinizin yolunu tarif ederken:
  - Yayla mahallesine git
  - Çınar sokağına git
  - Trafik ışıklarından sağa dön
  - İki sokak ileriden sola dön

- ▶ İnsana kolay gelen bu adımlar bilgisayara anlamsız gelir:
  - ▶ ▪ Yayla mahallesine nasıl gideceğim?
  - ▶ ▪ Çınar sokağını nasıl belirleyeceğim?
  - ▶ ▪ Işıklardan ne kadar sonra sağa döneceğim?
  - ▶ ▪ Sola dönünce arabayı nasıl park edeceğim?

IT'S WITHIN WALKING DISTANCE IF YOU HAVE THE TIME, SIR!

# Programlama

► Programlar her zaman işe yaramayabilir. 1980'li yıllarda Amerikan ordusu "Teğmen York" isminde bir pilotsuz uçan uçak savar planlamıştı. Basit olarak amaç, düşman uçağını bulup yok etmektir. Milyonlarca dolar harcadıktan, saatlerce program yazıldıktan ve test edildikten sonra programcılar çalışacağına emin oldular. En üst rütbeli subaylar ve görevliler önünde, deneme için hazırlık yapıldı. Maalesef deneme sırasında, "Teğmen York" tüm silahlarını düşmana değil, komutanların üzerine doğrultmuştu. Neyse ki insanlar etrafa dağılırken, füzelerini ateşlememişti! Bu korkunç denemeden sonra proje rafa kaldırıldı.

# Programlama için

- ▶ Kullanmaktan çok, program yazma konusunda istekli iseniz, zaten program yazmak için gerekli şeye sahipsiniz demektir.
- ▶  **İstek:** Önünüze ne kadar engel çıksa da, isteğiniz varsa öğrenirsiniz. (Kanuni olmayan bir şey ile ilgileniyorsanız, hapisanede geçirilecek zamanınız olabilir!)
- ▶  **Merak:** Bu sayede öğreneceğiniz dil size angarya gibi gelmez.
- ▶  **Hayal gücü:** Böylece daha ilginç ve faydalı program yapabilirsiniz.

# Programlama Dili

- ▶ Bilgisayar Türkçe, İngilizce veya başka bir dilden anlamaz.
- ▶ Bilgisayarın fonksiyonel bir beyni olmadığı için insanlar komutlar yazmalıdır. Bu özel dile **programlama dili** denir.
- ▶ Komutlar bir araya gelerek **programı** meydana getirir. Belli bir dil ile yazılmış komutlara **kaynak kod** da diyebiliriz.





# Makine Dili

- ▶ 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.

# Assembly

- ▶ 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 basit, hatırlanması kolay deyimlerden oluşur.
- ▶ Bilgisayarın bu yazılanlarla ilgili hiçbir fikri yoktur. Bu sebeple programcılar “assembly dil” komutlarını “makine dili”ne çeviren programlar yazmışlardır; **assembler**.

# Assembly

- ▶ 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.
- ▶ “Assembly dili” makine dilinden daha rahat yazılmasına rağmen, hala bazı dezavantajlara sahiptir:
  - ▶ □ Kod yazmak uzun sürer. İyi bir ekip ister.
  - ▶ □ Başka dile çevrilmeleri zordur.
  - ▶ □ Çok uzun program yazımına elverişli değildir.

```
section '.data'
dosseg
.model small
.stack 100h
.data
my_message db "ouch!", 0, 0ah, 0, 0
main proc
mov ax, 1
mov ds, ax
mov ah, 9
mov dx, offset my_message
int 21h
mov ax, 4c0h
int 21h
add esp, 4
end main
```

# Programlama Dilleri

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

# Programlama Dilleri

## List of programming languages by category

From Wikipedia, the free encyclopedia

This is a list of **programming languages** grouped by category. Some languages are listed in multiple categories.

### Contents [hide]

- 1 Array languages
- 2 Aspect-oriented languages
- 3 Assembly languages
- 4 Authoring languages
- 5 Command line interface languages
- 6 Compiled languages
- 7 Concurrent languages
- 8 Curly-bracket languages
- 9 Dataflow languages
- 10 Data-oriented languages
- 11 Data-structured languages
- 12 Declarative languages
- 13 Esoteric languages
- 14 Extension languages
- 15 Fourth-generation languages
- 16 Functional languages
  - 16.1 Pure

### Programming language lists

- [Alphabetical](#)
- [Categorical](#)
- [Chronological](#)
- [Generational](#)

# Programlama Dilleri

- ▶ Ekranına “Selam!” yazan bu örnek İngilizce diline benzer kodlamaya sahiptir.
- ▶ İnsanların konuşma diline yakın olan dillere **yüksek seviye dil** denilmiştir.
- ▶ Aslında programlama dilinin İngilizce olması şart değildir. Kendi dilimizde olan örnek program Tupol ile yazılmıştır

## C dilinde kod örneği

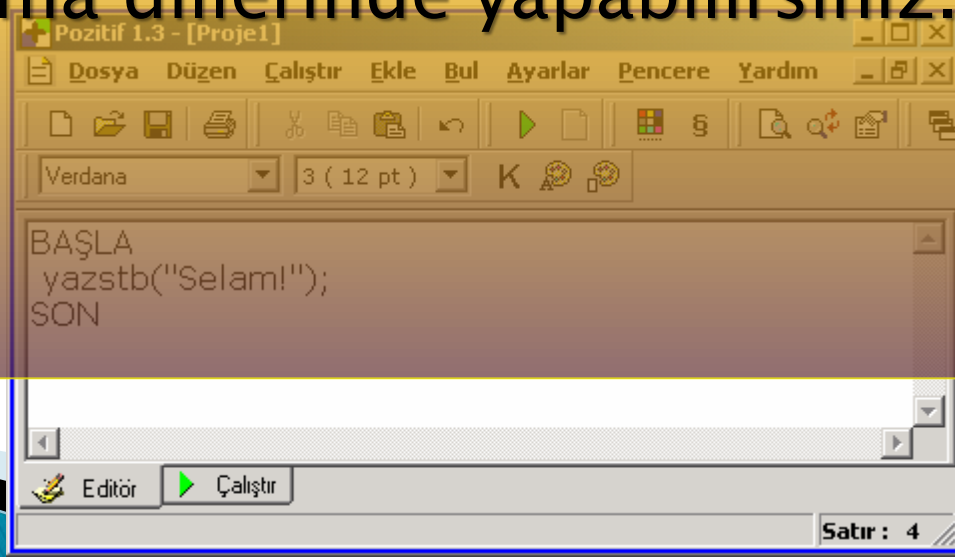
```
main()  
{  
printf ("Selam!\n");  
}
```

## Tupol dilinde kod örneği

```
PROGRAM SelamProgrami;  
Basla  
Yazi(#i,"Selam!");  
Bitti.
```

# Programlama Dilleri

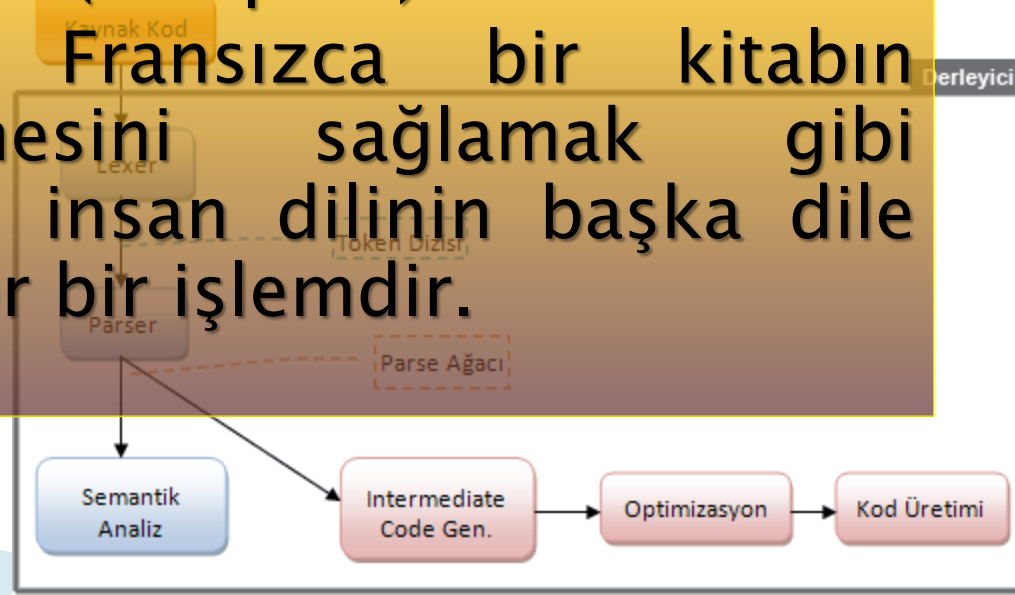
- ▶ Her ne kadar kendi dilimizde program yapmak gurur ve heyecan verici olsa da, ileride göreceğiniz bir dilin temel gereklilikleri sebebi ile bu dillerde büyük (karmaşık) programlar yapmak zordur. Programlamaya ısınmak için, ilk denemelerinizi bu programlama dillerinde yapabilirsiniz.



```
BAŞLA
yazstb("Selam!");
SON
```

# Programlama Dilleri

- ▶ **Dil**
  - Anlaşılır
  - Kolay kodlanabilir
  - Bilgisayar donanımına ulaşabilmeli
  - Başka bilgisayarda da rahatlıkla çalışabilmeli
- ▶ Yüksek seviye bir dili makine diline çeviren programlara **derleyici (compiler)** denir.
- ▶ Derleyiciyi aynen Fransızca bir kitabın Türkçeye çevrilmesini sağlamak gibi düşünebiliriz. Tabii insan dilinin başka dile çevrimi çok daha zor bir işlemdir.





# Programlama Dilleri

- ▶ İyi bir derleyici programınızın kaynak kodunu başka bilgisayarda derleyerek çalıştırabilir. 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

- ▶ 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.

Basic dilinde kod örneği

```
PRINT "Selam!"
```

Pascal dilinde kod örneği

```
Program Message;  
Begin  
    Writeln ('Selam!');  
End.
```

# Yüksek Seviye PL

- ▶ □ “Makine diline” göre daha şişkin ve yavaş kod meydana getirirler.
- ▶ □ Tüm sistem kaynaklarına ulaşamayabilir, tabii amacınız disk yardımcı programı (Norton SystemWorks...) gibi program yazmak değilse...
- ▶ □ Bir derleyici gereklidir.
- ▶ □ Kısa zamanda program yazmaya başlanır.
- ▶ □ Öğrenme ve ustalaşma için geçen zaman fazla değildir.
- ▶ □ Yanlışlıkla sistem kaynaklarının bozulmaması için kalkanları vardır.
- ▶ □ Okuması ve değiştirmesi kolaydır.
- ▶ □ Başka bilgisayar çeşitlerinde de çalışabilirler, yani taşınabilirlerdir.

# PL: SSS

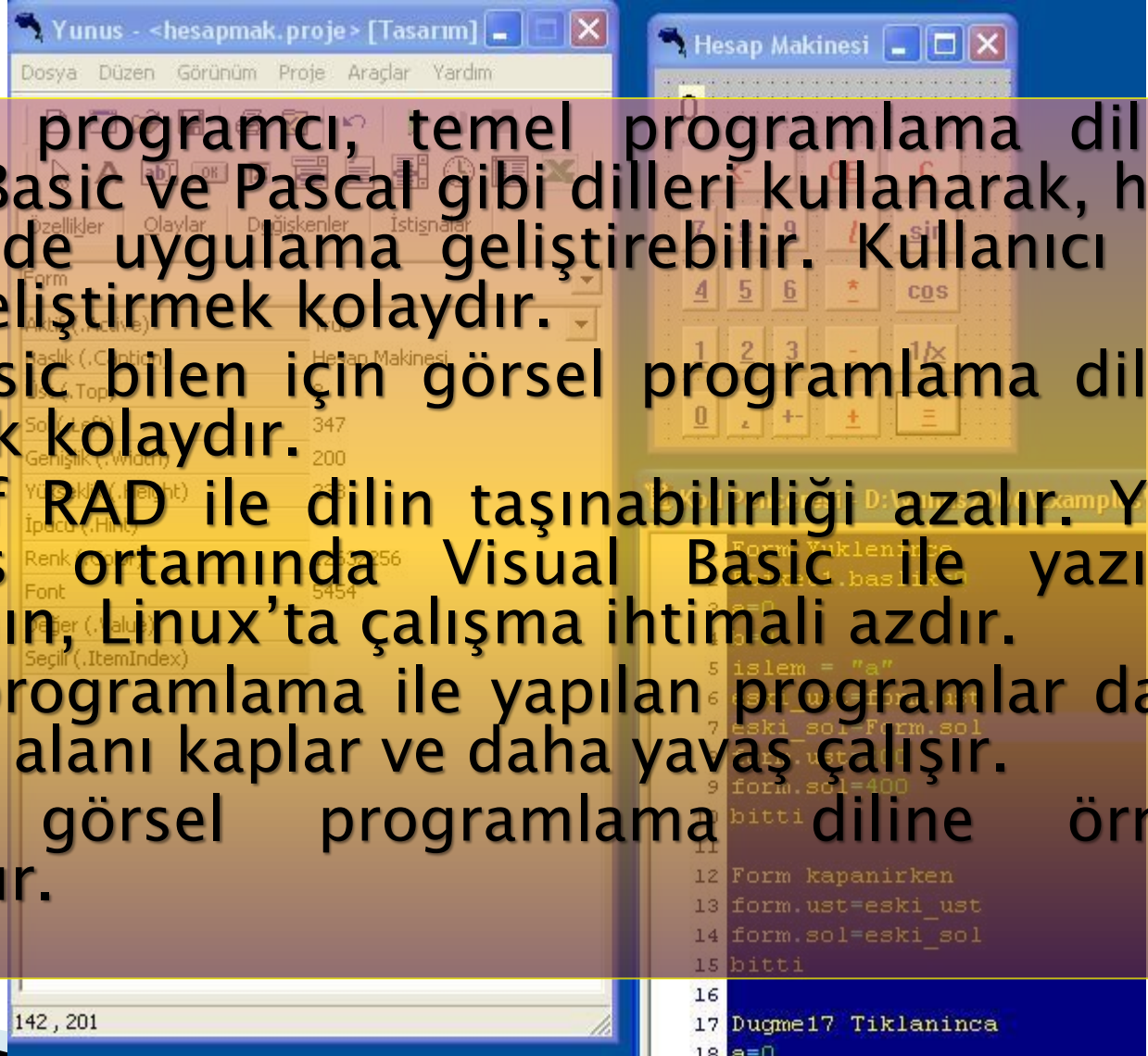
- ▶  Assembly dili ile yüksek seviye dil arasındaki hız farkı ne kadardır?
  - Cevap: Assembly dili yaklaşık 2 ile 20 kat arasında daha hızlıdır.
- ▶ 2. Assembly dili ile yüksek seviye dil arasındaki program tasarım süresi ne kadardır?
  - Cevap: Assembly dili yaklaşık 10 ile 100 kat arasında daha yavaş sürede tamamlanır.
- ▶ 3. Derleyiciler hangi dilde geliştirilmişlerdir?
  - Cevap: Önceleri Assembly dili ile yazılmışlardır. Daha sonra yüksek seviye diller güçlendikçe bu diller kullanılarak derleyiciler yazılmıştır

# RAD

- ▶ Hızlı Uygulama Geliştirme Ortamları
- ▶ Eski metin tabanlı ortamlarda çalışılırdı.
- ▶ Pencere ve düğme gibi kavramlar ise, 1984 yılında Apple "System 1" ve 1992 yıllarında Microsoft Windows 3.1'in ortaya çıkması ile günlük hayatımıza girdi.
- ▶ Pencere, kaydırma çubukları, araç çubukları, menüler programcının daha hızlı bir şekilde program yapmasına sebep olduğu için, bu ortamlara "Hızlı Uygulama Geliştirme - Rapid Application Development (RAD)" denilmiştir.
- ▶ Görsel Programlama Dilleri (Visual Programming Languages) olarak da adlandırılırlar. Programcı oluşturacağı programın ara birimini, istediği gibi tasarlayabilir ve bu ara birimi işlevsel hâle getiren kod kısmını yazabilir.
- ▶ Popüler RAD: Visual Basic, Delphi, JBuilder ve Visual C#.

# RAD

- ▶ RAD ile programcı, temel programlama dilleri olan C, Basic ve Pascal gibi dilleri kullanarak, hızlı bir şekilde uygulama geliştirebilir. Kullanıcı ara birimi geliştirmek kolaydır.
- ▶ C ve Basic bilen için görsel programlama diline geçiş çok kolaydır.
- ▶ Maalesef RAD ile dilin taşınabilirliği azalır. Yani Windows ortamında Visual Basic ile yazılan programın, Linux'ta çalışma ihtimali azdır.
- ▶ Görsel programlama ile yapılan programlar daha çok disk alanı kaplar ve daha yavaş çalışır.
- ▶ Türkçe görsel programlama diline örnek Yunus'tur.



# Veritabanı

- ▶ Bilgisayarların en çok kullanım alanı: bilgi kaydetme ve düzeltmedir. İsimler, adresler, elektronik posta adresleri, telefon numaraları, iş deneyimleri, vs. veri tabanlarında saklanır.
- ▶ Neredeyse tüm firmalar veri tabanı sayesinde işlerini daha rahat yürütürler.
- ▶ Veritabanı işlemlerini kolaylaştırmak için “veritabanı programcılığı” geliştirilmiştir.
- ▶ En çok kullanılan veri tabanı programları Microsoft Access, MSSQL ve MySQL'dir.
- ▶ Verileri düzenleme için kendilerine has dilleri vardır. Bu dile SQL (Structured Query Language – Yapısal Sorgu Dili) denir.
- ▶ dbase, FileMaker ve FoxPro programları ise artık pek kullanılmıyor.
- ▶ Oracle ise, çok büyük veri tabanları barındıran, güvenlik ve hız gerektiren uygulamalarda kullanılır.

# Veritabanı

- ▶ Veri tabanı programcılığının dezavantajları şunlardır:
- ▶ Veri tabanı programları, veri tabanını işleyen program (DBMS) sistemde kurulu değilse çalıştırılmazlar.
- ▶ Sadece veri tabanı programını kullanarak antivirüs, oyun, kelime işlemci gibi program yapılamaz.



# Betik Programlama

- ▶ Birçok program kendi programlama dilinin kullanılması imkânını sağlar.
- ▶ Örneğin Word ve Excel içinde, Visual Basic for Applications (VBA) adı verilen bir dil vardır.
- ▶ Office programları içindeki makrolar (VBA) kullanıcıya yardımcı işlevler sağlar



```
VBA dilinde kod örneği  
Private Sub CommandButton1_Click()  
    MsgBox "Selam!"  
End Sub
```

# Betik Programlama

- ▶ Macintosh işletim sisteminde AppleScript
- ▶ DOS ve Windows'da Batch
- ▶ Windows'da VBS

## AppleScript dilinde kod örneği

```
on DisplayMessage()  
    display dialog "Selam!" buttons {"Tamam"}  
end DisplayMessage  
DisplayMessage()
```

## Batch dilinde kod örneği

```
@Echo off  
Echo Selam!  
Pause
```

## VBS dilinde kod örneği

```
WScript.Echo "Hello World!!"
```

# Betik Programlama

- ▶ Betik dilleri normalde diğer programlama dillerinden daha kolaydır. Bu dillerle;
  - Var olan bir programı özelleştirebiliriz.
  - Çok az kod yazarak, karmaşık bir programı kullanışlı hale getirebiliriz.
- ▶ ◻ Betik dili ana programa bağımlıdır, sadece ana programın kurulu olduğu bilgisayarlarda yazdığınız program çalışabilir.
- ▶ ◻ Programınızın satış ve dağıtımı zordur. Almak isteyen ana programı da satın almak zorunda kalır.
- ▶ ◻ Genel programlama dillerinden daha kısıtlı imkânlarınız vardır.

# Web Programlama

- ▶ Kolay web programcılığı için HTML (HyperText Markup Language - Hiper Yazı İşaret Dili) icat edilmiştir.
- ▶ Bir sitenin HTML olarak kaynağını incelediğinizde çok karmaşık ve dağınık gelebilir.
- ▶ HTML ile site tasarımı çok zor olduğu için JavaScript gibi web sayfası yardımcı programlama dilleri yapılmıştır. Etkileşimli site yapmak, kullanıcının formlara girdiği verileri kontrol etmek, site içine oyun eklemek isterseniz, bunları JavaScript, Java, CGI veya Macromedia Flash ile yapabilirsiniz.

# Web Programlama

- ▶ Kolay web programcılığı için HTML (HyperText Markup Language - Hiper Yazı İşaret Dili) icat edilmiştir.
- ▶ Bir sitenin HTML olarak kaynağını incelediğinizde çok karmaşık ve dağınık gelebilir.
- ▶ HTML ile site tasarımı çok zor olduğu için JavaScript gibi web sayfası yardımcı programlama dilleri yapılmıştır. Etkileşimli site yapmak, kullanıcının formlara girdiği verileri kontrol etmek, site içine oyun eklemek isterseniz, bunları JavaScript, Java, CGI veya Macromedia Flash ile yapabilirsiniz.

# Web Programlama

- ▶ Web programlama dilleri, sitenin ziyaretçi sayısını artırır ve ziyaretçinin uzun süre sitede kalmalarını sağlar.
- ▶ Öğrenmesi kolaydır ve dünyanın herhangi bir yerinden siteye (uygulamaya) ulaşım, değişiklikler yapabilmek imkânı vardır.
- ▶ Dezavantajları:
  - ▶ □ Tüm web tarayıcıları VBScript, JavaScript ve Java desteklemez. (Daha eski tarayıcılar gibi)
  - ▶ □ Bağlantı hızını düşürür.
  - ▶ □ İnternet bağlantısı yok ise sayfanın görüntülenmesi mümkün olmayabilir.

# Programlama Dilleri

## ► Sonuç olarak

- Her konuda bir işe yarayan “mükemmel” dil yoktur.
- Profesyonel olarak program yazmak istiyorsanız, C# gibi yüksek seviye bir dil ve SQL gibi veri tabanı dili öğrenmelisiniz. İş bulmakta zorlanmazsınız.
- Eğer sizden site yapılması isteniyor ise Java, JavaScript, PHP ve ASP.NET gibi dillere aşina olmak kaçınılmazdır.
- Eski bilgisayarlarda hâlâ eski kodların barınmasından dolayı, piyasada az kullanılan diller ile iyi kariyer yapabilirsiniz. Hatta bilen sayısı az olmasından dolayı, iyi ücret alabilirsiniz.

# Programlama Dilleri

## ► Sonuç olarak

- Her konuda bir işe yarayan “mükemmel” dil yoktur.
- Profesyonel olarak program yazmak istiyorsanız, C# gibi yüksek seviye bir dil ve SQL gibi veri tabanı dili öğrenmelisiniz. İş bulmakta zorlanmazsınız.
- Eğer sizden site yapılması isteniyor ise Java, JavaScript, PHP ve ASP.NET gibi dillere aşina olmak kaçınılmazdır.
- Eski bilgisayarlarda hâlâ eski kodların barınmasından dolayı, piyasada az kullanılan diller ile iyi kariyer yapabilirsiniz. Hatta bilen sayısı az olmasından dolayı, iyi ücret alabilirsiniz.



# Çok Dil Kullanma

- ▶ Bazı derleyiciler, kaynak kodu önce özel bir dosya türüne çevirir. “Object file – nesne dosyası” denilen bu dosya, kaynak dilden bağımsız hâle gelir. Farklı dilde oluşturulmuş nesne dosyaları, “linker – bağlayıcı” adı verilen programlar ile birleştirilir. Sonuçta ise çalıştırılabilir tek dosya oluşur.
- ▶ Microsoft Windows, dillerin ortak kod kullanımii için DLL (Dynamic Link Libraries – Dinamik Bağ Kütüphanesi) yöntemi kullanır. Ara birimi olmayan bu dosyalar, farklı programlama dilleri kullanabilsin diye sisteme tanıtılmıştır.
- ▶ Son bir yöntem, Microsoft’un .NET teknolojisidir. Aynı ara birimde ister C#, ister Basic veya başka bir dil ile program yazılabilir. Aynı veri tabanınıve çözümü ortak kullanarak birçok programcı bir arada çalışabilir. Her dilin kendine has avantajı kullanılarak, uygulamalar daha güçlü hâle gelebilir.

Native Code

Operating System Services