



CENG 302-YAZILIM MÜHENDİSLİĞİ

Hafta 1: Tanımlar, Tarihçe, Kapsam

ÖNCE TANISALIM!

GÖKÇEN YILMAZ – KISA ÖZGEÇMİŞ

- ❑ PhD. & MSc. in Information Systems, BSc. in Systems Engineering
- ❑ 9 years of research experience in Software Engineering
 - Pamukkale University, METU, University of Cambridge
- ❑ 9 years of lecturing experience
 - Pamukkale University (for 2 years), METU + seminars in various companies
- ❑ Interested research areas
 - Software Engineering (especially Software Process and Quality Management)
 - Smart Cities and Factories

SIRA SİZDE ?

☐ Adınız-soyadınız,

☐ İlgil alanınız

YAZILIM MÜHENDİSLİĞİ

☐ Yazılım ne demek?

☐ Mühendislik ne demek?

MODERN DÜNYADA YAZILIMIN YERİ

❑ Modern dünya yazılım olmadan işlemiyor.

- Ulusal altyapı ve hizmetler
- Elektrikli ürünler
- Giyilebilir teknoloji
- Endüstriyel üretim
- Eğlence
- ...



→ Yazılım her yerde!!

YAZILIM NEDİR?

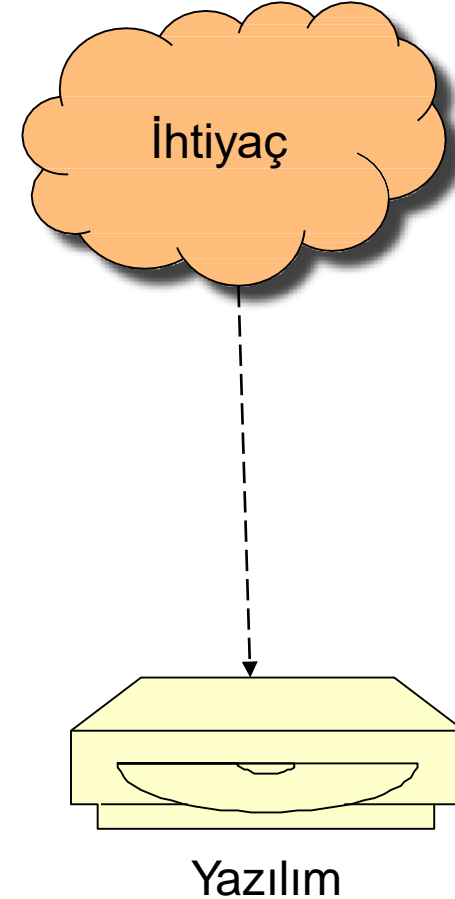
■ Bilgisayar programları ile bunlarla ilişkili yapılandırma ve belgelerin bütünüdür.

► Genel yazılım

- ◆ Özellikleri pazardaki genel ihtiyaca göre belirlenerek geliştirilir.
- ◆ “Generic software”, “commercial-off-the-shelf (COTS) software”
- ◆ Örnek: MS Office yazılımı

► Müşteriye özel yazılım

- ◆ Özellikleri belirli bir müşterinin ihtiyacına göre belirlenerek geliştirilir.
- ◆ “Custom software”
- ◆ Örnek: Hava trafiği kontrol yazılımı



YAZILIMIN GELİŞİMİ

■ 1940'lar:

- ▶ Elle yazılan basit makine kodları

■ 1950'ler:

- ▶ Üretkenlik ve kaliteyi arttırmak için yazılan makro birleştiriciler ve yorumlayıcılar
- ▶ İlk kuşak derleyiciler

■ 1960'lar:

- ▶ Üretkenlik ve kaliteyi arttırmak için yazılan ikinci kuşak derleyiciler
- ▶ Yazılım Mühendisliği NATO Konferansı, 1968: "Yazılım mühendisliği" kavramının tartışılması
- ▶ İlk büyük yazılım projeleri (1000 programcı)
- ▶ Büyük iş alanları için ana-bilgisayarlar ve ticari yazılımlar

■ 1970'ler:

- ▶ UNIX ve kod kütüphaneleri için araçlar
- ▶ Mini-bilgisayarlar ve küçük iş alanları için yazılımlar

■ 1980'ler:

- ▶ Kişisel bilgisayarlar ve iş istasyonları
- ▶ Ticari yazılımlar

■ 1990'lar:

- ▶ Avuç-ıç bilgisayarlar ve web teknolojileri
- ▶ Teknolojideki gelişmeler sebebiyle düşen fiyatlar ve karmaşıklaşan iş talepleri

■ 2000'ler:

- ▶ Globalleşme sebebiyle artan talepler
- ▶ Bütünleşik geliştirme ortamları

■ 2010'lar:

- ▶ Bulutta uygulamalar
- ▶ Giyilebilir teknoloji

YAZILIM ÜRÜNLERİNE TALEP

■ Yazılım ürünlerine olan talep ve yazılım ürünlerinden beklentiler çok hızlı artıyor

► Boeing 777

- ◆ A.B.D. ve Japonya'da 1700 iş istasyonu
- ◆ 4,000,000 Kod Satırı
- ◆ “Kanatları olan yazılım”

► Beyaz eşyalar, cep telefonları, otomobiller

► Akıllı ev ve ofis sistemleri

► ...

FARKLI ÖZELLİKTEKİ YAZILIMLAR

- Gömülü Sistemler (Embedded System)

- İnsülin pompa kontrol sistemi

- Bilgi Sistemi (Information System)

- Medikal kayıt sistemi

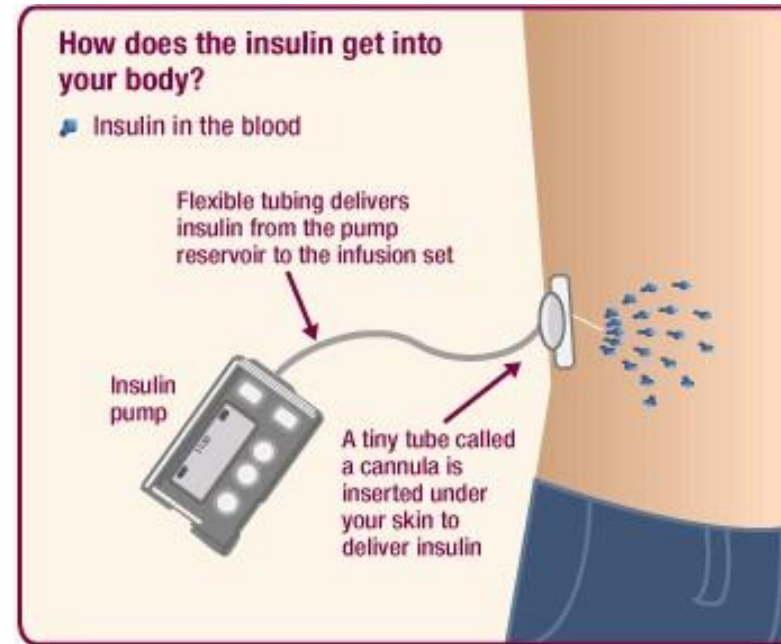
- Sensör Tabanlı Veri Toplama Sistemi (Sensor-based Data Collection System)

- Hava olayları tespiti sistemi

GELENEKSEL DİYABET TESPİTİ



İNSÜLİN POMPA KONTROL SİSTEMİ



“İYİ YAZILIM” – TEMEL ÖZELLİKLER

- Yazılım ürünü, müşterinin beklediği işlevsellik ve performansa ek, aşağıdaki özellikleri de taşımalıdır.
 - ▶ Bakım-yapılabilir (“maintainable”)
 - ♦ Yazılım, değişen ihtiyaçlara göre değişebilir olmalıdır.
 - ▶ Güvenilir (“dependable”)
 - ♦ Yazılım, güvenilir olmalıdır.
 - ▶ Etkin (“efficient”)
 - ♦ Yazılım, sistem kaynaklarının israfına sebep olmamalıdır.
 - ▶ Kabul-edilebilir (“acceptable”)
 - ♦ Yazılım, kullanıcıları tarafından kabul edilebilmelidir.

PROFESYONEL YAZILIM GELİŞTİRME

- **Profesyonel yazılım** dediğimizde, geliştiricisinden başka kişiler tarafından kullanılması amaçlanmış, genellikle ekiplerce geliştirilen yazılım anlaşılmalıdır. Bu tür yazılım, kullanım süresi (*life-cycle*) boyunca, bakımı yapılarak güncellenir.
- **Yazılım mühendisliği** bireysel yazılım geliştirmeden çok, profesyonel yazılım geliştirmeyi destekler. Yazılımı tanımlama, tasarlama ve evrimleştirme gibi aşamalar ve bunların altında uygulanan teknikleri içerir.

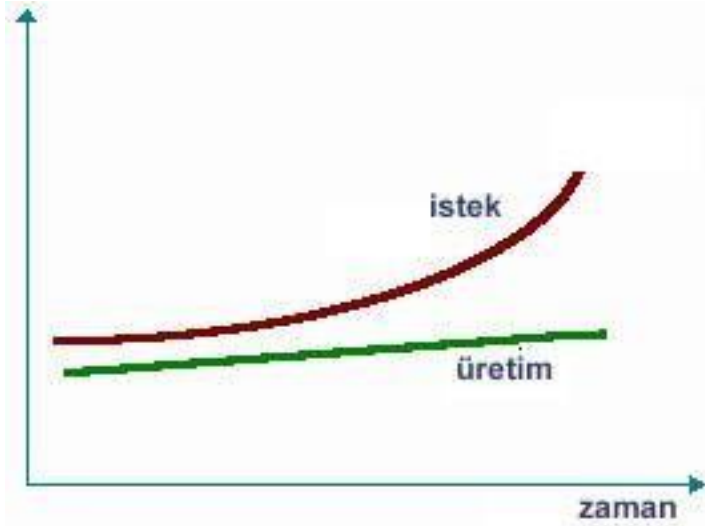
YAZILIM MÜHENDİSLİĞİ - TARİHÇE

- **Yazılım mühendisliği** ifadesi ilk kez, 'yazılım krizi'nin tartışıldığı, 1968 yılında düzenlenen NATO konferansında gündeme gelmiştir. (Naur and Randell, 1969)
- Takip eden 1970 ve 1980'li yıllarda, yazılım mühendisliği için (yapısal programlama, bilgi gizleme, nesneye-yönelik geliştirme, vb.) yeni yöntem ve teknikler önerilmiştir.
- Daha fazla bilgi için;

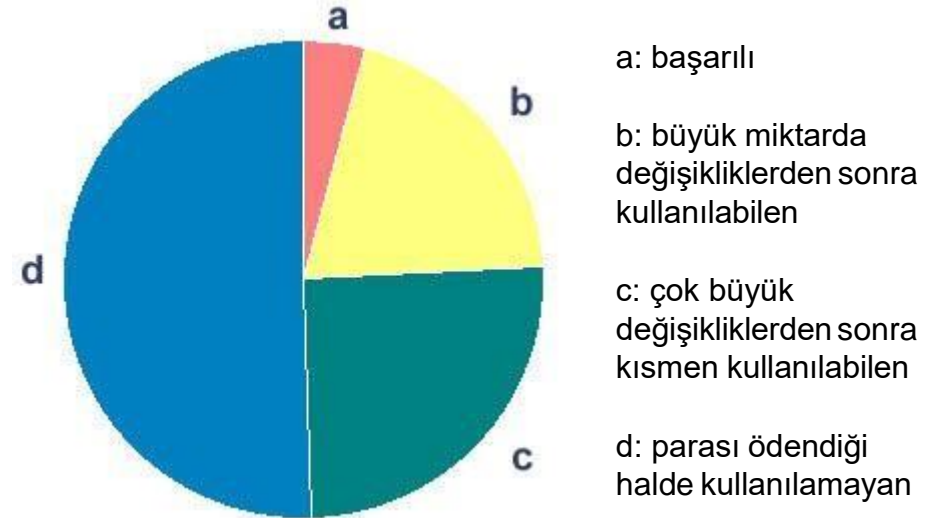
<http://www.SoftwareEngineering-9.com/Web/History/>

YAZILIM KRİZİ

- 1960’lardan itibaren yazılım ürünlerine artan talepler karşısında otomasyonu da içeren değişik çözümler uygulandıysa da yeterli üretim kapasitesine erişilemedi. Bu yetersizlik, “yazılım krizi” söylemiyle dünya literatüründe yerini aldı.
- Yazılım krizinin temel sebebi, üstel olarak artan talebe karşılık doğrusal hızla artış gösteren üretim kapasitesidir. Diğer yandan yazılım projelerindeki başarısızlık oranı da şaşırtıcı derecede yüksek olup bu krizi beslemektedir.

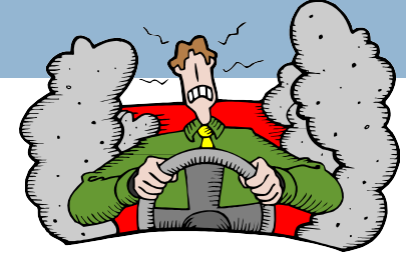


Yazılım ürününe arz ve talep artışları



Toplam maliyete göre yazılım başarısı

BAZI BAŞARISIZLIK HİKAYELERİ



■ Denver havaalanı otomatik bagaj sistemi

- Açılış 2 yıl gecikti
- \$27 milyon maliyet aşımı
- \$360 milyon gecikme maliyeti

■ Hava trafik kontrol (FAA modernizasyon)

- \$5.6 milyar maliyet aşımı
- 8 yıl gecikme
- 4 sistemden 2 tanesi iptal edildi, üçüncü sistemin gereksinimlerinin %52'si karşılandı

■ Comanche Helikopteri

- 10 yılda maliyeti 10 kat arttı, \$34.4 milyon
- Gereksinimleri %74 azaltıldı

ÜRÜN BÜYÜKLÜĞÜ – BAŞARI İLİŞKİSİ

Kestirim (C Satır)	Önce	Zamanında	Gecikme	İptal
13.000	6.06%	74.77%	11.83%	7.33%
130.000	1.24%	60.76%	17.67%	20.33%
1.300.000	0.14%	28.03%	23.83%	48.00%
13.000.000	0.0%	13.67%	21.33%	65.00%

Referans:
“Patterns of Software
Failure and Success”,
C. Jones

**Büyük kapsamlı yazılım ürünleri için,
mühendislik yaklaşımı zorunlu hale gelmiştir!**

MEVCUT DURUM

Cost Overruns	% of Responses
Under 20%	15.5%
21 - 50%	31.5%
51 - 100%	29.6%
101 - 200%	10.2%
201 - 400%	8.8%
Over 400%	4.4%

Time Overruns	% of Responses
Under 20%	13.9%
21 - 50%	18.3%
51 - 100%	20.0%
101 - 200%	35.5%
201 - 400%	11.2%
Over 400%	1.1%

% of Features/Functions	% of Responses
Less Than 25%	4.6%
25 - 49%	27.2%
50 - 74%	21.8%
75 - 99%	39.1%
100%	7.3%

Veri seti: 365 firmadan 8,380 uygulama geliştirme

Referans: Standish Group, CHAOS report, 2014
(<https://www.projectsmart.co.uk/white-papers/chaos-report.pdf>)

BAŞARISIZLIK HİKAYELERİ – TEMEL SEBEPLER

■ *Artan talepler*

- ▶ Hızlı geliştirme ve teslimat
- ▶ Daha büyük ve karmaşık sistemler
- ▶ Daha önce mümkün olmadığı düşünülen özellikler
- ▶ Eskiye geliştirme yaklaşım ve yöntemleri

■ *Düşük beklentiler*

- ▶ Metodoloji kullanmadan programlama kolaylığı
- ▶ Mevcut fakat bilinmeyen/kullanılmayan geliştirme yöntemleri
- ▶ Daha pahalı ve daha az güvenilir yazılım

"Profesyonel Yazılım"

→ Ekip işi !

→ Yazılım mühendisliği eğitimi önemli !!



YAZILIM MÜHENDİSLİĞİ NEDİR?

- Yazılım üretiminin tüm etkinliklerini kapsayan mühendislik disiplini.
- Sistem tanımlama gibi erken aşamalardan sistemin kullanımı esnasındaki bakımına kadar uzanan etkinlikleri kapsar.
- Sadece teknik etkinlikleri değil, aynı zamanda yönetim etkinliklerini de içerir.



MÜHENDİSLİK YAKLAŞIMI !

- Yazılım Mühendisliği; yazılım ürününün geliştirilmesi, işletilmesi ve bakımı, için uygulanan; sistematik, disiplinli ve ölçülebilir yaklaşımdır. [IEEE, 1990]
- Mühendislik, herhangi bir bilim alanındaki bilgi birikimini sistematik olarak pratiğe geçirmeyi hedefler; bilimi ve matematiği kullanır.
 - ◆ Yönetim parametreleri: İşlev, maliyet, zaman
 - ◆ Kalite parametreleri: Dayanıklılık, bakım kolaylığı, güvenlik, kullanım kolaylığı, vb.
- Tekrarlanabilir başarılar için mühendislik yaklaşımı şarttır.
 - ◆ Mühendislik öğretisi ile bir yöntem uygulandığında, benzer sonuçları her zaman elde etme güvenliği vardır.

YAZILIM MÜHENDİSLİĞİ - BAZI TANIMLAR (I)

- The practical application of scientific knowledge in the design and construction of computer programs and the associated documentation to develop, operate and maintain them.” [Boehm, 1976]
- “... the technological and managerial discipline concerned with systematic production and maintenance of software products that are developed and modified on time and within cost estimates” [Fairley, 1985]
- “The application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software” [IEEE, 1990]

YAZILIM MÜHENDİSLİĞİ - BAZI TANIMLAR (II)

- Software engineering is concerned with the definition, refinement and evaluation of principles, methods, techniques and tools to support:
 - ▶ Individual aspects of software development and maintenance (design, coding, etc.)
 - ▶ Planning of software development projects
 - ▶ Performing development, project management and quality assurance activities according to the plan
 - ▶ Assessing the performance of the development and improving products, methods, techniques and tools.

[Rombach and Verlage, 1995]

YAZILIM MÜHENDİSLİĞİ NELERİ KAPSAR?

- Yazılım Gereksinimleri
- Yazılım Tasarımı
- Yazılım Gerçekleme
- Yazılım Test
- Yazılım Bakım
- Yazılım Konfigürasyon Yönetimi
- Yazılım Mühendislik Yönetimi
- Yazılım Mühendisliği Süreci
- Yazılım Mühendisliği Model ve Yöntemleri
- Yazılım Kalitesi
- Yazılım Mühendisliği Profesyonel Pratikleri
- Yazılım Mühendisliği Ekonomisi
- Hesaplama Temelleri
- Matematiksel Temeller
- Mühendislik Temelleri

~Programlama

*Yazılım
Geliştirme
Kapsamı*

[“Guide to the Software Engineering Body of Knowledge”, 2015]

BİLGİSAYAR BİLİMLERİ VE YAZILIM MÜHENDİSLİĞİ ARASINDAKİ İLİŞKİ NEDİR?

- Bilgisayar Bilimleri; bilgisayar ve yazılım sistemlerinin temelinde yatan teoriler ve yöntemlerle ilgilenir.
 - Algoritmalar ve veri yapıları, programlama dilleri, mimari, bilimsel hesaplama, işletim sistemleri, bilgi ve veri yönetimi, grafik görüntüleme ve çoklu-ortam, bilgisayar ağları, akıllı sistemler, vb.
 - (Fizik bilimlerinin, Elektrik ve Elektronik Mühendisliğinin temelinde yattığı gibi)
- Yazılım Mühendisliği; yazılım üretmenin pratik problemleriyle ilgilenir.
 - Bunu yaparken Bilgisayar Bilimlerinin sunduğu kavramsal altyapıyı kullanır.
 - Bir yazılım mühendisinin, ilişkili uygulama alanına göre (örneğin veri-işleme, animasyon, vb.), altta yatan bilgisayar bilimine hakim olması zorunludur.

SİSTEM MÜHENDİSLİĞİ VE YAZILIM MÜHENDİSLİĞİ ARASINDAKİ İLİŞKİ NEDİR?

- Sistem Mühendisliği; yazılımın önemli rol oynadığı, karmaşık bilgisayar sistemlerinin geliştirilmesi ve idamesi ile ilgilenir.
 - ▶ Örnek: ATM sistemi
 - ▶ Sistem Mühendisliği; sistemin genel çatısıyla ilgilidir.
 - ◆ Sistemin kullanılacağı alana ilişkin süreçlerin analiz edilmesi, sistem gereksinimlerinin tanımlanması, sistem mimarisinin oluşturulması, sistem bileşenlerinin tümleştirilmesi gibi aşamaları içerir.
 - ▶ Söz konusu bilgisayar sistemi; donanım, ağ, yazılım bileşenlerinden oluşur.
Sistem Mühendisliği, bileşenlere ilişkin mühendislik etkinliklerinin detaylarıyla pek ilgilenmez.
 - Bu tür sistemler için Yazılım Mühendisliği (kapsamındaki tüm etkinliklerle birlikte),
 - Sistem Mühendisliği altında ve onun bir parçası olarak uygulanır.
 - ▶ Sistem Mühendisliği, Yazılım Mühendisliğine kıyasla çok daha eski bir mühendislik disiplini.

DONANIM MÜHENDİSLİĞİ VE YAZILIM MÜHENDİSLİĞİ ARASINDAKİ FARKLAR NELERDİR?

- Donanım mühendislikleri ile yazılım mühendisliğinin en belirgin farkı ürünlerindedir.
 - ▶ Yazılım ürünü diğer mühendislik ürünlerine oranla daha soyuttur.
 - ▶ Yazılım projesi geliştirme ile sonlanırken donanım projelerinde ek olarak imalat safhası vardır.
 - ▶ Seri üretim, yazılım geliştirme içerisinde neredeyse hiçten ibarettir.
 - ▶ Donanım ürünleri kullanıldıkça aşınır; yazılım ürünlerinde ise aşınma olmaz. Yalnızca baştan beri gizli bulunan hatalar, yazılım kullanıldıkça ortaya çıkar.
- Donanım mühendisliklerinde maliyet odağı seri üretim ve yıpranmayken, yazılım mühendisliğinde maliyet odağı geliştirmedir.

YAZILIM MÜHENDİSLİĞİNDE MALİYETLERİN DAĞILIMI NEDİR?

- Kabaca söylersek, yazılım maliyetinin;
 - %60 : Geliştirme maliyeti,
 - %40 : Test maliyetidir (test maliyetine bulunan hataları düzeltmenin maliyeti de dahildir.)
- Müşteriye özel üretilen yazılımlar için idame (bakım) maliyeti, geliştirme maliyetinin birkaç katına çıkmaktadır

YAZILIM ÜRÜNÜ FARKLI (GİLLİES, 1990)

- Yazılım ürünün fiziksel bir varlığı yok.
- Proje başında müşteriye ve alanı genelde bilmiyoruz.
- Müşteri ihtiyaçları zaman içinde değişiyor.
- Donanım teknolojisi ve dolayısıyla üzerinde çalışacak yazılıma olan ihtiyaç hızla gelişiyor.
- Müşterilerin özellikle değişebilirlikle ilgili yüksek beklentileri var.
- Tasarım kalitesi ve üretim kalitesi genelde ayrılamıyor.

YAZILIM MÜHENDİSLİĞİNDE HEDEFLER

■ Heterojenlik (“heterogeneity”)

- ▶ Yazılım geliştirme için, heterojen platformları ve çalıştırma ortamlarını destekleyecek teknikler geliştirmek

■ Teslim (“delivery”)

- ▶ Yazılımın zamanında teslimi için teknikler geliştirmek

■ Güven (“trust”)

- ▶ Yazılımın kullanıcıları tarafından güvenilebileceğini gösteren teknikler geliştirmek

YAZILIM MÜHENDİSLİĞİNDE ZORLUKLAR

- Yazılım ürünün doğasından kaynaklanan zorluklar
 - ▶ Soyut, değişebilir, insan-duyarlı
- Farklı uygulama alanları, farklı teknolojiler
 - ▶ Geliştirme süreci her zaman izlenebilir değil
- Tasarım ve gerçekleştirme süreçleri tam olarak ayrılamıyor
 - ▶ Bu zorluklarla birlikte, maliyet odağı geliştirmenin kendisi

YAZILIM MÜHENDİSLİĞİNDE PROFESYONEL VE ETİK GEREKLİLİK

- Yazılım mühendisliği, teknik yetkinliğin uygulanmasının yanında, aşağıdaki konularda sorumluluk gerektirir:

- ▶ Gizlilik

- ◆ Örnek: Çalışanların ve müşterilerin gizlilik haklarına saygı göstermek

- ▶ Rekabet

- ◆ Örnek: Kendi yetkinliğiniz dışında iş kabul etmemek

- ▶ Özlük hakları

- ◆ Örnek: Patent, mülkiyet, vb. haklarına dikkat etmek

- ▶ Bilgisayarın amaç-dışı kullanımı

- ◆ Örnek: Başkasına ait bir makinede oyun oynayarak virüs bulaştırmamak

BİR MESLEK OLARAK YAZILIM MÜHENDİSLİĞİ

- Sektörde çalışanların yaklaşık yarısı “Bilgisayar Mühendisliği” lisans eğitime sahip
 - ▶ Alana göre farklılık gösteren lisans dereceleri var
 - ◆ Örneğin; Yönetim Bilgi Sistemleri uygulamaları için İşletme lisansı, gömülü uygulamalar için Elektrik ve Elektronik Mühendisliği lisansı
- Yazılım mühendisliği geniş kesimlerce bir meslek olarak algılanmakta
- Üniversitelerin Yazılım Mühendisliği lisans ve yüksek lisans programları var