

**Boğaziçi Üniversitesi**

**Fen Bilimleri Enstitüsü**

**Bilgisayar Mühendisliği Anabilim Dalı**

**Yazılım Mühendisliği  
Yüksek Lisans Programı**

**Tanıtım ve Program Kuralları  
Kitapçığı**

**<http://www.cmpe.boun.edu.tr/swe/>  
[swe@cmpe.boun.edu.tr](mailto:swe@cmpe.boun.edu.tr)**

**26 Eylül 2008, v3.2**

## İçindekiler

1. Giriş .....	3
2. Yazılım Mühendisliği Yüksek Lisans Programı Hakkında .....	3
3. Başvurular ve Başvuru Koşulları .....	4
3.1. Kimler Başvurabilir? .....	4
3.2. Başvuru Tarihleri .....	4
3.3. Kontenjan .....	5
3.4. Başvuru Koşulları .....	5
3.5. Ön Şartlar ve Ön Şart Dersleri .....	6
3.6. Başvuru Yapılması .....	6
3.7. Mülakatlar ve Başvuruların Değerlendirilmesi .....	6
3.8. Başvurularla ilgili Diğer Sıkça Sorulan Sorular ve Cevapları.....	7
4. Programa Kabul Edilenler İçin Program Kuralları .....	8
4.1. Ön Kayıtlar.....	8
4.2. Derslere Kayıt .....	8
4.3. Ön Şart Derslerinin Alınması .....	9
4.4. Diğer Program Kuralları ve Bazı Prosedürler.....	9
5. Ders Programı .....	11
5.1. Dönemlere Göre Ders Programı (Güz 2007'den itibaren geçerli) ....	11
5.2. Ön Şart, Zorunlu ve Seçmeli Dersler Listesi .....	12
6. Ders Tanımları .....	14

Not: Bu kitapçıkta ki yanlışlık ve eksiklikleri ve kitapçık ile ilgili görüş ve önerilerinizi Yazılım Mühendisliği Yüksek Lisans Programı Koordinatörü Prof. Dr. M. Ufuk Çağlayan'a, [caglayan@boun.edu.tr](mailto:caglayan@boun.edu.tr), Tel: (212) 359 6698 iletebilirsiniz.

# 1. Giriş

Bilişim sektörü bir çok ülke ekonomisinde gelişmenin motoru rolü oynamaktadır. Ülkemizde bilişim sektörü oldukça zayıftır, Ar-Ge ve üretime dönük değildir ve sektörde önemli bir insangücü açığı vardır. Bilişim sektörünün en önemli üretim ve dolayısıyla en önemli gelir alanlarından biri yazılım sistemleridir. Günümüzde yazılım sistemleri, bankalar, sigorta kurumları, sağlık sektörü, şirket yönetimi, telekomünikasyon sistemleri, otomotiv endüstrisi, kara/hava/deniz taşımacılığı gibi hemen hemen tüm alanlarda kullanılan bilgisayar sistemlerinin çok önemli ve kritik bir parçasını oluşturmaktadır. Yazılım Mühendisliği 1968 yılından bu yana var olan yeni bir mühendislik dalıdır ve yazılım sistemlerinin mühendislik prensipleri çerçevesinde tasarımı, üretimi ve işletilmesini hedefler. Bilgisayar ve yazılım sistemleri artık günlük hayatın her alanında yoğun ve etkin bir şekilde kullanılmakta olduğundan, Yazılım Mühendisliği disiplinlerarası uygulamaları olan ve tüm disiplinlerde rol oynayan bir alandır.

Yazılım sistemlerinin geliştirilmesinde ve bu yolla ülke ekonomilerine katkı sağlamada son yıllarda önemli aşamalar kaydetmiş ülkeler arasında Hindistan ve İrlanda, önemli aşamalar kaydetmeyi hedeflemiş ülkeler arasında ise Meksika sayılabilir. Bu ülkelerin yazılım alanında yaptığı atılımlar incelendiğinde, söz konusu atılımların yazılım alanında her seviyede insan gücünün eğitime dayandığı görülmektedir. Bu eğitim seviyeleri, iki yıllık meslek yüksek okulları eğitimi, dört yıllık üniversite/yüksek okul lisans eğitimi, yüksek lisans ve doktora seviyesinde ihtisas eğitimleri olarak tanımlanabilir.

Yazılım sistemlerinin geliştirilmesi emek yoğun bir iştir ve bu alanda başarılı olunması yüksek sayıda ve iyi eğitim almış insan gücünün varlığına bağlıdır. Yazılım alanında insan gücü eğitimi, bilgisayar mühendisliği, bilgisayar bilimleri, yazılım mühendisliği, iletişim ve bilgi teknolojileri, bilgi sistemleri, yönetim bilişim sistemleri gibi bilişimin farklı alanlarında dört yıllık lisans eğitimi almış kişilerin yazılım mühendisliği alanında yüksek lisans seviyesinde ihtisaslaşması şeklinde olabildiği gibi, yazılım alanının disiplinler arası özelliklerinden dolayı, elektrik, makina, inşaat gibi diğer mühendislik alanlarında, fizik, kimya, matematik, biyoloji, jeoloji, psikoloji, sosyoloji gibi temel bilim alanlarında, özet olarak bilişim alanlarından farklı alanlarda dört yıllık lisans eğitimi almış olan kişilerin yazılım mühendisliği alanında eğitilmesi şeklinde de gerçekleştirilebilmektedir. Gerek Türkiye’de gerek diğer ülkelerde, yazılım sistemlerinin geliştirilmesinde rol oynayan insan gücü arz-talep farklılığından dolayı pek çok üniversite mezunu yazılım alanında çalışmak istemektedir.

Yazılım mühendisliği alanında yüksek lisans seviyesinde ihtisas eğitimi vermek ve Türkiye’de bu alandaki insan gücü açığının kapatılmasına katkıda bulunmak amaçlarıyla, **Yazılım Mühendisliği Yüksek Lisans Programı (MS Program in Software Engineering)**, Boğaziçi Üniversitesi, Bilgisayar Mühendisliği Bölümü ve Anabilim Dalı tarafından Ağustos 2003’de başlatılmıştır. **Yazılım Mühendisliği Yüksek Lisans Programı**, tezsiz bir ikinci öğretim (mesai günleri akşamları ve Cumartesi günleri) yüksek lisans programı olarak sunulmaktadır.

## 2. Yazılım Mühendisliği Yüksek Lisans Programı Hakkında

Yazılım Mühendisliği Tezsiz Yüksek Lisans Programı'nın ana hedefi 4 yıllık üniversite/yüksek okul lisans programlarından birini bitiren kişilere yazılım mühendisliği alanında ihtisas eğitimi verilmesidir. Programı başarıyla bitirenlere "**Yazılım Mühendisliği Yüksek Lisans Diploması**" verilmektedir. Yazılım Mühendisliği Yüksek Lisans Programı'nda esas olarak **yazılım proje yönetimi ve yazılım geliştirme** alanlarında ihtisaslaşma amaçlanmaktadır.

**Yazılım Mühendisliği Yüksek Lisans Programı**, üç yarıyıda bitirilecek şekilde planlanmıştır ve ön şart dersleri hariç en fazla altı yarıyıda bitirilmelidir. Dersler, önşart dersleri, zorunlu dersler ve seçmeli dersler olarak üç gruba ayrılmıştır. Yüksek lisans tezi yerine, kişisel ve takım halinde yapılacak projeleri kapsayan uygulama ve proje dersleri vardır. Eğitim dili İngilizce'dir.

Dersler, hafta içi gündüzleri bir iş yerinde çalışanları gözetmek amacıyla, Pazartesi-Salı-Çarşamba-Perşembe-Cuma günleri 19:00-21:50 ve Cumartesi günleri 10:00-12:50, 14:00-16:50 saatleri arasında ve Kuzey Kampüs, Bilgisayar Mühendisliği Binası'nda yapılmaktadır.

**Yazılım Mühendisliği Yüksek Lisans Programı** Güz 2003 döneminde başlamıştır. Haziran 2008 itibariyle mezun sayısı 62'dir.

Öğretim ücreti, her yıl Haziran ayında Boğaziçi Üniversitesi Yönetim Kurulu tarafından ders kredisi başına saptanmakta ve YÖK tarafından onaylanmaktadır. Ön şart dersleri dahil olmak üzere bir yarıyıl alınan derslerin toplam kredisi kredi ücreti ile çarpılarak hesaplanan yarıyıl öğretim ücreti yarıyıl başında üniversitenin belirttiği tarihler aralığında peşin olarak ödenmektedir. Yarıyıl öğretim ücreti için ayrıca bir taksitlendirme yapılmamaktadır.

**Yazılım Mühendisliği Yüksek Lisans Programı** hakkında detaylı ve daha güncel bilgiler <http://www.cmpe.boun.edu.tr/swe/> web adresindedir. [swe@cmpe.boun.edu.tr](mailto:swe@cmpe.boun.edu.tr) e-posta adresine yazarak ek bilgi istenebilir ve olası sorular bu adrese gönderilebilir. Ayrıca, Program Koordinatörü Prof. Dr. M. Ufuk Çağlayan, [caglayan@boun.edu.tr](mailto:caglayan@boun.edu.tr), Tel: (212) 359 6698 veya Program Sekreteri Hülya Torbik, Tel: (212) 359 4508 veya Bilgisayar Mühendisliği Bölümü Sekreterliği, (212) 359 4524 ile de temas kurulabilir.

### 3. Başvurular ve Başvuru Koşulları

#### 3.1. Kimler Başvurabilir?

Yazılım mühendisliği alanında eğitim almayı ve/veya ihtisaslaşmayı arzu eden ve her hangi bir 4 yıllık lisans programını bitirmiş olan adaylar yazılım mühendisliği yüksek lisans programına başvurabilirler.

Programa başvuran bilgisayar mühendisliği mezunları, yazılım proje yönetimi ve yazılım geliştirme alanlarında ihtisaslaşmayı hedeflemektedir. Şimdiye kadar programa başvuran ve programı başarıyla tamamlayan adaylar sadece bilgisayar mühendisliği lisans mezunları değildir. Elektrik/Elektronik, Makina, İnşaat, Kimya, Endüstri, Petrol, Çevre vb diğer mühendislik alanlarında, Bilgisayar Bilimleri, Matematik, Fizik, Kimya, Biyoloji, Jeoloji, Psikoloji, Sosyoloji vb Fen/Edebiyat Fakülteleri alanlarında, Eğitim Fakülteleri alanlarında ve bunlara ek olarak İşletme, Ekonomi, Bilgi Teknolojileri, Bilişim Sistemleri, Yönetim Bilişim Sistemleri, vb alanlarda 4 yıllık lisans programını bitirmiş olan adaylar da yazılım mühendisliği yüksek lisans programına başvurabilmektedir. Bilgisayar mühendisliği dışındaki farklı alanlardan gelen adaylar genelde bilişim/yazılım sektöründe çalışmayı planlayanlar ya da bir süredir bilişim/yazılım sektöründe çalışmakta olmalarına rağmen yeterli bir eğitim almadan sektörde çalıştıklarını düşünenler ya da tekrar eğitim alarak bilişim sektöründe yazılım proje yönetimi ve yazılım geliştirme alanlarında çalışmayı planlayan kişilerdir.

Şimdiye kadar programı bitirenler ve halen öğrenci olanlar arasında, bilgisayar mühendisliği mezunlarına ek olarak Elektrik/Elektronik Mühendisliği, Endüstri Mühendisliği, Makina Mühendisliği, Bilgisayar Bilimleri, Matematik, Fizik, Psikoloji, Bilgi Teknolojileri/Bilişim Sistemleri, İşletme, Ekonomi ve Eğitim Fakültesi lisans mezunları bulunmaktadır.

#### 3.2. Başvuru Tarihleri

Başvurular, programa Güz ya da Bahar yarıyıllarında başlamak için yapılabilir. Güz yarıyılı başvuruları, "şartlı ön kabul" başvurusu olarak Mart ayı sonu-Nisan ayı başında veya normal başvurular olarak Temmuz ayının ikinci yarısındadır. Bahar yarıyılı başvuruları Ocak ayının birinci yarısındadır.

Güz yarıyılı için yapılan Mart-Nisan şartlı ön kabul başvuruları ve Temmuz başvuruları arasında temelde bir fark yoktur, şartlı ön kabul başvuruları daha çok Haziran'da mezun olmayı bekleyenler ve programa kabul edilip edilmediğini daha önceden bilmek isteyenler içindir. Şartlı ön kabul başvuru sahipleri, başarılı bir şekilde mezun olmaları ön şartı ile programa kabul edilebilirler ve mezuniyetlerini takibeden ilk yarıyıl kayıt zamanında programa kayıt olabilirler.

Başvuruların kesin olarak hangi tarihler arasında yapılabileceği, Boğaziçi Üniversitesi web sitesi <http://www.boun.edu.tr/>'de bulunan Boğaziçi Üniversitesi Akademik Takvimi'nde verilmektedir.

### 3.3. Kontenjan

Kontenjan Güz ve Bahar yarıyılları için 35'er kişidir. Ayrıca sadece Güz yarıyılında ve sadece "Advanced" İngilizce seviyesindeki adaylar için 10 kişilik bir İngilizce Hazırlık Okulu kontenjanı vardır.

### 3.4. Başvuru Koşulları

#### Genel Not Ortalaması (GNO):

- Bir üniversite veya yüksek okulun her hangi bir 4 yıllık lisans programından mezuniyet için zaten gerekli olan en az 2.00/4.00 lisans mezuniyet Genel Not Ortalaması (GNO) başvuru için yeterlidir.

#### ALES ve/veya GRE/GMAT:

- ALES Sayısal en az 55 (veya LES Sayısal en az 45) olmalıdır. ALES/LES sınavlarının geçerlilik süresi 2 yıldır. ALES/LES sınavı ile ilgili daha detaylı bilgiler <http://www.osym.gov.tr> web adresindedir.
- ALES Sayısal 55 eşdeğer puanları GRE Sayısal (Quantitative) en az 610 ve GMAT en az 450'dir. GRE ve GMAT sınavlarının geçerlilik süresi 5 yıldır.

#### İngilizce Yeterlilik:

- Boğaziçi Üniversitesi İngilizce Yeterlilik Sınavı (Boğaziçi University English Proficiency Test, BUEPT) veya eşdeğer TOEFL (IBT Test en az 79 / TWE en az 24 veya bilgisayarlı test en az 213 / TWE en az 4.5 veya "paper-based" en az 550 / writing en az 4.5) veya eşdeğer IELTS (en az 7.0) sonucu ile belirtilmelidir.
- İngilizce Yeterlilik konusunda daha detaylı bilgiler [http://www.boun.edu.tr/graduate/organization/admission\\_grad\\_tur.html](http://www.boun.edu.tr/graduate/organization/admission_grad_tur.html) web adresindedir. BUEPT, Boğaziçi Üniversitesi Yabancı Diller Okulu, YADYOK tarafından düzenlenmektedir ve bu sınav ile ilgili detaylı bilgiler <http://www.yadyok.boun.edu.tr/> web adresindedir.
- TOEFL ve BUEPT sınavlarının geçerlilik süresi 2 yıldır. Boğaziçi Üniversitesi, lisansüstü başvurularında ÜDS ve KPDS sonuçlarını İngilizce yeterlilik için kabul etmemektedir.
- Başvurudan sonra İngilizce yeterlilik sınavına girilebilir, İngilizce yeterlilik belgesi esas olarak kayıtlar sırasında gereklidir. Diğer taraftan, başvuranlar arasında değerlendirme yapılırken İngilizce yeterliliği olanların kabul edilmesine öncelik verilir, dolayısıyla mülakatlar sırasında başvuru sahibinin İngilizce yeterlilik belgesi olmasında yarar vardır.
- Güz 2007 Dönemi'nden itibaren, *İLERİ (ADVANCED)* seviyede İngilizce bilgisi olup da İngilizce Yeterliliği geçemeyen adaylar İngilizce Hazırlık sınıfına kabul edilebilmektedir. Seviye tespit sınavı Boğaziçi Üniversitesi Yabancı Diller Okulu, YADYOK tarafından düzenlenmektedir. Intermediate ve Pre- Intermediate seviyede olanlar İngilizce Hazırlık sınıfına kabul edilmemektedir. İngilizce Hazırlık sınıfı sadece Güz dönemlerinde vardır.

#### Referans Mektubu:

- Referans mektubu en az iki adet olmalıdır. Referans mektupları bir öğretim üyesinden ya da kişinin çalıştığı kurumda yetkili bir kişiden alınabilir. Referans Mektup Formu <http://www.cmpe.boun.edu.tr/graduate/msInSwe/recommendation.pdf> web adresindedir.

### 3.5. Ön Şartlar ve Ön Şart Dersleri

Yazılım Mühendisliği Yüksek Lisans Programı'na başvuracak üniversite/yüksek okul mezunlarının **temel bilgisayar okur yazarlığı ile en az bir programlama dili ve temel programlama kavramları bilgisine sahip olması ön şart olarak beklenmektedir.** Temel bilgisayar okur yazarlığı, bilgisayar donanım ve yazılım birimleri hakkında genel bilgi, bir bilgisayar işletim sistemini kurma ve kullanma, yaygın olarak kullanılmakta olan kelime işlemci, tablolu, sunum ve veri tabanı gibi programları kullanma ve Internet erişimi ile Internet servisleri bilgilerini kapsamaktadır. En az bir programlama dili (C, C++, Java, Fortran, vb) ve temel programlama kavramları bilgisi ise Boğaziçi Üniversitesi, Bilgisayar Mühendisliği Bölümü'nde verilmekte olan CmpE 150 Hesaplamaya Giriş dersi eşdeğeridir. Bu ön şartı yerine getiremeyen öğrenciler programa başvuru **yapmamalıdır.**

Yazılım Mühendisliği Yüksek Lisans Programı'na başvuracak öğrencilerin, SWE 501, SWE 510, SWE 514 ve SWE 521 ön şart derslerinin eşdeğerlerini lisans eğitimlerinde ya da bitirmiş oldukları başka bir yüksek lisans programında alıp almadıkları başvurular için önemli değildir. Ön şart dersleri eşdeğerlerinin daha önce alınıp alınmadığına öğrencinin akademik transkript(ler)i ya da temin edeceği diğer belgeler (örneğin sertifika programlarında alınan dersler) bazında karar verilmektedir. Sözkonusu ön şart derslerinin eşdeğerlerini daha önce almamış olan öğrenciler, almamış oldukları ön şart derslerini programda almak şartıyla programa kabul edilebilirler.

### 3.6. Başvuru Yapılması

Başvurular doğrudan, Güney Kampus'da bulunan Boğaziçi Üniversitesi Kayıt İşleri'ne yapılmaktadır. **Başvuru Formu**, başvuru sırasında gereken belgeler, **başvuru ücreti** ve diğer Kayıt İşleri başvuru bilgileri [http://www.boun.edu.tr/graduate/organization/admission\\_grad\\_tur.html](http://www.boun.edu.tr/graduate/organization/admission_grad_tur.html) web adresindedir.

### 3.7. Mülakatlar ve Başvuruların Değerlendirilmesi

Tüm başvuru sahipleri, <http://www.cmpe.boun.edu.tr/swe/> web adresinde ilan edilen tarih, zaman ve yerde yapılan kısa bir mülakata katılmalıdır. Mülakata katılmayan başvuru sahiplerinin başvuruları değerlendirmeye alınmamaktadır. Mülakatlar soyadı sırasında yapılmaktadır. Mülakatlarda genel olarak, başvuru sahibinin aldığı lisans eğitimi, lisans bitirme projesi, varsa stajları, yazılım mühendisliği alanına ilgisi, bu alandaki amaçları, tamamladığı yazılım projeleri ve diğer projeler ve varsa iş tecrübesi ile ilgili sorular sorulmaktadır.

Genel Not Ortalaması, Genel Not Ortalaması'nın hangi üniversite/yüksek okuldan olduğu, LES/ALES/GRE puanı, İngilizce yeterlilik ve bunun seviyesi, referans mektupları içeriği, mülakat sonuçları ve varsa iş tecrübesi programa kabul edilme kriterleri arasındadır. Başvuranlar değerlendirilirken, İngilizce yeterliliği olanların kabul edilmesine öncelik verilir, dolayısıyla mülakatlar sırasında başvuru sahibinin İngilizce yeterlilik belgesi olmasında yarar vardır.

İngilizce yeterlilik belgesi olmayan öğrenciler programa İngilizce şartlı olarak kabul edilebilmektedir. İngilizce şartlı olarak kabul edilen öğrenciler kabul edildikleri dönemin geç kayıt tarihi sonuna kadar İngilizce yeterlilik belgelerini temin ederek kayıt olmalıdırlar, yoksa kayıt haklarını kaybederler. İngilizce şartlı olarak kabul edilen öğrenciler İngilizce yeterlilik belgesi temin edemeyip "Advanced" seviyede oldukları saptanırsa sadece Güz dönemleri için İngilizce Hazırlık Okulu'na kabul edilebilmektedirler.

SWE 501, SWE 510, SWE 514 ve SWE 521 ön şart derslerini almamış olan öğrenciler, almamış oldukları ön şart derslerini programa başladıklarında almak şartıyla programa kabul edilebilmektedirler. Belirli bir ön şart dersi için gerekirse öğrenciye muafiyet sınavı da verilebilmektedir.

Programa kabul edilen öğrenciler, Bilgisayar Mühendisliği Bölüm Kurulu önerisi ve Fen Bilimleri Enstitüsü Yönetim Kurulu kararı ile belirlenmekte ve Fen Bilimleri Enstitüsü ve/veya Kayıt İşleri tarafından başvuru sahiplerine Boğaziçi Üniversitesi web sitesi aracılığı ile duyurulmaktadır. Programa kabul edilen öğrencilerin, kabul edildikleri yarıyıla kayıt olmaları beklenmektedir. Kabul edildikleri yarıyıldaki kayıt olmayan öğrenciler kayıt haklarını kaybedeceklerdir.

### **3.8. Başvurularla ilgili Diğer Sıkça Sorulan Sorular ve Cevapları**

#### **İngilizce Hazırlık Okulu**

İngilizce'leri yeterli olmayan fakat Boğaziçi Üniversitesi, Yabancı Diller Okulu Seviye Tespit Sınavı'nda "Advanced" seviyede olduğu saptanan adaylar sadece Güz dönemleri için doğrudan İngilizce Hazırlık Okulu'na kabul edilebilmektedir. Bu öğrencilerin İngilizce Hazırlık Okulu'nda sadece bir dönem içinde İngilizce'lerini yeterli seviyeye çıkartarak programa başlamaları beklenmektedir. 2. Öğretim Yüksek Lisans Programları için İngilizce Hazırlık Okulu ancak yeterli sayıda öğrenci varsa açılabilir. Son iki dönemde yeterli öğrenci sayısına ulaşamadığı için İngilizce Hazırlık Okulu açılmamıştır, dolayısıyla "Advanced" seviyede olan öğrenciler için İngilizce Hazırlık Okulu açılacağı garantisizdir. İngilizce Hazırlık Okulu dersleri hafta içi akşamları ve Cumartesi günleri yapılmak için planlanmıştır ve farklı bir şekilde ücretlendirilmektedir.

#### **Doktora Devam**

Yazılım Mühendisliği Yüksek Lisans Programı'ndan mezun olanlar, tezsiz bir Yüksek Lisans Programı'ndan mezun olmalarına rağmen, Boğaziçi Üniversitesi, Bilgisayar Mühendisliği Anabilim Dalı'nda ya da arzu ettikleri başka bir üniversitenin ilgili Anabilim Dalı'nda doktora yapabilirler. Güz 2008 Yarıyılı itibariyle, Yazılım Mühendisliği Yüksek Lisans Programı'nın 5 mezunu Boğaziçi Üniversitesi, Bilgisayar Mühendisliği Anabilim Dalı'nda doktora programına başlamıştır. Her hangi bir tezsiz Yüksek Lisans Programı'ndan mezun olanların Boğaziçi Üniversitesi, Bilgisayar Mühendisliği Anabilim Dalı'nda doktora yapmaları için ön şart olarak bir Yüksek Lisans Tez dersine kayıt olmaları ve tez çalışmasını en çok iki akademik yarıyıldaki başarıyla tamamlamaları beklenmektedir. Boğaziçi Üniversitesi, Bilgisayar Mühendisliği Anabilim Dalı doktora programı ile ilgili detaylı bilgiler <http://www.cmpe.boun.edu.tr/graduate/phdInCmpE/> web adresindedir.

#### **Özel Öğrenciler**

Özel sektör ya da kamuda, tercihan bilişim ya da yazılım mühendisliği ilgili bir işte çalışanlar, başka bir üniversitede öğrenci olup olmadıklarına bakılmaksızın Yazılım Mühendisliği Yüksek Lisans Programı'ndan özel öğrenci statüsünde ders alabilirler. Özel öğrenciler, diğer Yazılım Mühendisliği öğrencilerinden beklenen başvuru şartlarını (GNO, ALES, İngilizce Yeterlilik, vb) yerine getirmeli ve aldıkları derslerin kredi ücretini diğer öğrenciler gibi ödemelidir. Aldıkları dersleri başarıyla bitiren özel öğrenciler Kayıt İşleri tarafından bir transkript verilmektedir.

## 4. Programa Kabul Edilenler İçin Program Kuralları

### 4.1. Ön Kayıt

Programa kabul edilen öğrencilerin, Boğaziçi Üniversitesi Akademik Takvimi'nde belirtilen tarih aralığında ön kayıt için gerekli belgeleri temin ederek ve Kayıt Ücreti ödeyerek ön kayıt olmaları beklenmektedir. Ön kayıt sırasında öğrencilere, Kayıt İşleri ÖBİKAS <https://registration.boun.edu.tr/> web sitesine öğrenci olarak erişimleri için bir kullanıcı kimliği, bir şifre ve @boun.edu.tr alan adlı bir e-posta adresi tahsis edilecektir. Bu öğrencilere ayrıca bir akademik danışman atanacaktır. Ön kayıt işlemlerini tamamlayan öğrenciler, ders seçimi, ders eklemesi ve çıkartması gibi kayıt ile ilgili tüm işlemlerini akademik danışmanlarının gözetiminde Boğaziçi Üniversitesi Kayıt İşleri Müdürlüğü ÖBİKAS <https://registration.boun.edu.tr/> web sitesi aracılığı ile yürütmektedir.

Geç kayıt süresi sonuna kadar ön kayıt ve/veya kayıt olmayanlar kayıt haklarını kaybedeceklerdir ve takip eden dönemlerde programa başlamayı arzu ederlerse tekrar başvuru yapmaları gerekecektir.

### 4.2. Derslere Kayıt

Öğrencinin alacağı derslerin seçimi ve derslere kayıtlar, Boğaziçi Üniversitesi Akademik Takvimi'nde belirtilen tarih aralığında, Kayıt İşleri'nin <https://registration.boun.edu.tr/> adresindeki ÖBİKAS Kayıt Sistemi web arayüzünü kullanarak bizzat öğrenci tarafından ve öğrencinin Akademik Danışman'ının onayıyla gerçekleştirilir. Tüm kayıt işlemleri için detaylı bilgi bu web adresindedir.

Bir yarıyıl da hangi Yazılım Mühendisliği derslerinin verileceği, derslerin hangi günler ve saatlerde olduğu, hangi öğretim üyesi tarafından verildiği, <https://registration.boun.edu.tr/> adresinde, General Services menüsü, Schedule altmenüsü altında, önce Akademik Yarıyıl seçilerek (örneğin 2008-2009-1 2008 Güz dönemidir), daha sonra da gelen ekranda Software Engineering seçilerek görülebilir. Gösterilen tabloda M, T, W, Th, F, St işaretleri haftanın Pazartesi-Cumartesi günlerini, 1, 2, 3, ..., 11, 12 ve 13 sayıları ise Boğaziçi Üniversitesi ders saati gösterme usulünde 1 rakamı saat 9:00'dan başlamak üzere ders saatlerini göstermektedir. Örneğin StStSt234 Cumartesi günleri saat 10:00-12:50 arası, TTT111213 ise Salı günleri saat 19:00-21:50 arasını gösterir. Benzer şekilde, önceki yarıyılarda açılmış dersler de farklı bir Akademik Yarıyıl seçilerek görülebilir.

**2008-2009 Akademik Yılı'nda tüm Yazılım Mühendisliği öğrencilerinin Akademik Danışman'ı Doç. Dr. Fatih Alagöz'dür** (Ofis Tel: 359 6652, fatih.alagoz@boun.edu.tr). Kayıt sırasında seçilen dersler ve daha sonra yapılabilecek add/drop işlemleri öğrenci tarafından Akademik Danışman'a muhakkak onay için gönderilmelidir, onay için gönderilmez ve onay alınmazsa derslere kayıt tamamlanmamış olacaktır. Kaydın Akademik Danışman tarafından zamanında onaylanıp onaylanmadığını kontrol etmek öğrencinin sorumluluğundadır. Bu sebeple öğrencilerin "Kayıtsız" kalmamaya itina etmesi gerekmektedir. Öğrencinin seçtiği dersleri akademik danışman onayladıktan sonra, üniversitenin ilgili mali birimi alınan derslerin kredi ücretinin yatırılıp yatırılmadığını kontrol eder, doğru ödeme bazında öğrencinin kaydını onaylar ve mali onaydan sonra öğrencinin kaydı tamamlanmış olur, öğrenci sınıf listelerinde görünür ve öğrenci aldığı dersleri web sisteminde kendi görebilir.

Yazılım Mühendisliği öğrencilerinin her yarıyıl en az 9 kredi (genelde 3 ders) yük alması beklenmektedir, daha az kredi/ders alınmasını gerektiren özel durumlar için öğrencinin Akademik Danışman'ı ile temas kurması gerekmektedir. Son iki yarıyıl da alınması gereken kredi sayısı 12 ise (genelde 4 ders), söz konusu dersler iki yarıyıl da 6+6 kredi (2 ders+2 ders) olarak alınabilir.

Programın en fazla **6 yarıyıl**da bitirilmesi beklenmektedir. Programı bitirmek için Genel Not Ortalaması (GNO) en az 3.00 olmalıdır. F alınan bir dersin hemen ertesi yarıyıl ya da ilk açıldığı yarıyıl da alınması gerekmektedir. Bir dersten arka arkaya iki kere F alınması, öğrencinin programdan

atılmasına sebep olacaktır. Derslerin müfredatta belirtilen ders sırasında alınması gerekmektedir. Akademik Danışman öğrenci kayıtlarını onaylamadan önce öğrencinin ders sırasına uyup uymadığını kontrol edecektir.

### 4.3. Ön Şart Derslerinin Alınması

SWE 501, SWE 510, SWE 514 ve SWE 521 ön şart derslerinin bir bölümü ya da hepsini almak şartıyla Yazılım Mühendisliği Yüksek Lisans Programı'na kabul edilen öğrenciler, almaları gereken ön şart derslerini programın ilk döneminde ve en çok ilk iki dönemde almaları ve bu dersleri başarıyla tamamlamaları gerekmektedir. Bir **ön şart dersinden F alınması** veya **bir yarıyıldan alınan ön şart derslerinin yarıyıl ortalamasının 2.50 altında olması** öğrencinin programdan atılmasına sebep olacaktır. Bir yarıyıldan sadece bir tane ön şart dersi alınması durumunda 2.50 ortalama şartı, alınan tek ön şart dersinden en az CB notu alınmasını gerektirmekte, aksinde öğrenci programdan atılmaktadır.

### 4.4. Diğer Program Kuralları ve Bazı Prosedürler

#### Lisansüstü Giriş Eğitim ve Öğretim Yönetmeliği

Program kurallarının detayları için, [http://www.boun.edu.tr/government/rules/lisansustu\\_index.htm](http://www.boun.edu.tr/government/rules/lisansustu_index.htm) adresindeki **Boğaziçi Üniversitesi Lisansüstü Giriş Eğitim ve Öğretim Yönetmeliği**'nin maddeleri esastır ve **Yönetmelik**'in ilgili bölümleri ve maddeleri tüm öğrenciler tarafından dikkatli bir şekilde okunmalıdır.

#### Öğrencilerle İletişim

Yazılım Mühendisliği öğrencileri ve Akademik Danışman/Program Koordinatörü arasındaki iletişim genelde öğrencinin OBİKAS Kayıt Sistemi'ndeki e-posta aracılığı ile yürütülmektedir. Bu sebeple, tüm öğrencilerin e-posta adreslerini OBİKAS Kayıt Sistemi'nde güncel tutmaları gereklidir. Yazılım Mühendisliği tüm öğrencileri, destek personeli, araştırma görevlileri ve öğretim elemanlarının üye olduğu [swe-all-students@listeci.cmpe.boun.edu.tr](mailto:swe-all-students@listeci.cmpe.boun.edu.tr) e-posta listesi öğrencilere duyuru ve tüm öğrencileri ilgilendiren tartışmalar içindir. Ayrıca her ders için dersin verildiği yarıyıldan geçerli ve dersle ilgili iletişimin yürütüldüğü [swe5xx@listeci.cmpe.boun.edu.tr](mailto:swe5xx@listeci.cmpe.boun.edu.tr) formatında bir ders e-posta listesi bulunmaktadır.

#### Öğrenci Dilekçeleri

Öğrencilerin, Fen Bilimleri Enstitüsü Yönetim Kurulu (FBE YK) kararını gerektiren (örneğin yarıyıl izin talebi, ders eşdeğerliliği, ders kredisi aktarılması, vb) durumları için dilekçe verme hakları saklıdır. Öğrenciler, taleplerini açıklayan, ilgili evraklarla taleplerini destekleyen ve Akademik Danışman ve Koordinatör tarafından görüş belirtilmiş ve onaylanmış bir dilekçe ile FBE Sekreterliği'ne başvurabilir. FBE YK kararının sonucunu takip etmek öğrencinin sorumluluğundadır. Olumlu ya da olumsuz FBE YK kararları Kayıt İşleri'ne iletilerek ÖBİKAS'ta öğrenci ile ilgili alana kayıt edilmektedir.

#### Yarıyıl İzni

Yarıyıl izni alabilmek için öğrencinin ilk önce minimum kredi yükü olarak ve bunun ücretini ödeyerek kayıt olması gerekmektedir. Öğrenci, kaydının onayını takiben, yarıyıl izin talebini açıklayan, bazı evraklarla talebini destekleyen ve Akademik Danışman ve Koordinatör tarafından onaylanmış bir dilekçe ile FBE Sekreterliği'ne başvurabilir. Yarıyıl izni ancak FBE YK kararı ile verilebilir. Askerlik ve sağlık sebepleri (Sağlık Kurulu Raporu) ile yarıyıl izni alınması durumunda yarıyıl izni süreye sayılmamakta, diğer durumlarda yarıyıl izni süreye sayılmaktadır. Örneğin, öğrencinin çalıştığı kurum ya da firma öğrenciyi yarıyılın çoğunu içine alacak bir zaman dilimi için başka bir şehire ya da ülkeye eğitime göndermesi durumunda verilebilecek yarıyıl izni süreye sayılacaktır. Yarıyıl izni, öğrencinin talebi üzerine otomatik olarak verilmemektedir. Öğrencinin yarıyıl izin talebinin sebeplerini detaylı olarak belirtmesi ve belgelemesi gerekmektedir. Öğrenci yarıyıl izni aldığı anda o yarıyıl için ödediği harçlar üniversite tarafından iade edilmemekte ve ilerdeki bir yarıyıldan alacağı dersler için mahsup edilmemektedir.

### **Dersten Çekilme**

Yarıyıl içinde belirli bir dersteki performansını tatmin edici bulmayan bir öğrenci, Akademik Takvim'deki dersten çekilme son tarihine kadar dersten çekilmek için başvuru yapabilir. Dersten çekilen öğrenciye W notu verilir bu not GNO'na dahil edilmez. Öğrenci dersten çekildiğinde o ders için ödediği harç üniversite tarafından iade edilmemekte ve ilerdeki bir yarıyıldaki dersler için mahsup edilmemektedir.

### **Ders Aktarma**

Öğrenci tarafından Boğaziçi Üniversitesi'nde ya da başka bir üniversitede alınmış ve bir diploma için kullanılmamış dersler, Yazılım Mühendisliği Yüksek Lisans Programı'ndaki derslere eşdeğerse, öğrencinin dilekçe ile talebi ve FBE YK kararı ile öğrencinin Yazılım Mühendisliği Yüksek Lisans Programı'ndaki dersleri karşılığı olarak programına aktarılabilir. Aktarılabilecek derslerin toplam kredisi Yazılım Mühendisliği Yüksek Lisans Programı toplam kredisinin (32 kredinin) yarısını (yani 16 krediyi) aşamaz. Aktarılan dersler ve bunların ders notu öğrencinin kaydında özel bir not ile görünür ve aktarılan derslerin notu öğrencinin Yazılım Mühendisliği Yüksek Lisans Programı'ndaki GNO'na dahil edilmez.

### **Seçmeli Dersler**

Yazılım Mühendisliği Yüksek Lisans Programı'nda her yarıyıldaki bir seçmeli ders alınması gerekmektedir. Öğrenciler almak istedikleri bir seçmeli dersi o yarıyıldaki sunulan seçmeli dersler havuzundan kendi ilgi alanları bazında seçebilmektedir. Her yarıyıldaki 3 ila 5 seçmeli ders seçmeli dersler havuzunda bulunmaktadır. Belirli bir yarıyıldaki bir seçmeli dersin açılması, öğrencilerden o seçmeli dersi alabileceklerini gösteren talep gelmesi ve talep edilen seçmeli ders(ler)i o yarıyıldaki verebilecek öğretim üyesi/görevlisi bulunmasına bağlıdır. Seçmeli dersler ancak 6 ya da daha fazla öğrenci kayıt olduğunda açık kalacaktır. 6'dan az öğrencinin kayıt olduğu seçmeli dersler Add/Drop süresi içinde kapatılacak ve kapatılan dersleri alan öğrencilerin diğer seçmeli derslerden birini almaları gerekecektir. Öğrencilerin hangi seçmeli dersleri almayı tercih ettiği yılda bir kere [swe-all-students@listeci.cmpe.boun.edu.tr](mailto:swe-all-students@listeci.cmpe.boun.edu.tr) e-posta listesi aracılığı ile belirlenmektedir. Öğrenciler, diğer 2. Öğretim Yüksek Lisans Programları'ndan, Yazılım Mühendisliği alanı ile ilgili olabilecek bazı dersleri danışmanlarının onayıyla seçmeli ders olarak alabilirler. Bu durumda dersin ücreti, Yazılım Mühendisliği ders kredisi ücreti temelinde hesaplanır.

### **Final Sınavları**

Yazılım Mühendisliği Yüksek Lisans Programı derslerinin finalleri geleneksel olarak finallerin ikinci ya da son haftasında her dersin verildiği günde, saatte ve sınıfta yapılmakta, birinci hafta ise review'lara, eksik yapılan derslere, projelere vs ayrılmaktadır.

### **Yaz Dönemi**

Yaz Dönemi'nde dersi açılması, öğrencilerden bazı dersleri alabileceklerini gösteren talep gelmesi ve talep edilen ders(ler)i Yaz Dönemi'nde verebilecek öğretim üyesi/görevlisi bulunmasına bağlıdır. 2008 Yaz Dönemi'nde bazı dersler ilk defa açılmıştır. Öğrencilerin Yaz Dönemi'nde hangi dersleri almayı tercih ettiği Bahar Yarıyılı'nda [swe-all-students@listeci.cmpe.boun.edu.tr](mailto:swe-all-students@listeci.cmpe.boun.edu.tr) e-posta listesi aracılığı ile belirlenmektedir.

### **Mezuniyet**

Yazılım Mühendisliği Yüksek Lisans Programı'ndan mezun olabilmek için varsa ön şart derslerinden başarıyla geçilmesi ve tüm zorunlu dersler ile seçmeli derslerin toplam kredisi olan 32 kredinin en az 3.00 Genel Not Ortalaması (GNO) ile tamamlanması gerekmektedir. 32 krediyi tamamlayan fakat 3.00 GNO'sını sağlayamayan öğrenciler, 3.00 GNO'sını sağlamak amacıyla daha önceden CC ve/veya CB notu aldıkları derslerini tekrar edebilirler. Program, ön şart dersleri yarıyılı/yarıyılları hariç, süreye sayılan izinler dahil en fazla **6 yarıyıldaki** bitirilmelidir. Programdan Güz ya da Bahar yarıyılı sonunda mezun olunabilir. Güz yarıyılı sonunda mezun olanlar Bahar yarıyılı sonunda yapılan mezuniyet törenine katılıp diplomalarını alırlar.

## 5. Ders Programı

Yazılım mühendisliği programı dersleri, ön şart dersleri, zorunlu dersler ve seçmeli dersler olarak üç grupta toplanmaktadır. Bu üç ders grubundaki dersler ve her dönemde verilecek dersler aşağıda sıralanmıştır. Tüm derslerin İngilizce ve Türkçe tanımları takip eden bölümde verilmiştir.

### 5.1. Dönemlere Göre Ders Programı (Güz 2008'den itibaren geçerli)

#### 1. Dönem

- SWE 513 Principles of Software Engineering (3+0+0) 3
  - SWE 522 Software Requirements Engineering (3+0+0) 3
  - SWE 5xx Elective (3+0+0) 3
  - SWE 577 Directed Studies I (0+4+0) 0
- Toplam dönem kredisi : 9**

#### 2. Dönem

- SWE 523 Managing Software Development I (3+0+0) 3
  - SWE 530 Software Design Process (3+0+0) 3
  - SWE 573 Software Development Practice (2+0+4) 4
  - SWE 5xx Elective (3+0+0) 3
  - SWE 578 Directed Studies II (0+4+0) 0
- Toplam dönem kredisi : 13**

#### 3. Dönem

- SWE 550 Software Quality Assurance and Reliability (3+0+0) 3
  - SWE 574 Software Development as a Team (2+0+4) 4
  - SWE 599 Project (0+3+0) 0
  - SWE 5xx Elective (3+0+0) 3
- Toplam dönem kredisi : 10**  
**Toplam kredi : 32=9+13+10**

#### Not:

- SWE 577 ve 578 Directed Studies I ve II derslerinin ders programının 1. ve 2. Dönemi'ne zorunlu ders olarak eklenmesi Boğaziçi Üniversitesi Senatosu 24.9.2008 tarihli toplantısı ile onaylanmıştır ve bu yeni zorunlu dersler programa Güz 2008 yarıyılı ve daha sonra kabul edilen öğrenciler için geçerlidir.

## 5.2. Ön Şart, Zorunlu ve Seçmeli Dersler Listesi

### Ön Şart Dersleri (4 ders)

1. SWE 501 Introduction to Object Oriented Programming	(3+0+0) 3
2. SWE 510 Data Structures and Algorithms	(3+0+0) 3
3. SWE 514 Computer Systems	(3+0+0) 3
4. SWE 521 Database Systems	(3+0+0) 3

### Zorunlu Dersler (8+2 ders)

1. SWE 513 Principles of Software Engineering	(3+0+0) 3
2. SWE 522 Software Requirements Engineering	(3+0+0) 3
3. SWE 523 Managing Software Development I	(3+0+0) 3
4. SWE 530 Software Design Process	(3+0+0) 3
5. SWE 550 Software Quality Assurance and Reliability	(3+0+0) 3
6. SWE 573 Software Development Practice	(2+0+4) 4
7. SWE 574 Software Development as a Team	(2+0+4) 4
8. SWE 577 Directed Studies I	(0+4+0) 0
9. SWE 578 Directed Studies II	(0+4+0) 0
10. SWE 599 Project	(0+3+0) 0

### Seçmeli Dersler (Şimdiye kadar en az bir kere verilmiş olan seçmeli dersler koyu renktir)

<b>1. SWE 511 Computer Architecture</b>	<b>(3+0+0) 3</b>
<b>2. SWE 512 Operating Systems</b>	<b>(3+0+0) 3</b>
<b>3. SWE 520 Computer Networks</b>	<b>(3+0+0) 3</b>
<b>4. SWE 531 Managing Software Development II</b>	<b>(3+0+0) 3</b>
5. SWE 540 Multimedia and Web	(3+0+0) 3
6. SWE 541 Electronic Commerce	(3+0+0) 3
7. SWE 542 Advanced Software Engineering	(3+0+0) 3
8. SWE 543 Decision Support Systems	(3+0+0) 3
<b>9. SWE 544 Internet Programming</b>	<b>(3+0+0) 3</b>
<b>10. SWE 545 Distributed Systems Programming</b>	<b>(3+0+0) 3</b>
<b>11. SWE 546 Data Mining</b>	<b>(3+0+0) 3</b>
<b>12. SWE 547 Human Computer Interaction</b>	<b>(3+0+0) 3</b>
13. SWE 548 High Performance Computing	(3+0+0) 3
14. SWE 549 Systems Software and Programming	(3+0+0) 3
15. SWE 551 Lightweight Client Programming	(3+0+0) 3
<b>16. SWE 552 Telecommunications Software Engineering</b>	<b>(3+0+0) 3</b>
17. SWE 553 Embedded Systems	(3+0+0) 3
18. SWE 554 CAD/CAM Software Development	(3+0+0) 3
<b>19. SWE 555 Artificial Intelligence Techniques</b>	<b>(3+0+0) 3</b>
<b>20. SWE 556 Advanced Database Systems</b>	<b>(3+0+0) 3</b>
<b>21. SWE 571 Software Engineering Project I</b>	<b>(0+0+3) 0</b>
<b>22. SWE 572 Software Engineering Project II</b>	<b>(0+0+3) 0</b>
23. SWE 575 Case Studies in Software Engineering I	(0+4+0) 0
24. SWE 576 Case Studies in Software Engineering II	(0+4+0) 0
25. SWE 580-595 Special Topics in Software Engineering	(3+0+0) 3
<b><u>Şimdiye kadar tanımlanmış ve verilmiş“Special Topics” dersler</u></b>	
<b>26. SWE 592 Advanced Database Systems</b>	<b>(3+0+0) 3</b>
<b>27. SWE 594 Multi-Core Programming</b>	<b>(3+0+0) 3</b>
<b>28. SWE 595 Secure Software Development</b>	<b>(3+0+0) 3</b>
<b>29. SWE 596 Patterns in Software Engineering</b>	<b>(3+0+0) 3</b>
<b><u>Tanımlanmış fakat verilmemiş“Special Topics” dersler</u></b>	
30. SWE 591 Web Services	(3+0+0) 3
31. SWE 593 Formal Methods in Software Verification and Validation	(3+0+0) 3

## 6. Ders Tanımları

### **SWE 501 Introduction To Object Oriented Programming (Nesneye Yönelik Programlamaya Giriş)**

(3+0+0) 3

Programming methodology: Specification, design, coding, program correctness. Review of data types: scalar types, structured types. Data Structures: Linear structures (stacks, queues, linear lists), nonlinear structures (trees, sets), files (sequential, random access). Implementation of data structures: Linked list implementation, multidimensional array implementation, record implementation, character string, stack, queue, tree and set implementations. Recursion.

Programlama yöntemleri. Belirtim, tasarım, kodlama, program doğruluğu. Veri tiplerinin gözden geçirilmesi, sayıl tipler, yapıli tipler. Veri yapıları: Doğrusal yapılar( yığıt, kuyruk, doğrusal listeler), doğrusal olmayan yapılar (ağaçlar, kümeler), dosyalar( sıradan, rasgele erişimli). Veri yapılarının gerçekleştirilmesi. Bağlı liste gerçekleştirilmesi, çok boyutlu dizi gerçekleştirilmesi, kayıt gerçekleştirilmesi, karakter katarı, kuyruk, ağaç ve küme uygulamaları. Özyineleme.

### **SWE 510 Data Structures and Algorithms (Veri Yapıları ve Algoritmalar)**

(3+0+0) 3

Specification, usage and implementation and analysis of advanced data structures and algorithms. Hashing, heap structures, advanced sorting techniques, graphs and algorithm design techniques.

İleri veri yapılarının ve algoritmalarının kullanımı, gerçekleşmesi ve analizi. Kurs, formül ile veriye ulaşma, yığın yapıları, ileri sıralama teknikleri, çizgeler ve algoritma tasarım teknikleri.

### **SWE 511 Computer Architecture (Bilgisayar Mimarisi)**

(3+0+0) 3

Hierarchical treatment of a computer system. Architectural elements, classification of computer architectures. Assembly level machine organization, instruction sets, address modes. CPU organization, local memory, data operators, interconnection topology. Memory hierarchy, cache memory, virtual memory, associative memory, memory management. I/O organization. Limitations of Von Neumann Architecture. RISC VS CISC. Pipelined processors. Multiprocessor architectures.

Bilgisayar sistemleri hiyerarşik yapısı. Bilgisayar mimari elemanları, bilgisayar mimarilerinin sınıflandırılması. Bilgisayar yapısının “derleme” düzeyinde irdelenmesi, komut kümeleri ,adresleme türleri. İşlemci yapısı, ayrılmış bellek, sanal bellek, bellek yönetim mekanizması. Bilgisayar giriş/çıkış yapıları. Von Neumann mimarisinin kısıtlı yapısı. RISC ve CISC işlemci yapılarının karşılaştırılması. Boruhatlı işlemciler. Dizi işlemcileri. Çok işlemcili sistem mimarileri.

### **SWE 512 Operating Systems (İşletim Sistemleri)**

(3+0+0) 3

Evolution of operating systems. Multi-programming and time sharing concepts. User and program interfaces. Concurrent processes, CPU scheduling, process synchronization, critical section problem. Deadlock prevention, avoidance, detection and recovery. Memory management, swapping, multiple partitions. Paging, segmentation, virtual memory, page replacement algorithms. File system structures, allocation methods, directory implementation. I/O interfaces, secondary storage structure. Protection and Security: Access matrix and rights, capabilities, security issues.

İşletim Sistemlerinin Evrimi. Çoklu programlama ve zaman paylaşırma. Kullanıcı ve program arayüzleri. Eşzamanlı işlemler, CPU zamanlaması, süreç eşzamanlaması, kritik bölüm problemi. Ölümcül kilitlenmenin önlenmesi, kaçınılması, belirlenmesi ve kurtulma. Bellek yönetimi, çoklu bölmeleme. Sayfalandırma, bölümlendirme, sanal bellek, sayfa yenileme algoritmaları. Dosya sistemi yapıları, yer alma yöntemleri, girdi/çıkıı arayüzleri, ikincil depolama yapıları. Koruma ve güvenlik: Ulaşım hakları, yetkileri, güvenlik konuları.

### **SWE 513 Principles of Software Engineering (Yazılım Mühendisliğinin Temelleri)**

(3+0+0) 3

Introductory concepts in software engineering covering an overview of topics in software life cycle, project planning, software management, requirements capture and analysis, human factors, functional specification, software architecture, design methods, programming for reliability and maintainability, team programming, testing methods, configuration management, system delivery and maintenance, process and product evaluation and improvement and project documentation.

Yazılım mühendisliğine giriş kavramları. Yazılım hayat döngüsü, proje planlaması, yazılım yönetimi, kullanıcı istekleri toplama ve analizi, insan faktörü, işlevsel özellikler, yazılım mimarisi, tasarım metodları, güvenilirlik ve bakım için programlama, programlamada takım çalışması, program test yöntemleri, program teslimi ve bakımı, programlama süreçleri ve ürün değerlendirmesi, süreç ve ürün mükemmeliyeti, proje belgelendirilmesi.

### **SWE 514 Computer Systems (Bilgisayar Sistemleri) (3+0+0) 3**

Classification of computer architectures. Computer organization, instruction sets, address modes. Processor and memory structures. I/O organization. Limitations of Von Neumann Architecture. RISC and CISC processors. Pipelined processors. Multiprocessor architectures. Multi-programming and time sharing concepts. User and program interfaces. Concurrent processes, processor scheduling, process synchronization, critical section problem. Deadlocks. Memory management. Multiple partitions, paging, segmentation, virtual memory. File and directory system structures. Secondary storage structures.

Bilgisayar mimarilerinin sınıflandırılması. Bilgisayar yapısı, komut kümeleri ,adresleme türleri. İşlemci ve bellek yapıları. Giriş/çıkış düzenekleri. Von Neumann mimarisinin kısıtları. RISC ve CISC işlemciler. Boruhatlı işlemciler. Çok işlemcili mimariler. Çoklu programlama ve zaman paylaşırma. Kullanıcı ve program arayüzleri. Eşzamanlı işlemler, işlemci zamanlaması, süreç eşzamanlaması, kritik bölüm problemi. Ölümcül kilitleme. Bellek yönetimi. Çoklu bölmeleme, sayfalama, bölümlendirme, sanal bellek. Dosya ve dizin sistemi yapıları. İkincil depolama yapıları.

### **SWE 520 Computer Networks (Bilgisayar Ağları) (3+0+0) 3**

Network layers and architectures, reference models. Physical layer and data communications issues. Data link layer issues, sliding window protocols. Local area networks, Ethernet, bridges and switches. Network layer issues, routing and congestion control, Internet Protocol, routing protocols. Transport layer services and protocols. TCP and UDP. Network programming. Application layer issues and protocols.

Ağ katmanı ve mimarileri, dayanak modelleri. Fiziksel katman ve veri iletişimi konuları. Veri bağı katmanı konuları, kayan pencere protokolları. Yerel ağlar, Ethernet, köprü ve anahtarlar. Ağ katmanı konuları, yönlendirme ve tıkanıklık denetimi, Internet Protokolü, yönlendirme protokolları. Taşıma katmanı servisleri ve protokolları. TCP ve UDP. Ağ programlama. Uygulama katmanı konuları ve protokolları.

### **SWE 521 Database Systems (Veritabanı Sistemleri) (3+0+0) 3**

Practical and intuitive understanding of databases and database management systems. The relational model. Other important data models. SQL as a standard query language for the relational model. Design and use of data base management systems.

Veritabanı ve veritabanı yönetim sistemleriyle ilgili temel konular. Bağıntısal veritabanı modeli ve diğer önemli veri modelleri. Bağıntısal veritabanılarda standart sorgulama dili olarak SQL. Veritabanı yönetim sistemlerinin kullanımı ve tasarımı.

### **SWE 522 Software Requirements Engineering (Yazılım İsterleri Mühendisliği) (3+0+0) 3**

System and software requirements engineering. Identification, elicitation, modeling, analysis, specification, management, and evolution of functional and non-functional requirements. Interactions

and trade-offs among hardware, software, and organization. Transition from requirements to design. Critical issues in requirements engineering.

Sistem ve yazılım istekleri mühendisliği, İşlevsel ve işlevsel olmayan isteklerin tanımlanması, ortaya çıkarılması, modellenmesi, çözümlenmesi, belirtimi, yönetimi ve evrimi. Donanım, yazılım ve örgütlenme arasındaki ödünleşim ve etkileşimler. Gerekirlerden tasarıma geçiş. İstekler mühendisliğinde kritik konular.

**SWE 523 Managing Software Development I (Yazılım Geliştirme Yönetimi I) (3+0+0) 3**

Overview of general project management issues. Specialized management techniques unique to nature of software development. Fundamental issues of software project management. Project personnel and organization. Software size, development duration, cost, budgeting and schedule estimation.

Proje yönetimi konularına genel bakış. Yazılım geliştirme alanına özgü yönetim teknikleri. Yazılım projesi yönetimi konularının temelleri. Proje personeli ve organizasyonu. Yazılım büyüklüğü, geliştirme süresi, maliyet, bütçeleme ve zamanlama kestirimleri.

**SWE 530 Software Design Process (Yazılım Tasarımı Süreci) (3+0+0) 3**

Representations of design and architecture. Software architectures and design plans. Design methods. Design state assessment. Design quality assurance. Design verification. Group design and implementation of an application.

Tasarım ve mimarinin gösterimleri. Yazılım mimarileri ve tasarım planları. Tasarım yöntemleri. Tasarım durum değerlendirmesi. Tasarım nitelik güvencesi. Tasarım doğrulaması. Grup halinde bir uygulamanın tasarımı ve gerçekleştirilmesi.

**SWE 531 Managing Software Development II (Yazılım Geliştirme Yönetimi II) (3+0+0) 3**

Detailed project planning and scheduling in software development. Software process management. Software configuration management. Public domain and commercial tools for software project management.

**Prerequisite: SWE 523**

Yazılım geliştirmede ayrıntılı proje planlama ve zamanlama. Yazılım süreçleri yönetimi. Yazılım yapılış yönetimi. Yazılım projesi yönetimi için açık ve ticari araçlar.

**Önkoşul: SWE 523**

**SWE 540 Multimedia and Web (Çokluortam ve Web) (3+0+0) 3**

Types of multimedia information: text, graphics, hypertext, sound, images, video and animation. Formats and standards of multimedia information: MPEG. The WWW as a medium for multimedia data. Low, medium and high level web page design. HTML, DHTML, Javascript and Java.

Çokluortam bilgi tipleri: metin, grafik, yardımcı metin, ses, imgeler, video ve canlandırma. Çokluortam bilgi biçim ve standartları: MPEG. Çokluortam veri sunumu için WWW. Alt, orta ve yüksek düzeyde web sayfası tasarımı. HTML, DHTML, Javascript ve Java.

**SWE 541 Electronic Commerce (Elektronik Ticaret) (3+0+0) 3**

A critical review and analysis of electronic commerce, with emphasis on multidisciplinary aspects and team projects. Technology infrastructure for electronic commerce, examples of web-based businesses, economics and business models of the Internet, Internet security, social, legal, ethical and public policy issues of the Internet and a business plan creation for a web-based operation.

**Prerequisites: SWE 544**

Elektronik ticarete disiplinlerarası bakış ve takım projeleri. Elektronik ticaretin teknoloji alt yapısı, web tabanlı işletme örnekleri, İnternet ekonomisi, iş ve teknoloji modelleri, İnternette güvenlik, sosyal, yasal ve iş ahlaki ile ilgili problemler. Web tabanlı işletme için iş planı oluşturulması.

**Önkoşul: SWE 544****SWE 542 Advanced Software Engineering (İleri Yazılım Mühendisliği) (3+0+0) 3**

Study of specific advanced topics in software engineering. Program construction and verification, software testing and reliability, advances in software development tools.

**Prerequisites: SWE 513**

İleri yazılım mühendisliği konularının incelenmesi. Program geliştirilmesi ve doğrulanması, yazılım test etme ve güvenilirliği, yazılım geliştirme araçlarında ilerlemeler.

**Önkoşul: SWE 513****SWE 543 Decision Support Systems (Karar Destek Sistemleri) (3+0+0) 3**

Supporting business decision-making. Gaining competitive advantage with Decision Support Systems (DSS). Analyzing business decision processes. Designing and developing decision support systems. Designing and evaluating DSS user interfaces. Understanding DSS architecture. Networking and security issues. Evaluating decision support system projects.

İşletmelerde karar almayı destekleme, karar destek sistemleri ile rekabet avantajını artırma, işletmelerde karar alma süreçlerinin analizi, karar destek sistemlerinin tasarımı ve programlanması, önyüz tasarımı, karar destek sistemlerinin alt yapı mimarisinin tasarımı ve incelenmesi, ağ ve güvenlik problemleri ve karar destek sistemleri projelerinin değerlendirilmesi.

**SWE 544 Internet Programming (İnternet Programlama) (3+0+0) 3**

Programming in client and server side. Web-application-data base server architectures. Web servers, servlets, applets. Peer-to-peer applications. Network programming.

İstemci-sunucu taraflarında programlama, Web-uygulama-veri tabanı sunuculu 3 katmanlı yazılım mimarileri, web sunucular, servlet ve applet'ler. Akrandan-akrana uygulamalar, ağ programlama.

**SWE 545 Distributed Systems Programming (Dağıtık Sistem Programlama) (3+0+0) 3**

TCP/IP basics. Inter-process communication. Threads and condition variables. Sockets – Berkeley sockets. Transport Layer Interface (TLI) and STREAMS library. Remote Procedure Call (RPC). RMI and CORBA. XML and SOAP, Web Services. Mobile Agents

TCP/IP temelleri. Süreçler arası haberleşme. Süreç parçacıkları ve koşullama değişkenleri. Soketler – Berkeley soketleri. Taşıma Katmanı Arayüzü (TLI) ve STREAMS kütüphanesi. Uzaktan Yordam Çağırma (RPC). RMI ve CORBA. XML ve SOAP, ağ servisleri. Hareketli ajanlar.

**SWE 546 Data Mining (Veri Madenciliği) (3+0+0) 3**

Databases, data warehouses, and data mining. Data preprocessing and cleansing. Association mining. Feature extraction, Classification and regression. Case studies: Basket analysis and credit risk scoring.

Veritabanları, veri ambarları ve veri madenciliği. Veri önışleme ve temizleme. Bağntı öğrenme. Öznitelik çıkarma, sınıflandırma ve regresyon. Örnek çalışmalar: Sepet analizi ve kredi risk puanı hesaplama.

**SWE 547 Human Computer Interaction (İnsan Bilgisayar İletişimi) (3+0+0) 3**

An exploration of the languages, techniques and mechanisms used to define and enhance communication between people and computer applications, both for input and output, in the general case and for a variety of important special domains. Use of graphics in the interface; multimedia

environments; alternative I/O devices; issues in interface design, and user-centered design and task analysis. Overview of psychological, physiological, linguistic, and perceptual factors. Advantages and disadvantages of various interaction techniques. Command language syntax and data presentations.

İnsanlarla bilgisayarlar arasında iletişimi tanımlayan programlama dilleri, teknikler ve mekanizmaların gözden geçirilmesi. Önyüzde grafik kullanımı, çoklu sistemler, seçenekli donanımlar, önyüz tasarımlarında problemler, kullanıcı odaklı tasarımlar ve analizler. Psikolojik, fizyolojik ve algılamaya dayalı etkenler. Kullanılan tekniklerin üstünlük ve zayıflıkları, komut lisanları ve veri sunumları.

**SWE 548 High Performance Computing  
(Yüksek Başarımlı Hesaplama)**

**(3+0+0) 3**

High performance computer architectures: Multiprocessors, multicomputers, Flynn's taxonomy, SIMD, MIMD, SPMD, interconnection topologies, Beowulf clusters. Cache optimization Techniques. Vectorization. Parallel machine and performance models: PRAM, speedup, work efficiency, scalability. Parallel programming with message passing and multi-threading libraries. Parallel applications. Load balancing and graph partitioning.

Yüksek başarımlı bilgisayar mimarileri: Çok-işlemciler, çok-bilgisayarlılar, Flynn mimari sınıfları, SIMD, MIMD, SPMD, bağlama topolojileri, öbek bilgisayarları. Önbellek kullanımı eniyileme teknikleri. Vektörleştirme. Koşut makine ve başarımlı modelleri: PRAM, hızlanma, çalışma etkenliği, ölçeklenebilme. Mesaj yollama ve çoklu-sicim kütüphaneleri ile paralel programlama. Koşut uygulamalar. Yük dengeleme ve çizge parçalama.

**SWE 549 System Software and Programming  
(Sistem Yazılımı ve Programlama)**

**(3+0+0) 3**

Overview of compilers, interpreters, assemblers, linkers and loaders. Unix environment and administration. System calls. Signals and exceptions. Localization and Unicode. Perl and CGI programming. Assembly language programming.

Derleyicilere, yorumlayıcılara, çeviricilere, bağlayıcılara ve yükleyicilere bakış. Unix ortamı ve yönetimi. Sistem çağruları. Sinyaller ve istisnalar. Yöreselleştirme ve Unicode. Perl ve CGI programlama. Çevirici programlama.

**SWE 550 Software Quality Assurance and Reliability  
(Yazılım Kalite Güvencesi ve Güvenilirliği)**

**(3+0+0) 3**

Processes and methods for developing, assessing and maintaining quality software. Quality factors, reliability, robustness etc. Spftware quality assurance via verification, testing and fault-tolerance. Software quality standarts and enforcement processes. Software verification vs validation. Goals of SW testing : defect detection & removal and assurance of quality. Classification and details of software testing procedures Problems and policies in optimum SW release time. Fault-tolerant software design. Software reliability modeling.

Nitelikli yazılım geliştirme, güvenceleme ve bakımı ile ilgili yöntemler ve süreçler. Güvenilirlik, gürbüzlük vb nitelik etkenleri. Sağlama, test ve hataya-dayanıklılık yolu ile kalite güvenceleme. Yazılım kalite standartları ve uygulanması. Yazılımın "doğrulanması" ve "geçerlilik saptanması" yöntemlerinin karşılaştırılması. Kalite göstergeleme ya da hata bulup düzeltme amaçlı yazılım testi. Yazılım test yöntemleri ve ayrıntıları. En uygun test sonlandırma ve yazılım teslim zamanı ile ilgili problem ve politikalar. Hataya dayanıklı yazılım tasarımı. Yazılım güvenilirlik modellemesi.

**SWE 551 Lightweight Client Programming (Güçsüz İstemci Programlama)**

**(3+0+0) 3**

Examples of lightweight clients. The Java 2 Platform, Micro Edition (J2ME) and standard SML-based markup languages. Mobile information device profile (MIDP) and Connected Limited Device (CLCD) structures. CLCD and MIDP APIs. J2ME development tools. Technologies used to connect

mobile devices to consumer and enterprise services: Standard data formats, security and application layer protocols.

Güçsüz istemci örnekleri. Java 2 Micro Edition (J2ME) ve standart SML temelli diller. Gezgin Bilişim Cihaz Profili (MIDP), Bağlanmış Sınırlı Cihaz (CLCD) yapıları. CLCD ve MIDP uygulama geliştirme arayüzleri. J2ME geliştirme araçları. Bilgi sistemlerine gezgin araçlardan ulaşım, standart veri biçimleri, güvenlik, uygulama katmanı protokolları.

**SWE 552 Telecommunications Software Engineering (3+0+0) 3**  
**(Telekomünikasyon Yazılımı Mühendisliği)**

Telecommunications software specifics. OSI framework and standardization, protocol stack and layering, typical mechanisms: error and flow control, recovery. Protocol modeling, formal specification techniques, modeling languages SDL, MSC, LOTOS, ASN.1. Design of protocol stacks and layer interfaces. Typical software structures. Production process of telecommunications software. Importance of conformed design and testing. TTCN test specification language. Design of test cases. Testing tools and environments.

Haberleşme yazılımı özellikleri. OSI yapısı ve standardlaşma, protokol yığıtları ve katmanlar, temel mekanizmalar: hata ve akış kontrolü, hatadan kurtarma. Protokol modelleme, öznellik teknikleri, modelleme dilleri SDL, MSC, LOTOS ve ASN.1. Protokol yığıtı tasarımı ve katman arayüzleri. Tipik yazılım mimarileri. Haberleşme yazılımı üretim süreçleri. Standartlarla uyumlu tasarım ve test. TTCN test tanımlama dili. Test senaryosu tasarımı. Test araçları ve ortamı.

**SWE 553 Embedded Systems (Gömülü Sistemler) (3+0+0) 3**

Definition of embedded systems. Embedded processors, memory systems, basic peripherals, interfacing to the analogue world. Interrupts and exceptions. Real-time operating systems. Writing software for embedded systems. Emulation and debugging techniques. Buffering and other data structures. Memory and performance trade-offs. Software and design examples.

Gömülü sistemlerin tanımı. Gömülü işlemciler, hafıza sistemleri, temel dış birimler, analog dünya arayüzleri. Kesmeler ve istisnai durumlar. Gerçek zamanlı işletim sistemleri. Gömülü sistemler için yazılım. Benzetim ve hata ayıklama teknikleri. Tamponlama ve diğer veri yapıları. Bellek ve başarımlar arası ilişkiler. Yazılım ve tasarım örnekleri.

**SWE 554 CAD/CAM Software Development (3+0+0) 3**  
**(Bilgisayar Destekli Tasarım/ Bilgisayar Destekli Üretim (BDT/BDÜ) Yazılımı Geliştirme)**

Geometric modeling and computer graphics techniques that are used in CAD/CAM systems. Standards for CAD/CAM. Survey of CAD/CAM software. Computer aided engineering analysis. Manufacturing processes and machine tools. Computer numerical control. Robot technology and applications. Group technology and computer-aided process planning. Manufacturing systems.

BDT/BDÜ sistemlerinde kullanılan geometrik modelleme ve bilgisayar grafik teknikleri. BDT/BDÜ standartları. BDT/BDÜ yazılımlarının incelenmesi. Bilgisayar destekli mühendislik analizi. Üretim işlemleri ve makine araçları. Bilgisayarlı sayısal denetim. Robot teknolojisi ve uygulamaları. Grup teknolojisi ve bilgisayar destekli işlem planlama. Üretim sistemleri.

**SWE 555 Artificial Intelligence Techniques (Yapay Zeka Teknikleri) (3+0+0) 3**

Representation of knowledge. Search and heuristic programming. Logic and logic programming. Application areas of artificial intelligence: Problem solving, games and puzzles, expert systems, planning, learning, qualitative reasoning, and natural language understanding. An artificial intelligence language.

Bilgi gösterimi. Arama ve buluşsal programlama. Yapay zekanın uygulama alanları: Problem çözme, oyunlar ve bilmeceler, uzman sistemler, planlama, öğrenme, nitel uslamlama ve doğal dil anlama. Bir yapay zeka dili.

**SWE 556 Advanced Database Systems (İleri Veritabanı Sistemleri) (3+0+0) 3**

Examination of recent topics and advancements related to databases, database management systems and their applications: Extensible Markup Language (XML), deductive databases, object oriented databases, query processing, indexing, storage structures, concurrency control, security, data warehouses, medical databases, database design.

Veritabanları, veritabanı yönetim sistemleri ve veritabanı uygulamalarında yeni gelişmeler: Genişletilebilir Bağlantılı Metin Dili (Extensible Markup Language, XML), tımdengelimli veritabanları, nesneye yönelik veritabanları, sorgu işleme, dizin, depo yapıları, koşul zamanlılık denetimi, veri anbarları, tıbbi veritabanları, veritabanı tasarımı.

**SWE 571, 572 Software Engineering Project I, II (0+0+3) 0**  
**(Yazılım Mühendisliği Projesi I, II)**

Project under the supervision of a faculty member for developing software for real-life problems. Progress reports, final report and demonstration of developed software.

Öğretim üyesi tarafından yönlendirilen, gerçek hayat problemleri için yazılım geliştirme projesi. Ara gelişme raporları, sonuç raporu ve geliştirilen yazılımın çalışır şekilde gösterilmesi.

**SWE 573 Software Development Practice (2+0+4) 4**  
**(Yazılım Geliştirme Uygulamaları)**

Multi-programmer, real-world, object-oriented, versioned software development using source code management. Integrated development environments, internationalization, maintainable coding, commenting, refactoring, testing, design patterns and best practices.

Çok programcılı, gerçek hayat, nesneye odaklı, kaynak kodu yönetimi kullanılan çok sürümlü yazılım geliştirme. Bütünleşik geliştirme ortamları, uluslararasılaştırma, kolay bakımlı kodlama, yorumlama, yeniden düzenleme, test etme, tasarım örüntüleri ve en iyi uygulamalar.

**SWE 574 Software Development as a Team (2+0+4) 4**  
**(Yazılım Geliştirmede Ekip Çalışması)**

Hands on practice of software development process as a team. Scope, requirements specification, change management, integration and deployment. Tools, methods and techniques for managing the software development process. Use of software development tools. Communication and presentation skills. Development and presentation of software products.

Ekip olarak yazılım geliştirme süreci uygulaması. Kapsam, gereksinimlerin belirlenmesi, değişim yönetimi, bütünleştirme ve kullanıma sunma. Yazılım geliştirme süreci yönetiminde araçlar, yöntemler ve teknikler. Yazılım geliştirme araçları kullanımı. İletişim ve sunum teknikleri. Yazılım ürünleri geliştirilmesi ve sunumu.

**SWE 575 Case Studies in Software Engineering I (0+4+0) 0**  
**(Yazılım Mühendisliği'nde Vaka Çalışmaları I)**

Investigation of software engineering problems through seminars. Teamwork applications and projects on software development problems.

Yazılım mühendisliği problemlerinin seminerler aracılığı ile incelenmesi. Ekip uygulamaları ve yazılım geliştirme problemleri üzerine projeler.

**SWE 576 Case Studies in Software Engineering II (0+4+0) 0**  
**(Yazılım Mühendisliği'nde Vaka Çalışmaları II)**

Continuation of Case Studies in Software Engineering I. Investigation of software engineering problems through site visits and seminars. Teamwork applications and projects on software development problems. Demonstration of results through presentations.

Yazılım mühendisliği problemlerinin ziyaretler ve seminerler aracılığı ile incelenmesi. Ekip uygulamaları ve yazılım geliştirme problemleri üzerine projeler. Sonuçların sunumlarla ortaya konulması.

**SWE 577 Directed Studies I (Yönlendirilmiş Çalışmalar I)** (0+4+0) 0  
Research in the field of software engineering, supervised by faculty.

Öğretim elemanı denetiminde yazılım mühendisliği alanında araştırmalar.

**SWE 578 Directed Studies II (Yönlendirilmiş Çalışmalar II)** (0+4+0) 0  
Continued research in the field of software engineering, supervised by faculty.

Öğretim elemanı denetiminde yazılım mühendisliği alanında devam eden araştırmalar.

**SWE 590-595 Special Topics in Software Engineering** (3+0+0) 3  
**(Yazılım Mühendisliği'nde Özel Konular**  
Special studies on current topics in software engineering.

Yazılım mühendisliğinde güncel konular üzerine özel çalışmalar.

**SWE 599 Project (Proje)** (0+3+0) 0  
Project under the supervision of a faculty member on software development and on various aspects of software systems. Progress report, final report and a presentation of final results.

Öğretim üyesi tarafından yönlendirilen, yazılım geliştirme ve yazılım sistemlerinin değişik yönleri üzerine proje. Ara gelişme raporu, sonuç raporu ve sonuçların sunumu.

**Simdiye kadar tanımlanmış ve verilmiş“Special Topics” dersleri aşağıdadır.**

**SWE 592 Sp. Topics in Software Engineering: Advanced Database Systems** (3+0+0) 3  
**(Yazılım Mühendisliğinde Özel Konular: İleri Veritabanı Sistemleri)**  
(SWE 556 ders kodu ile yukarıdaki listede yer almıştır.)

**SWE 594 Sp. Topics in Software Engineering: Multi-Core Programming** (3+0+0) 3  
**(Yazılım Mühendisliğinde Özel Konular: Çok-Çekirdek Programlama)**

Introduction. to hyper-threading technology and multi-core architectures. Parallel processing concepts. System overview of threading. Windows, POSIX and OpenMP multi-threading application programming interfaces. Message passing programming with MPI. Tools for debugging and performance analysis.

Hiper sicimleme teknolojisi ve çok çekirdekli mimarilere giriş. Paralel işlem yapma kavramları. Windows, POSIX ve OpenMP çok-sicimli uygulama programlama arayüzleri. MPI ile mesaj geçerek programlama. Hata ayıklama ve başarımların analizi için araçlar.

**SWE 595 Sp. Topics in Software Engineering: Secure Software Development** (3+0+0) 3  
**(Yazılım Mühendisliğinde Özel Konular: Güvenli Uygulama Geliştirme)**

Security concepts in software. Information, operating system and network security. Symmetric and public key cryptography, key distribution, hash codes, digital signatures. Authentication protocols. Secure Socket Layer (SSL) application development and SSL API libraries. Secure application requirements, design and implementation.

Yazılımda güvenlik kavramları. Bilgi, işletim sistemi ve ağ güvenliği. Simetrik ve açık anahtar şifreleme, anahtar dağıtımı, kısımlı kodlar, sayısal imzalar. Kimlik kanıtlama protokolleri. Güvenli Soket Katmanı (SSL) uygulama geliştirme ve SSL API kütüphaneleri. Güvenli uygulama gereksinimleri, tasarımı ve gerçekleştirimi.

**SWE 596 Sp. Topics in Software Engineering: Patterns in Software Engineering (3+0+0) 3**  
**(Yazılım Mühendisliğinde Özel Konular: Yazılım Mühendisliğinde Desenler)**

Introduction to patterns. Use of patterns in different fields. Pattern languages. Fundamental requirements of patterns. Information and data entity requirement patterns. User function patterns. Analysis patterns; Accountability, observation and measurement, conversion, object reference, inventory and accounting patterns. Design patterns; Creational, structural, behavioral patterns.

Desenlere giriş. Diğer disiplinlerde desen kullanımı. Desen dilleri. Desenlerin temel istekleri. Bilgi ve veri birim istekleri desenleri. Kullanıcı işlev desenleri. Analiz desenleri; sorumluluk, gözlem ve ölçüm, dönüşüm, nesne referans, envanter ve muhasebe desenleri. Tasarım desenleri; yaratım, yapısal ve davranışsal desenler.

**Tanımlanmış fakat verilmemiş“Special Topics” dersleri aşağıdadır.:**

**SWE 591 Sp. Topics in Software Engineering: Web Services (3+0+0) 3**  
**(Yazılım Mühendisliğinde Özel Konular: Web Servisleri)**

Service-oriented architecture; Standards for Web service description; Web Service Description Language (WSDL); Discovery techniques; Universal Description Discovery and Integration (UDDI); Simple Object Access Protocol (SOAP); Ontologies; Web Ontology Language (OWL); Applications of Web services

Servis tabanlı mimari; Web Servis tanımlamaları için standartlar; Web Servis Tanım Dili (WSDL); Servis bulma teknikleri; Evrensel Tanım Bulma ve Entegrasyon (UDDI); Basit Nesne Ulaşım Protokolü (SOAP); Ontolojiler; Web Ontoloji Dili (OWL); Web servis uygulamaları

**SWE 593 Sp. Topics in Software Engineering: Formal Methods in Software Verification and Validation (3+0+0) 3**  
**(Yazılım Mühendisliğinde Özel Konular: Yazılım Doğrulama ve Sağlamada Biçimsel Metotlar)**

Propositional, predicate, and temporal logics. Formal systems and system modeling. Program and software verification and validation. Model checking. Basics of theorem proving. Resolution, natural deduction, and equality reasoning. Binary decision diagrams. Formal specification environments and applications. Case studies.

Önerme, yüklem ve zamansal mantıklar. Biçimsel sistemler ve sistem modelleme. Program ve yazılım doğrulama ve sağlama. Model denetimi. Teorem ispatlamanın temelleri. Çözümleme, doğal tündengelim ve eşitliğe dayalı akıl yürütme. İkili karar diyagramları. Biçimsel belirtim araçları ve uygulamaları. Örnek çalışmalar.