

**BAŐKENT ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ**

**YARI OTOMATİK DERS PROGRAMI SİSTEMİ**

**Gökhan MEMİŐ**

YÜKSEK LİSANS TEZİ

2008



# YARI OTOMATİK DERS PROGRAMI SİSTEMİ

## SEMI-AUTOMATIC COURSE SCHEDULING SYSTEM

Gökhan MEMİŞ

Başkent Üniversitesi  
Lisansüstü Eğitim Öğretim ve Sınav Yönetmeliğinin  
Bilgisayar Mühendisliği Anabilim Dalı İçin Öngördüğü  
YÜKSEK LİSANS TEZİ  
olarak hazırlanmıştır.

2008

Fen Bilimleri Enstitüsü Müdürlüğü'ne,

Bu çalışma, jürimiz tarafından **BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI**  
**'nda YÜKSEK LİSANS TEZİ** olarak kabul edilmiştir.

Başkan(Danışman) : Prof. Dr. M. Ümit KARAKAŞ

Üye : Yrd. Doç. Dr. Hasan OĞUL

Üye : Yrd. Doç. Dr. Ergün ERASLAN

ONAY

Bu tez 26/05/2008 tarihinde Enstitü Yönetim Kurulunca belirlenen yukarıdaki jüri  
üyeleri tarafından kabul edilmiştir.

...../06 /2008

Prof.Dr. Emin AKATA

FEN BİLİMLERİ ENSTİTÜSÜ MÜDÜRÜ

## TEŐEKKÖR

Yazar, bu alıőmanın gerekleőmesinde katkılarından dolayı, aőađıda adı geen kiői ve kuruluőlara itenlikle teőekkör eder.

Sayın Prof. Dr. M. Ümit KARAKAŐ'a (tez danıőmanı), alıőmanın sonuca ulaőtırılmasında ve karőtılaőtılan gülüklerin aőtılmasında gerek Ankara'da gerekse İstanbul'da her zaman yardımcı ve yol gösterici olduđu iin...

ÖZ

YARI OTOMATİK DERS PROGRAMI SİSTEMİ

Gökhan MEMİŞ

Başkent Üniversitesi Fen Bilimleri Enstitüsü

Bilgisayar Mühendisliği Anabilim Dalı

Çizelgeleme problemleri, belirli zaman aralıklarında kaynak kısıtlarını göz önünde bulundurarak makul bir şekilde atama gerektirir. Üniversiteler için ders programı hazırlama da bir çizelgeleme problemidir. Bu yerleştirme karar ağacında çok sayıda dallanma içeren bir konudur. Yarı Otomatik Ders Programı Sistemi, bir ön analiz ile eldeki atama sorununu “imkânsız”, “güç”, “tipik”, “kolay” altkümelerine ayırmakta, ilk gurutta atama yapma ağacını oluşturmayı reddetmekte, diğerlerinde ise stratejik altkümeğe uygun atamalar yapmaktadır.

Bu problemin çözümünde akıllı ajanlar kullanılmıştır. Bu sayede çok büyük olan dallanma sayısı akıllı ajanların isteklerine göre otomatik olarak azalmıştır. Akıllı ajanlar olarak öğrenci, öğretim üyesi, dekan, bölüm başkanı, bina sorumlusu kullanılarak bu kişilerin istekleri anket ile alınmıştır. Bu istekler de atamanın oluşmasına yön vermiştir.

Bu problemin verileri ORACLE veri tabanında saklanmış olup, atamanın sağlandığı programlama dili JAVA'dır. Herhangi bir fakültenin ihtiyaç duyduğu kısıtların kolayca girilebildiği, çözümlerin üretilebildiği, değiştirebildiği ve bilgilerin saklanabildiği, yarı otomatik ders programı hazırlayan bir uygulama geliştirilmiştir. Aynı zamanda kullanıcı dostu bir ara yüz oluşturulmuştur.

Bu problemde atamanın ne kadar iyi olduğunu görmek için altı akademisyenin mümkün olduğu kadar objektif değerlendirmeleri kullanılmıştır.

ANAHTAR SÖZCÜKLER: eğitim planlama, yapay us, ders programı,

Danışman: Prof.Dr. M. Ümit KARAKAŞ, Başkent Üniversitesi, Bilgisayar Mühendisliği Bölümü.

## ABSTRACT

Semiautomatic Course Schedule System

Gökhan MEMİŞ

Başkent Üniversitesi Fen Bilimleri Enstitüsü

Bilgisayar Mühendisliği Anabilim Dalı

A tabulation problem periodically requires a tenable appointment taking into account the resource constraints. Making schedules is also a tabulation problem for universities. This placement is a matter consisting multitudinous ramifications in decision tree. Semiautomatic Course Schedule System arranges the problem of scheduling in hand in subsets as “impossible”, “power”, “typical”, “easy” by means of pre-analysis, rejects to compose a tree of appointment in first group and makes appointments strategic, convenient for subsets.

In solving this problem, Intelligent Agents have been used. In this way, the number of ramification which is myriad has decreased according to wise agents' request. Using student, lecturer, dean, department chair, superintendent as wise agents, these people's requests have been learnt via public survey. And these requests have directed the formation of appointment. The data of this problem have been saved in ORACLE data base and the language of programming that ensured the appointment is JAVA. An implementation arranging automatic Schedule in which a constraint any university needs, can be entered easily, solutions can be found, altered and the data can be saved, has been developed.

At the same time an user friendly interface has been composed.

In this problem, an objective assessment of six academicians has been used as far as possible to understand how good the appointment is.

**KEY WORDS:** training scheduling, artificial intelligence, course scheduling.

# İÇİNDEKİLER LİSTESİ

Öz .....	i
Abstract.....	ii
İçindekiler Listesi.....	iii
Şekiller Listesi.....	v
Tablolar Listesi.....	vi
<b>1 GİRİŞ.....</b>	<b>1</b>
1.1 Problem Karmaşıklığı .....	1
<b>2 DERS PROGRAMI ÇİZELGELEME PROBLEMLERİ.....</b>	<b>3</b>
2.1 Yapay Zekâ .....	3
2.1.1 İnsan gibi davranan sistem (Turing Test yaklaşımı).....	4
2.2 Akıllı Ajanlar .....	8
2.2.1 Akıllı Ajan Nedir?.....	8
2.2.2 Akıllı Ajanların Sağladığı Faydalar .....	9
2.2.3 Akıllı Ajanlardan Hangi Koşullarda Yararlanılmalı? .....	10
2.3 Çizelgeleme Yöntemlerinde Kullanılmış Çözüm Yöntemleri ve Anlatımı .....	10
2.3.1 Sezgisel Algoritma.....	10
2.3.2 Aşamalı Derinleşme Algoritması(Iterative Deepening) .....	11
2.3.3 Tabu Arama .....	12
2.3.4 Benzetilmiş Tavlama Algoritması (Simulated Annealing Algorithm) .....	14
2.3.5 Çizge Boyama Algoritması (Graph Coloring Algorithm) .....	16
2.3.6 Genetik Algoritma .....	17
2.3.7 Yapay Sinir Ağları .....	20
2.4 Çizelgeleme Problemi Çözümünde Daha Önce Yapılan Çalışmalar.....	23
2.5 Yazılım Piyasasındaki Ders Programı Yapan Benzer Yazılımlar .....	27
2.5.1 aSc Timetables .....	27
2.5.2 iMagic Timetable Master .....	28
2.5.3 Mimosa Scheduling Software .....	28
2.5.4 Lantiv Timetabler 6.....	29
2.5.5 Ders Dağıtım Programı .....	30
2.5.6 Dersmatik.....	30
2.5.7 Progmatic Magic.....	31
<b>3 YARI OTOMATİK DERS PROGRAMI SİSTEMİ (yODEp v.1) .....</b>	<b>33</b>
3.1 Platform Bağımsız Tasarım Kararları.....	34
3.2 Gerçekleştirilen Sistemin Zekâ Modelleri (Akıllı Ajanlar) .....	38
3.2.1 Fakülte Dekanı: .....	39
3.2.2 Öğrenci Dekanı: .....	39
3.2.3 Fakülte Yöneticisi ya da Bina sorumlusu: .....	39
3.2.4 Bölüm Başkanı:.....	39
3.2.5 Öğretim Üyesi .....	40
3.2.6 Asistan: .....	41
3.2.7 Öğrenciler: .....	41
3.2.8 Yarı Otomatik Ders Programı Sistemi İçin Yapılan Anket ve Sonuçları .	41



3.3	Platform Bağımlı Tasarım Kararları .....	42
3.3.1	Tasarlanan Sistemin Önerilen İnsan Bilgisayar Etkileşimi .....	42
3.3.2	Sistem Mimarisi .....	42
3.3.3	Sistemde kullanılacak Olan Ekranlar .....	47
3.3.4	yoDep v1.0 Kolay Belgele Tablosu .....	62
3.3.5	Hibernate .....	69
3.3.6	Uygulama Sunucusu .....	71
3.3.7	Ders Programı Oluşturma .....	72
3.3.8	Veri Tabanı Tabloları .....	74
3.3.9	Veri Tabanı Tablolarının Birbirleri ile Bağlantıları .....	80
3.4	Tez Çalışmasının İşletilmesinden Sonraki Değerlendirmeler .....	82
<b>4</b>	<b>SONUÇ</b> .....	<b>88</b>
<b>5</b>	<b>Kaynaklar Listesi</b> .....	<b>90</b>
<b>6</b>	<b>Ekler Listesi</b> .....	<b>93</b>

## Şekiller Listesi

Şekil	Açıklaması	Sayfa Numarası
1	Akıllı Ajan çalışması	8
2	Bir ajan hareket seçmesi	9
3	Dört iterasyonlu aşamalı derinleşme algoritması	12
4	Tabu Arama Algoritması genel yapısı	13
5	Benzetilmiş Tavlama Algoritması Genel Yapısı	15
6	Çizge Boyama Algoritması örnekleri	16
7	Yapay Sinir Ağı genel yapısı	21
8	Genel yapay sinir ağı yapısı	22
9	Hatayı geriye yayma ağları	23
10	Yazılım normal masaüstü bilgisayar için tasarımı	44
11	Yazılım internetten kullanım için tasarımı	46
12	Yarı Otomatik Ders Programı Sistemi Ana Ekran Yarı Otomatik Ders Programı Sistemi Derslik - Yeni Öğretim	47
13	Elemanı - Yeni Ders Girişi Yarı Otomatik Ders Programı Sistemi Bölüm Başkanı Yarıyıl	49
14	Yönetim Ekranı Yarı Otomatik Ders Programı Sistemi Ders Programı Görüntüleme	51
15	Ekranı Yarı Otomatik Ders Programı Sistemi Listeleme Ekranı	52
16	Akademisyen Listesi	54
17	Yarı Otomatik Ders Programı Sistemi Listeleme Ekranı Ders Listesi Yarı Otomatik Ders Programı Sistemi Akıllı Ajanlar Tatmin Düzeyi	55
18	İzleme Ekranı Yarı Otomatik Ders Programı Sistemi Öğretim Elemanı Zaman	56
19	Tercih Bildirim Ekranı Yarı Otomatik Ders Programı Sistemi Öğretim Elemanı Zaman	57
20	Tercih Bildirim Ekranı (İnternet’li kullanım)	58
21	Yarı Otomatik Ders Programı Sistemi Ön Analiz Ekranı	59
22	Yarı Otomatik Ders Programı Sistemi Manüel Ders Ekleme Ekranı	60
23	Yarı Otomatik Ders Programı Sistemi Yeni Kullanıcı Ekleme Ekranı Yarı Otomatik Ders Programı Sistemi Oluşmuş Olan Programın PDF	61
24	Görüntüsü	62
25	Hibernate genel yapısı	70
26	Ön analiz uyarısı	72
27	Program oluşturulmadan önceki ön analiz	73
28	Program oluşturma	74
29	Veri Tabanı Tabloları Bağlantıları	81
30	Örnek öğretim üyesi tercih edilmeyen zaman	83
31	Örnek öğretim üyesi ders programı	84
32	Makine Mühendisliği 4. sınıf ders programı	85
33	B305 dersliğinin haftalık ders programı	86

## Tablolar Listesi

<b>Tablo</b>	<b>Açıklaması</b>	<b>Sayfa Numarası</b>
1	Karmaşa matrisi örneği	6
2	Karmaşa matrisi örneği 2	7
3	TatminManager sınıfı için kolay belgele tablosu	63
4	CreateProgram sınıfı için kolay belgele tablosu	63
5	TezManager sınıfı için kolay belgele tablosu	64
6	HibernateManager sınıfı için kolay belgele tablosu	64
7	SatisfactionFrame sınıfı için kolay belgele tablosu	65
8	ProgramViewerFrame sınıfı için kolay belgele tablosu	65
9	MainFrame sınıfı için kolay belgele tablosu	66
10	InstructorSelectionFrame sınıfı için kolay belgele tablosu	66
11	FirstAnalses sınıfı için kolay belgele tablosu	66
12	DepartmentPresidentFrame sınıfı için kolay belgele tablosu	67
13	SystemUtility sınıfı için kolay belgele tablosu	67
14	PDFConverter sınıfı için kolay belgele tablosu	68
15	ClientHandler sınıfı için kolay belgele tablosu	68

# 1 GİRİŞ

Bir okulda (üniversite, fakülte, lise, büyük bir dersane ) ders programının düzenlenmesi karar ağacında (decision tree) çok sayıda dallanma (branching) içeren bir konudur. Öğrencilerin ve öğretim elemanlarının memnun olacakları ders ve sınav programı hazırlama her üniversitede problem olmaktadır. Bu problemin öğretim üyesi eli ile yapılması durumunda çeşitli A3 boyutlu kâğıtların birleştirilmesine, renkli kâğıt parçalarına, toplu iğnelere ve saatlerce çalışacak yetenekli insan gücüne ihtiyaç duyulacaktır. Çizelgeleme problemleri belirli zaman aralıklarının kullanıcılara kısıtlar göz önünde bulundurularak makul bir şekilde atanmasını gerektirir. Üniversitelerde otomatik ders programı hazırlama bir çizelgeleme problemidir. Hedef, belirli kısıtlar kapsamında ilgili dönemde açılan bütün derslerin zaman çizelgesine yerleştirilmesidir. Bu çizelgeleme problemi polinom zamanda çözülemeyen tam (NP-complete) bir problemidir.

## 1.1 Problem Karmaşıklığı

Çizelgeleme problemleri, verimliliği esas alarak, kaynakların kullanıcılara belli kısıtlarla tahsis edilmesini hedefler.

Bir zaman çizelgeleme problemi olan ders programı hazırlamada (çizelgelemede) hedef, her sınıf için açılacak derslerin buluşma saatlerinin istenen kısıtlar sağlanarak belirlenmesidir. Örneğin, herhangi bir üniversitede bir bölümün ders programının belirlenmesi demek açılan bütün derslerin saatlerinin belirlenmesi, destek verecek asistanlar dâhil öğretim üyelerinin zaman planı ve derslik / laboratuvar kullanım planlarının buna uyumlu olarak kesinleşmesi demektir.

Doğal ve kesin kısıtlar;

- Aynı bölümün aynı dönem dersleri kesiştirilemez.
- Bir öğretim elemanın verdiği dersler aynı saate atanamaz.
- Aynı dersliğe, aynı saatte ikinci bir ders atanamaz.

Doğal fakat çok kesin olmayan kısıtlar da vardır. Örneğin;

- Bölüm koordinatörü, aynı bölümün bir önceki dönem derslerinden, öğrencilerin sıkça başarısız olduğu dersi meslek dersleri ile kesiktirmez.
- Öğretim elemanına aynı günde çok fazla ders saati koymak iyi bir çizelgeleme yöntemi değildir.

Fakülte düzeyinde ele almış olduğumuz örnek bir çizelgeleme probleminde yerleşecek 200+ ders parçasından biri, yerleştirmek için seçilecektir.

- A) Bu süreç 200+ ders parçası ile uygulandığında, oluşan ağaç yapısının (tree structure) derinliği 200+ olmaktadır. Bir bölümün derslik sayısı genelde 4 civarında olmaktadır. Derslerin atanması için arka arkaya saat dilimlerimiz olmalıdır. Bu nedenle gün içerisinde 5 saat dilimi yerleştirme için bulunmaktadır. Çünkü sabah ilk saat dilimine ve ikinci dilimine ders konulabilir. Fakat üçüncü dilime sabah başlayışı ile aynı sınıfa 2 saatlik ders konulamaz. Yani sabah için 2 saat grubu seçilmektedir. Aynı analiz öğleden sonra için yapılırsa 3 saat dilimi de öğleden sonra olmaktadır. 5 iş günü olduğuna göre gün ve saat seçerken  $5 \times 5 = 25$  gün saat çiftimiz vardır. İlk atamaya başlarken seçim karmaşıklığımız  $200 \times 25 \times 4 = 20000$  olarak modellenebilir.
- B) Derslerin dörtte biri yerleştirildikten sonra elimizde 150 dersimiz olacaktır. Dersliklerimiz 3 e inerse, günümüz 4 e iner, saatlerimiz de 3e inerse  $150 \times 3 \times 4 \times 3 = 5400$  lük dallanmamız vardır.
- C) Derslerimizin yarısı yerleştirildiğinde dallanmamız 2400 dolaylarındadır.
- D) Derslerimizin dörtte üçü yerleştiğinde de dallanmamız 200 dolaylarında olmaktadır.

Bu durumda her yarıyıl yaklaşık 200+ ders açan tipik bir fakültede “ders atama ağacı” bu şekilde olmaktadır.

Öte yandan standart büyüklükteki bir Go tahtasında 381 nokta vardır. Her bir nokta bağımsız olarak üzerine 3 değer alabilmektedir. Bunlar siyah taş, beyaz taş ve boşluktur. Bu değerler bize  $3^{381}$  değer konulabileceğini yani yaklaşık olarak  $10^{172}$  olduğunu göstermektedir[20]. Bu da bize gösteriyor ki GO oyununda

dallanma sayısı 381, çözüm derinliği 381 ile algoritma karmaşıklığı (computational complexity)  $O(381^{381})$  olmaktadır.

Satrançta geçerli pozisyon sayısı  $10^{43}$  ile  $10^{50}$  arasındadır. Buda yaklaşık olarak  $10^{123}$  karar karmaşıklığı oluşturmaktadır[21].

Tavlada zar atımı ile oluşan 21 farklı olasılık, yaklaşık 20 tane geçerli hamle ile birlikte dallanma sayısını 400 dolayına getirir ve algoritma karmaşıklığı (computational complexity)  $O(400^{80})$  civarındadır[23].

Bu açıdan bu tez çalışmasında ele alınan problem bir yapay us problemidir.

Zaman çizelgeleme problemleri için geliştirilen modeller genelde birbirinden oldukça farklıdır. Bu sebeple literatürde bulunan sonuçları karşılaştırmak zordur. Pratikte, otomatik çizelgeleme problemleri çok daha karışık kısıt ve yapıları gerektirir. Çoğu uygulamada, kısıtlar, zorunlu olanlar ve tercihe bağlı olanlar olarak iki gruba ayrılmıştır. Geliştirmiş olduğumuz sistemde ise, tüm sağlanması gerekli kısıtlara birer ağırlık atanarak bunların arasında öncelik belirlenebilmesine izin verilmiştir. Ayrıca, gerçek bir uygulamada, göz önüne alınması gereken kısıt sayısı bir hayli fazladır; bu sebeple, kısıtların girilebilmesi için kullanıcı dostu bir ara yüze de ihtiyaç duyulur.

Bu tez çalışmasında problem bir fakülte düzeyinde ele alınmıştır ve Başkent Üniversitesi Mühendislik Fakültesi verileri ile sınaama yapılmıştır.

## **2 DERS PROGRAMI ÇİZELGELEME PROBLEMLERİ**

### **2.1 Yapay Zekâ**

Yapay zekâ'nın tanımını yapmak önemli olduğu kadar da zordur. Çünkü yapay zekâ çok kapsamlı bir konu olup pek çok alt alan içermektedir ve alt alanı olarak düşünülemez pek çok alanla da ilintilidir. Yapay zekânın içerdiği ve ilintili olduğu tüm alanları kapsayacak şekilde yapılmaya çalışılacak tek bir tanım ya çok uzun olacaktır, ya çok soyut olacaktır ya da eksik olacaktır. Dolayısıyla yapay zekânın tanımını yapmaya çalışan kişiler genellikle kendi uzmanlık alanlarını ön plana çıkaran tanımlar yapmışlardır.

Zeki sistemler bu gün zeki etmenler (intelligent agents) olarak yeniden tanımlanmaktadır[4]. Buna göre zeki etmenlerin şu temel bileşenlere sahip olması gerekmektedir:

- Algılama
- Düşünme
- Eylem

Ayrıca, bu her bir bileşenin de "zeki" diye vasıflandırılabilen niteliklere sahip olması gerekiyor. Zeki bir etmen, algılamasını gerçek zamanda, seçimli, öncelikli ve bağımsız olarak yapabilmelidir. Aynı şekilde, düşünme bileşeni de benzer özelliklere sahip olmalı, olayların gelişimine göre algılama bileşeninden gelen verilerden ve kendi içinde oluşturduğu modellerden gerçek zamanda durum tespiti ve değerlendirmesi yapabilmeli, hafıza, anlama, problem çözme, öğrenme (ve hatta buluş yapabilme), planlama ve kontrol gibi özelliklere sahip olmalıdır. Eylem bileşeni de aynı şekilde zeki özelliklere sahip olmalı ve düşünme bileşeninden gelen kararları gerçek zamanda uygulayabilmelidir. Ayrıca, bu üç bileşen (algılama, düşünme, eylem) özellikle hızlı hareketi gerektiren durumlarda birbirleriyle tam bir uyum içinde olmalıdır.[4]

Yapay zekâ tanımları Russell ve Norvig' e göre 4 ana başlıkta toplanmaktadır. Bunlar "İnsan gibi düşünen sistem", "İnsan gibi davranan sistem", "Mantıklı, rasyonel olarak düşünen sistem" ve "Mantıklı, rasyonel olarak davranan sistem" dir [4, sf.5].

### **2.1.1 İnsan gibi davranan sistem (Turing Test yaklaşımı)**

Buradaki sistemin ana yaklaşımı Turing Testi yaklaşımıdır. Buradaki ana amaç, sistemin insanın yaptığı sistem ile çok yakın hatta daha iyi bir sistem yapmasını sağlamaktır.

### 2.1.1.1 Turing Test

İngiliz Matematikçi Turing'in kendi adıyla anılan "Turing Testi" 1950 tarihli "Computer Machinery and Intelligence" başlıklı makalesinde yayınlanmıştır.

Testin amacı bir makinenin, bir yazılımının görünürdeki (pseudo) zekâsını sınamaktır. İnsan kadar zeki diye nitelendirebilecek bir makinenin yazılımının bu testi geçebilmesi gerekmektedir.

Bu teste göre, bir sorgulayıcı (insan), bir hakem, zekâsı sınanacak bir yapay sistem, makine ya da yazılım (Sistem - A) ve bir alternatif insan (Sistem - B) ayrı ayrı kabinlere yerleştirilir. Sorgulayıcı, ne insanı ne de makineyi görür ve duyar. Sadece bir uç birim ekranı aracılığıyla onlarla iletişim kurabilir. Sorgulayıcının görevi, insana ve makineye yazı ekranından yönelttiği sorular aracılığı ile insanı ve makineyi ayırt etmektir. Eğer sorgulayıcı insan ile makineyi ayırt edemiyor ise makine Turing'e göre zeki olarak nitelendirilir.

Ümit Karakaş'ın yaptığı yapay zekânın Turing testi konusunda 1950 deki özgün tanımına göre;

- Yapay Us ürününü sınavan kişi (hakem) karşısındaki biri insan diğeri yapay sistem (AI sistemi) olan ortamdaki görsel yönden ve diğeri duyu algılamaları (klavye sesi, disk ya da manyetik şerit dönüş sesi) yönünden izole olmalıdır.
- Hakemin biri insan diğeri yapay sisteme verdiği sorular / problemler iki sisteme yaklaşık olarak eşit dağıtılacaktır. Yanıtlar, hakeme standart bir gecikme ile verilecek, hakemin yanıt süresinden teşhis etmesi önlenecektir.
- Hakemin görmediği ortama koyduğunuz insan, sınavdığınız yapay sistemin (AI sistemi) uygulama alanında konu uzmanı olmalıdır. (Tıp doktoru olmayan bir kişiyi, bir tıbbi uzman sisteme paralel olarak hakemin görmediği ortama koyarsanız değerlendirme sağlıklı olmaz). Ayrıca hakem de sınavanan uzman sistem alanında bilgili ve deneyimli olmalıdır.



- Hakem'i %50 dolayında yanılan ( Aşağıda karmaşa matrisi / confusion matrix konusuna bakınız ) yapay sistemler başarılı sayılır.
- Yapay Us'un yaklaşık elli yıllık geçmişinde konu sınırlaması olmaksızın turing Test'ten başarıyla geçen sistem olmamıştır ( olabilmesi için birkaç milyon üretim kurallı sistem yapılmış ve çıkarım sistemleri mükemmelen çalışıyor olmalıydı ).

Turing Test'ten daha ayrıntılı bir değerlendirme sistemi, Uzman Sistemler ve Yapay Us Sistemlerini değerlendirmede kullanılan Karmaşa Matrisi (Confusion Matrix) yöntemidir.

### 2.1.1.2 Karmaşa Matrisi (Confusion Matrix)

Bu matris "Turing Testi" nin daha ayrıntılı devamı niteliğindedir. Cevapların kaçının doğru bilindiğini göstermek için tutulmaktadır. Sistemde kaç soru varsa her bir soru için bir matris satırı açılacaktır. Bununla beraber bilgisayar ve insan için ise bir kolon açılacaktır. Soruların cevaplarını kime verdiği göre matris doldurulacaktır. Bu matrise göre %50'den fazla yanlış cevap verilirse yapay zekâ o kadar başarılıdır demektir.

**Tablo 1. Karmaşa matrisi örneği**

		Tahmin Edilen	
		Negatif	Pozitif
Gerçek	Negatif	a	b
	Pozitif	c	D

Matriste başarı ana köşegen değerlerinin toplamı olmaktadır.

Yukarıdaki tabloda bir hastalığın olup olmamasının gerçek ve tahmin edilen değerlerinin tablosu bulunmaktadır.

- Bu verilere göre doğruluk (Accuracy) oranı :

$$AC = \frac{a+d}{a+b+c+d}$$

- Hastalığın doğru pozitif olmasının oranı (True positive rate (TP)) :

$$TP = \frac{d}{c+d}$$

- Hastalığın yanlış pozitif olmasının oranı (False positive rate (FP)):

$$FP = \frac{b}{a+b}$$

- Hastalığın doğru negatif olmasının oranı (True negative rate (TN)) :

$$TN = \frac{a}{a+b}$$

- Hastalığın yanlış negatif olmasının oranı (False negative rate (FN)):

$$FN = \frac{c}{c+d}$$

- Pozitif tahmin oranı (Prediction):

$$P = \frac{d}{b+d}$$

olmaktadır.

Karmaşa matrisine gerçek hayattan örnek vermek gerekirse, laboratuarda yapılan tahlilde, bilgisayarın ve gerçeğin sonuçları aşağıdaki gibi olmaktadır.

**Tablo 2. Karmaşa matrisi örneği 2**

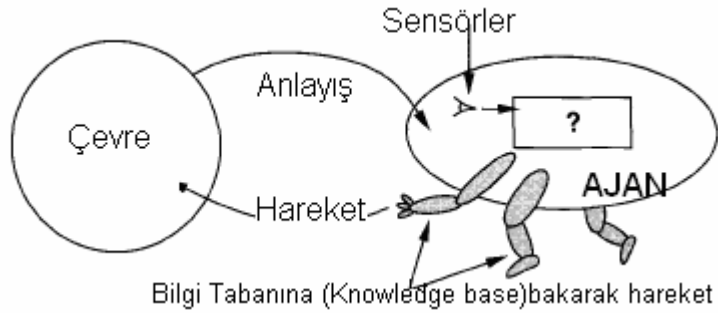
	Test sonucu = mikrop yok	Test sonucu = mikrop var
Mikrop yok ya da az	<b>68</b>	<b>2</b>
Mikrop var ve yoğun	<b>3</b>	<b>27</b>

## 2.2 Akıllı Ajanlar

### 2.2.1 Akıllı Ajan Nedir?

Özellikle 90'lı yılların ikinci yarısından itibaren, esasen bilgisayar bilimleri içinde bir alt disiplin olan yapay zekâ çalışmaları alanında yer alan "akıllı ajan" yazılım çalışmaları yeni bir ivme kazanmıştır. Akademik çerçevede artarak ilerleyen çalışmaların, yukarıda da belirtildiği gibi Internet'in yaygınlaşması ile birlikte yeni bir evreye girdiği söylenebilir. 1994 yılında yayınlanan bir makalede de [18] akıllı ajanlar alanında da belli bir bilimsel doygunluğa ulaşıp, uygulama dünyasına geçilmesi için önemli temelleri teşkil etmişlerdir.

Ajan, çevresinde bulunan sensorları ile algılayarak hareket eder. İnsan ajanlar gözü, kulağı, elleri, burnu ile doğru kararını vererek hareket eden varlıklar olduğunu düşünürsek bizim yaratacağımız ajanlar da sensorları ve algılayıcıları ile sonuca varacak olanlardır[4, sf.31].



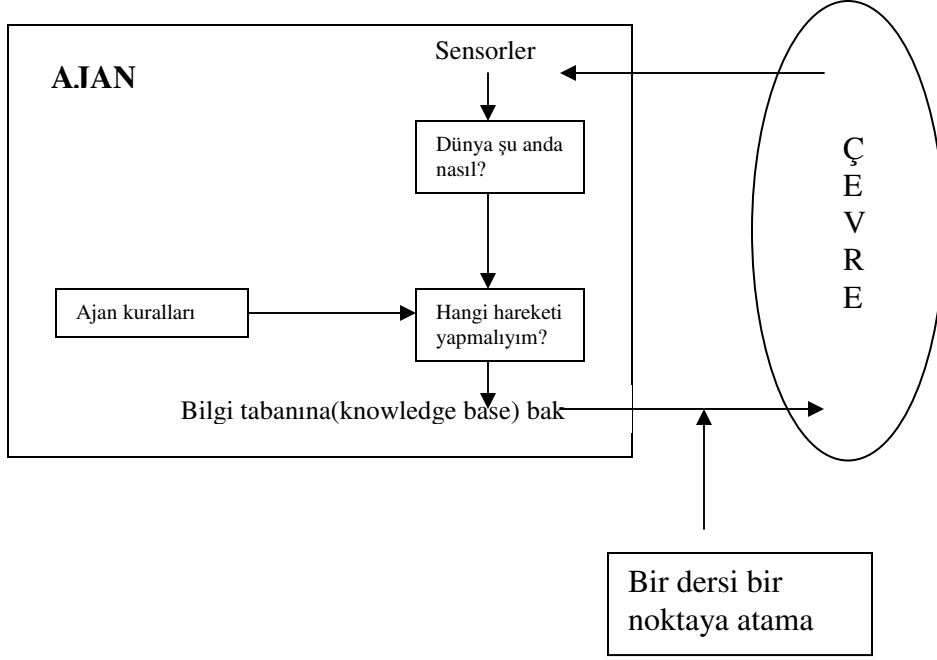
Şekil 1 . Akıllı Ajan çalışması

Bu ajanlar en iyi algılama yapacağı hareketin doğruluğunun maksimum olmasını sağlamaktadır. Bu da ajanın bilgisinin en fazla olmasını gerektirmektedir[4, sf.33].

Bir ajanın çalışma algoritması aşağıdaki şekilde olabilir:

- Hafızayı güncelle
- En iyi seçimi seç
- En iyi seçimi yap

Bir ajan çevresine göre hareketin doğruluğunu şu şekilde seçmektedir:



**Şekil 2 . Bir ajan hareket seçmesi**

## 2.2.2 Akıllı Ajanların Sağladığı Faydalar

Akıllı ajanların sağladığı faydalar diğer karar destek ve yönetim bilgi sistemlerinden biraz daha farklıdır.

- Akıllı ajanlar uzman şahısların bilgisini yeri gelince kullanmak üzere depolanmasına imkân verir. Belli bir uzman şahsın konusu ile ilgili sahip olduğu bilgileri akıllı ajan olarak kullanmak mümkündür. Örneğin bir iş yerinde alanıyla ilgili tercüman olan bir şahsın o işten emekli olması işletme açısından büyük sorun yaratması bekleniyorsa, bu şahsın sahip olduğu bilgileri akıllı ajan haline getirmek etkili bir çözüm olabilir.
- Bir tek akıllı ajandan birden fazla kullanıcının yararlanması mümkündür.
- Akıllı ajanlar karar vericilerin performans ve üretkenliğini artırır.
- Akıllı ajanlar belli bir konu ile ilgili karar verme sürecinde insanların aksine istikrarlı ve tutarlıdır. Yani, akıllı ajan belli bir karar verme durumunda her zaman aynı bilgiyi verir.

- Akıllı ajanlar kritik şahıslara olan bağımlılığı azaltır. İnsanlar emekli olabilir, hasta düşebilir, izne çıkabilir veya işten ayrılabilir. Bilgisayarlar ise çay molası bile istemez. Onlardan her zaman yararlanmak mümkündür.
- Akıllı ajanlar karar vericileri eğitmek üzere de istifade etmek mümkündür.

### **2.2.3 Akıllı Ajanlardan Hangi Koşullarda Yararlanılmalı?**

Akıllı ajanlar ancak gerekli olduğu zamanlarda kullanılmalıdır. Bazı işletmelerin bulunduğu koşullar, akıllı ajanın kullanılmasını gerek maliyet gerekse sağlanan faydanın önemsiz olması yüzünden haklı kılmayabilir. Akıllı ajanlardan ancak bu koşullar altında yararlanılması tavsiye edilmektedir:

- İşin tekdüzeliğine bağlı olarak Akıllı ajanın sık sık kullanılmasına gereksinim duyuluyorsa ve kullanıcı sayısı akıllı ajanların kullanılmasını maliyet boyutunda ekonomik kılacak kadar fazla sayıda ise,
- Karar verme durumu karmaşık(basit durumlar için basit bir bilgisayar programından da yararlanılabilir.)
- Karar verme mantığı bir kural hiyerarşisine dönüştürülebiliyorsa,
- Uygulama öneri, sınıflama, teşhis, yorum, açıklama, çözüm yolu seçme, durumu değerlendirmede tahmin etme üzerinde yoğunlaşıyorsa,

### **2.3 Çizelgeleme Yöntemlerinde Kullanılmış Çözüm Yöntemleri ve Anlatımı**

Çizelgeleme problemlerinde literatürde birçok çalışmaya rastlanmaktadır. Bu çalışmalar genelde farklı şekilde çözüm aramışlardır. Genel olarak kesin çözüm veren çalışmalar ve sezgisel çalışmalar ile çözüm aranmaya çalışılmıştır. Kesin çözüm veren çalışmalar büyük problemlerde çok uzun zamanlarda sonuca ulaştığı için genelde sezgisel algoritmalar son zamanlarda daha fazla kullanılmıştır.

#### **2.3.1 Sezgisel Algoritma**

Genel olarak kısmi bilgiye dayalı arama yöntemiyle problem çözme ya da yarı zeki yaklaşım denilebilir. Yüz binlerce değişkeninin bulunduğu denklemlerde en

iyiye en yakın çözümü bulmayı sağlayan algoritmalarıdır. En iyi çözüme ulaşması yerine bazen en iyinin çok yakın olan çözümüne ulaşmaktadırlar. Bunun sebeplerin başında sezgisellerin hesaplanmasının en iyi sonuçları veren algoritmalara göre daha kısa sürmesidir. Örnek olarak bir algoritma bir problemin en iyi çözümünü 10 saatte veriyor, fakat sezgisel metodunuz aynı problemle ilgili bir çözümü 2 dakikada veriyorsa ve de bulduğu çözüm en iyiye çok yakınsa sezgisel tercih edilir. Fakat hesaplama sürecinin kısa olması sezgisellerin tercih edilmesinde ana sebep değildir. Bir diğer önemli nokta da en iyi çözümlerin gerçek dünyadaki sistemlere tam olarak adapte edilememesidir. Bu süreçte en iyi çözüme ulaşmak için ufak da olsa bir kaç değişiklik yapılır ve dolayısıyla gerçek dünyada değiştirilen en iyi çözüm alt en iyi sonuçlar doğurur. Fakat sezgisel çözümler daha kolay bir şekilde gerçek dünya sistemlerine adapte edilebilir ve sezgisel çözümde oluşan kayıp çoğu zaman en uygun çözümlerin uygulanma esasında ortaya çıkan kayıplardan daha az olur.

### **2.3.2 Aşamalı Derinleşme Algoritması(Iterative Deepening)**

Bu algoritmanın ana mantığı ilk önce bütün bilgilere bakarak en iyiyi bulmaya çalışır. Daha sonra yavaş yavaş derinleşerek arama alanını küçültür ve en iyi sonuca daha çabuk, kolay şekilde ulaşmaya çalışır.

Algoritmasına Russell ve Norvig' e göre bakarsak[4, sf.79];

**Fonksiyon AŞAMALI DERİNLEŞME ARAMASI(problem) returns** çözüm dizisi

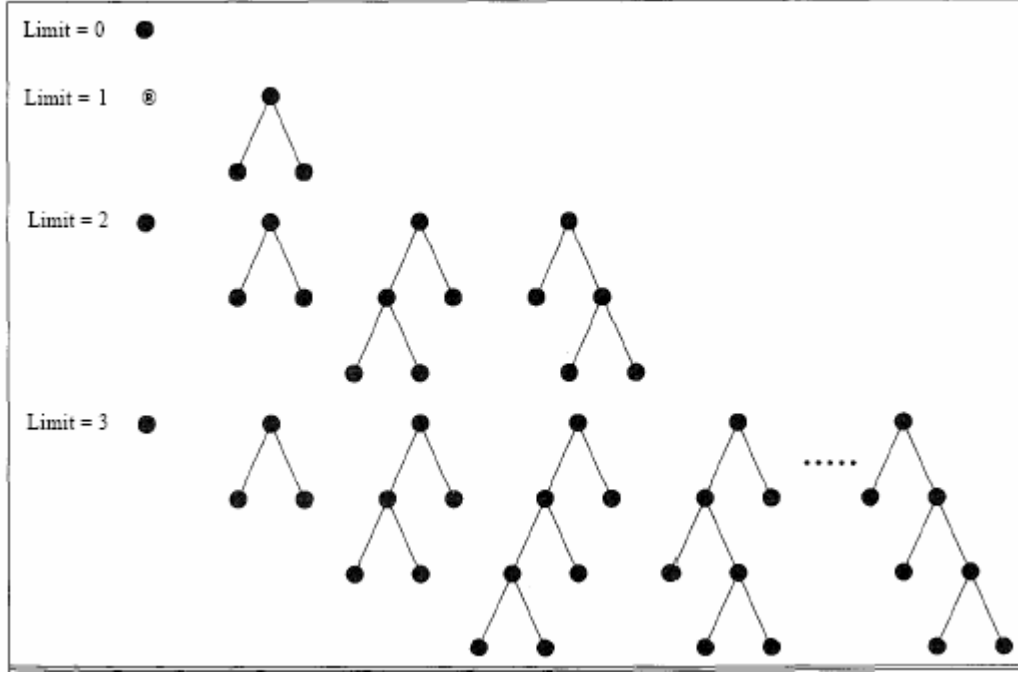
**Input** : problem

**For** derinlik < sonsuz, derinlik : 0 **do**

**İf**(derinlik limitini kontrol et(problem, derinlik)) başarılıdır. **Return** sonuç

**End**

**Return** başarısız



Şekil 3: Dört iterasyonlu aşamalı derinleşme algoritması

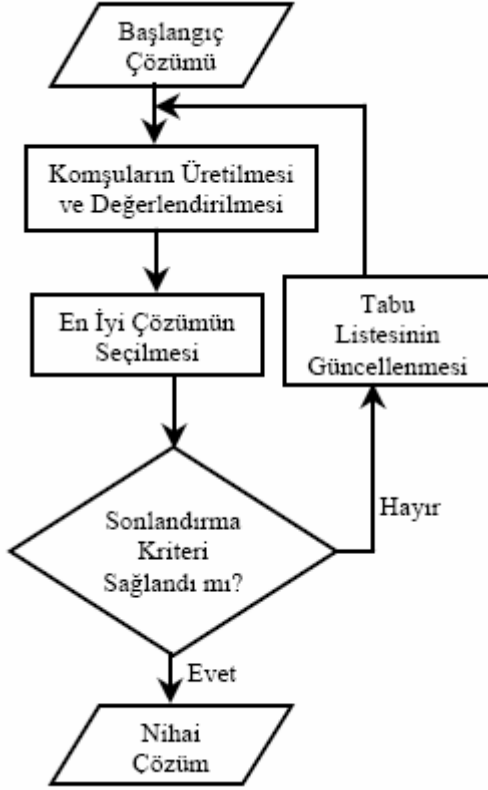
### 2.3.3 Tabu Arama

İlk olarak Hansen tarafından İtalya'daki sayısal metotlar kongresinde önerilmiş olup Glover[8] tarafından çözüm kümesi kesikli olan problemlerin çözümü için önerilmiş yüksek seviyeli bir sezgisel programlama tekniğidir.

Tabu arama yerel optimumlardan kurtulmayı sağlamak amacı ile yerel arama metotlarına yol gösteren sezgisel bir tekniktir.

Tabu aramada kısa dönemli hafıza ve uzun dönemli hafıza olmak üzere iki şekilde hafıza yapısı vardır.

Tabu arama algoritmasının genel akış şekli aşağıdaki gibidir:



**Şekil 4 . Tabu Arama Algoritması genel yapısı**

Tabu aramanın temel elemanları;

**Başlangıç çözümü:** Tabu aramada başlangıç çözümü rastsal olarak seçilebileceği gibi başlangıç çözümünün belirlenmesinde herhangi bir algorithmada kullanılabilir.

**Hareket mekanizması:** Mevcut çözüm üzerinde yapılan değişikliklerle elde edilebilecek yeni çözümleri belirleyen bir mekanizmadır. Hareket mekanizmasının problem yapısına bağlı olmasının yanı sıra uygun bir şekilde belirlenmesi tabu aramanın başarısı açısından oldukça önemlidir.

**Tabu yıkma kriterleri:** Tabu olarak bir hareketin yasak olmasına rağmen yapılabileceği durumları belirtmek için kullanılır. Uygun tabu yıkma kriterlerinin kullanılması tabu aramanın başarımı üzerinde oldukça etkilidir. En basit ve en sık kullanılan tabu yıkma kriteri, aramada mevcut iterasyona kadar bulunan en iyi çözümden daha iyi bir çözüm veren hareketin tabu olup olmamasına bakılmaksızın yapılmasıdır.



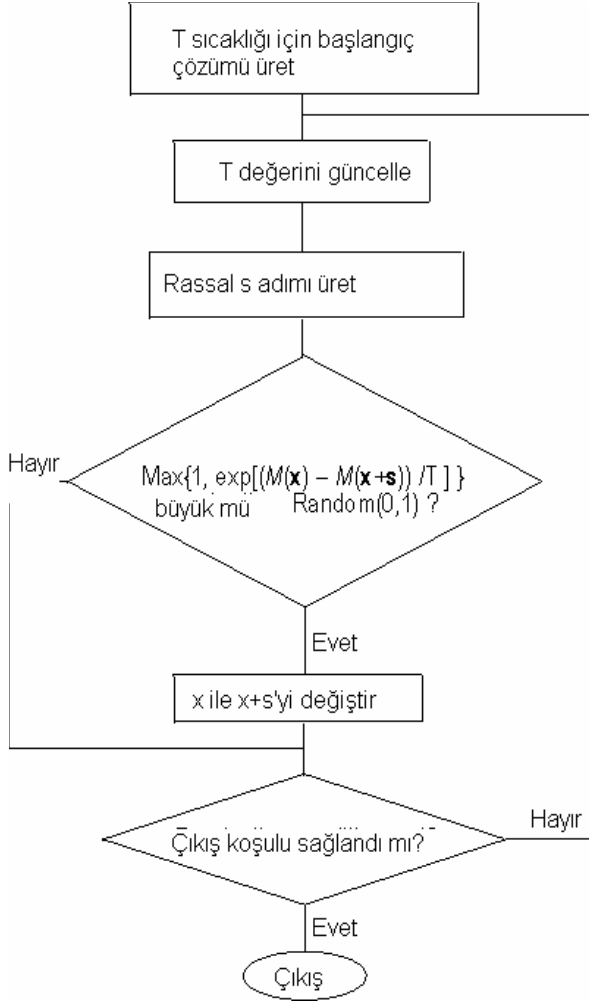
**Aday liste stratejileri:** Tabu arama mümkün ve tabu olmaya en iyi hareketin, makul zaman ve çaba ile seçilmesi kuralını kullanır. En iyi hareket amaç fonksiyonu açısından düşünüldüğünde en iyi amaç fonksiyonu değerini veren harekettir. En iyi kavramı ilgilenilen probleme göre değişebilir. Örneğin hareketler mevcut çözüm üzerinde yapılan etkiye göre değerlendirilebilirler. Mevcut  $x$  çözümün komşu kümesi geniş veya bu kümedeki elemanların değerlendirilmesinin hesaplama maliyeti yüksek ise, aday liste stratejilerinin kullanımı kaçınılmaz olmaktadır. Aday listeleri ilgilenilen problem yapısına göre veya problem yapısından bağımsız genel stratejilere göre oluşturur. Bir aday liste stratejisinin etkinliği zorluğunun azaltılmasının açısından düşünülmemelidir. Bunun yerine, belli bir bilgisayar zamanı içinde bulunan çözümün kalitesi, kullanılan aday liste stratejisi için etkinlik ölçütü olmalıdır.

**Hafıza:** Tabu aramada uzun ya da kısa dönem hafızalar mevcuttur. Bu hafızalar daha önceki en iyi çözümlerin bilgisini tutmaktadır.

#### **2.3.4 Benzetilmiş Tavlama Algoritması (Simulated Annealing Algorithm)**

Bu algoritma ilk olarak 1983 yılında Kirkpatrick and Gelatt and Vecchi tarafından [19] kullanılmıştır.

En iyileme problemlerini çözmek için kullanılan sezgisel yöntem metallerin tavlama işlemine benzediğinden bu ismi almıştır. Katı cisimlerin özellikle metallerin soğurken mükemmel atomik dizilişlerini örnek alan, yapay zekâ en iyileştirme algoritmalarının bir tanesidir. İçerisinde rasgelelik barındırır. Soğutma işlemi bu algorithmda daha iyi sonuçların bulunmasını sağlayacak yeni komşu çözümlerin üretilmesini sağlayan bir algoritmadır.

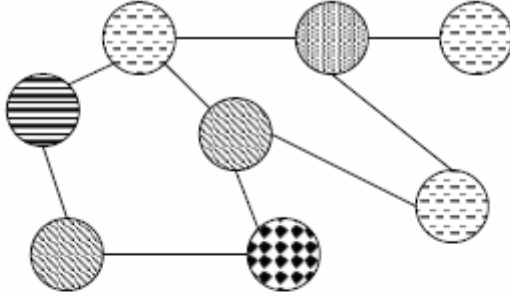


**Şekil 5 . Benzetilmiş Tavlama Algoritması Genel Yapısı**

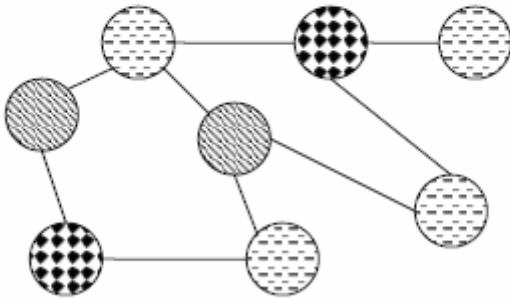
Benzetilmiş tavlama algoritması, pek çok değişkene sahip fonksiyonların en büyük veya en küçük değerlerinin bulunması ve özellikle pek çok yerel en küçük değere sahip doğrusal olmayan fonksiyonların en küçük değerlerinin bulunması için tasarlanmıştır. Diğer olasılıksal yaklaşımlar (genetik algoritmalar, tabu arama vb.) gibi en iyi çözümün en kısa zamanda üretimini sağlar. Bu sebeple, özellikle matematiksel modellerle gösterilemeyen kombinyonel problemlerin eniyileme uygulamalarında tercih edilir. Benzetilmiş tavlama algoritması; elektronik devre tasarımı, görüntü işleme, yol bulma problemleri, seyahat problemleri, malzeme fiziği benzetimi, kesme ve paketleme problemleri, akış çizelgeleme ve iş çizelgeleme problemlerinin çözümlerinde başarılı sonuçlar vermiştir[19].

### 2.3.5 Çizge Boyama Algoritması (Graph Coloring Algorithm)

Grafik renklendirme, grafik üzerinde birbirine komşu olan düğümlere farklı renk atama işlemidir. Amaç, en az sayıda renk kullanılarak tüm düğümlere komşularından farklı birer renk vermektir. Renklendirmede kullanılan renk sayısı kromatik sayı olarak adlandırılır. Uygulamada, grafik renklendirmenin kullanılacağı alanların başında, ilk akla gelen, harita üzerindeki bölgelerin renklendirilmesi olmasına karşın, grafik renklendirme bilgisayar biliminde ve günlük yaşamdaki birçok problemin çözümünde kullanılan bir yaklaşımdır. Örneğin, sınırlı sayıda işlemcisi olan bir sistemde işlemlerin(süreç) işlemcileri kullanma zamanları ve sıralamasının belirlenmesinde, üniversitenin bir fakültesinde sınav saat ve günlerinin çakışmayacak bir şekilde yerleştirilmesinde çözüm sunar, bir de tabi ki bölge renklendirmede en az sayıda renk kullanılmasını sağlar.



*k = 5 olacak şekilde k- renk boyama*



*k = 3 olacak şekilde en iyi boyama*

**Şekil 6 . Çizge Boyama Algoritması örnekleri**

### 2.3.6 Genetik Algoritma

**Genetik algoritmalar**, doğada gözlemlenen [evrimsel](#) sürece benzer bir şekilde çalışan arama ve eniyileme yöntemidir. Karmaşık çok boyutlu arama uzayında en iyinin hayatta kalması ilkesine göre bütünsel en iyi çözümü arar.

Genetik algoritmalar problemlere tek bir çözüm üretmek yerine farklı çözümlerden oluşan bir çözüm kümesi üretir. Böylelikle, arama uzayında aynı anda birçok nokta değerlendirilmekte ve sonuçta bütünsel çözüme ulaşma olasılığı yükselmektedir. Çözüm kümesindeki çözümler birbirinden tamamen bağımsızdır. Her biri çok boyutlu uzay üzerinde bir [vektördür](#).

Genetik algoritmalar problemlerin çözümü için evrimsel süreci bilgisayar ortamında taklit ederler. Diğer eniyileme yöntemlerinde olduğu gibi çözüm için tek bir yapının geliştirilmesi yerine, böyle yapılardan meydana gelen bir küme oluştururlar. Problem için olası pek çok çözümü temsil eden bu küme genetik algoritma terminolojisinde nüfus adını alır. Nüfuslar vektör, kromozom veya birey adı verilen sayı dizilerinden oluşur. Birey içindeki her bir elemana gen adı verilir. Nüfustaki bireyler evrimsel süreç içinde genetik algoritma işlemcileri tarafından belirlenirler.

Problemin bireyler içindeki gösterimi problemden probleme değişiklik gösterir. Genetik algoritmaların problemin çözümündeki başarısına karar vermedeki en önemli faktör, problemin çözümünü temsil eden bireylerin gösterimidir. Nüfus içindeki her bireyin problem için çözüm olup olmayacağına karar veren bir uygunluk fonksiyonu vardır. Uygunluk fonksiyonundan dönen değere göre yüksek değere sahip olan bireylere, nüfustaki diğer bireyler ile çoğalmaları için fırsat verilir. Bu bireyler çaprazlama işlemi sonunda çocuk adı verilen yeni bireyler üretirler. Çocuk kendisini meydana getiren ebeveynlerin (anne, baba) çeşitli aramalarda anne ve babasının özelliklerini taşır. Yeni bireyler üretilirken düşük uygunluk değerine sahip bireyler daha az seçileceğinden bu bireyler bir süre sonra nüfus dışında bırakılırlar. Yeni nüfus, bir önceki nüfusta yer alan uygunluğu yüksek bireylerin bir araya gelip çoğalmalarıyla oluşur. Aynı zamanda bu nüfus önceki nüfusun uygunluğu yüksek bireylerinin sahip olduğu özelliklerin büyük bir kısmını içerir. Böylelikle, pek çok nesil aracılığıyla iyi özellikler nüfus

içersinde yayılırlar ve genetik işlemler aracılığıyla da diğer iyi özelliklerle birleşirler. Uygunluk değeri yüksek olan ne kadar çok birey bir araya gelip, yeni bireyler oluşturursa arama uzayı içerisinde o kadar iyi bir çalışma alanı elde edilir. Probleme ait en iyi çözümün bulunabilmesi için;

- Bireylerin gösterimi doğru bir şekilde yapılmalı,
- Uygunluk fonksiyonu etkin bir şekilde oluşturulmalı,
- Doğru genetik işlemci(mutasyon, crossover) oranı seçilmeli.

Bu durumda çözüm kümesi problem için bir noktada birleşecektir. Genetik algoritmalar, diğer eniyileme yöntemleri kullanılırken büyük zorluklarla karşılaşılan, oldukça büyük arama uzayına sahip problemlerin çözümünde başarı göstermektedir. Bir problemin bütünsel en iyi çözümünü bulmak için garanti vermezler. Ancak problemlere makul bir süre içinde, kabul edilebilir, iyi çözümler bulurlar. Genetik algoritmaların asıl amacı, hiçbir çözüm tekniği bulunmayan problemlere çözüm aramaktır. Kendilerine has çözüm teknikleri olan özel problemlerin çözümü için mutlak sonucun hızı ve kesinliği açısından genetik algoritmalar kullanılmazlar. Genetik algoritmalar ancak;

- Arama uzayının büyük ve karmaşık olduğu,
  - Mevcut bilgiyle sınırlı arama uzayında çözümün zor olduğu,
  - Problemin belirli bir matematiksel modelle ifade edilemediği,
  - Geleneksel eniyileme yöntemlerinden istenen sonucun alınmadığı alanlarda,
- etkili ve kullanışlıdır.

Genetik algoritmalar parametre ve sistem tanılama, kontrol sistemleri, robot uygulamaları, görüntü ve ses tanıma, mühendislik tasarımları, planlama, yapay zekâ uygulamaları, uzman sistemler, fonksiyon ve kombinasyonel eniyileme problemleri ağ tasarım problemleri, yol bulma problemleri, sosyal ve ekonomik planlama problemleri için diğer eniyileme yöntemlerinin yanında başarılı sonuçlar vermektedir. [11,12]

### **2.3.6.1 Genetik Algoritmanın Aşamaları**

1. Başlangıç: n adet kromozom içeren popülasyonun oluşturulması (problemin uygun bir çözümü)

2. Uyumluluk: her x kromozomu için uyumluluğun f(x) değerlendirilmesi,

3. Yeni popülasyon: Yeni popülasyon oluşuncaya kadar aşağıdaki adımların tekrar edilmesi,

3a. Seçim: İki ebeveyn kromozomun uyumluluğuna göre seçimi (daha iyi uyum seçilme şansını artırır.),

3b. Çaprazlama: Yeni bir fert oluşturmak için ebeveynlerin bir çaprazlama olasılığına göre çaprazlanması. Eğer çaprazlama yapılmazsa yeni fert anne veya babanın kopyası olacaktır. Çaprazlama aşağıdaki örnekteki şekilde yapılabilmektedir:

*Kromozom 1: 11001 | 00100110110*

*Kromozom 2: 11011 | 11000011110*

*Oluşan Birey 1: 11001 | 11000011110*

*Oluşan Birey 2: 11011 | 00100110110*

3c. Mutasyon: Yeni ferden mutasyon olasılığına göre kromozom içindeki konumu (lokasyon) değiştirilir. Mutasyon aşağıdaki örnekteki şekilde yapılabilmektedir:

*Orijinal Birey 1: 1101111000011110*

*Orijinal Birey 2: 1101100100110110*

*Değişmiş Birey 1: 1100111000011110*

*Değişmiş Birey 2: 1101101100110110*

3d. Ekleme: Yeni bireyin yeni popülasyona eklenmesi.

4. Değişirme: Algoritmanın yeniden çalıştırılmasında oluşan yeni popülasyonun kullanılması,

5. Test: Eğer sonuç tatmin ediyorsa algoritmanın sona erdirilmesi ve son popülasyonun çözüm olarak sunulması.

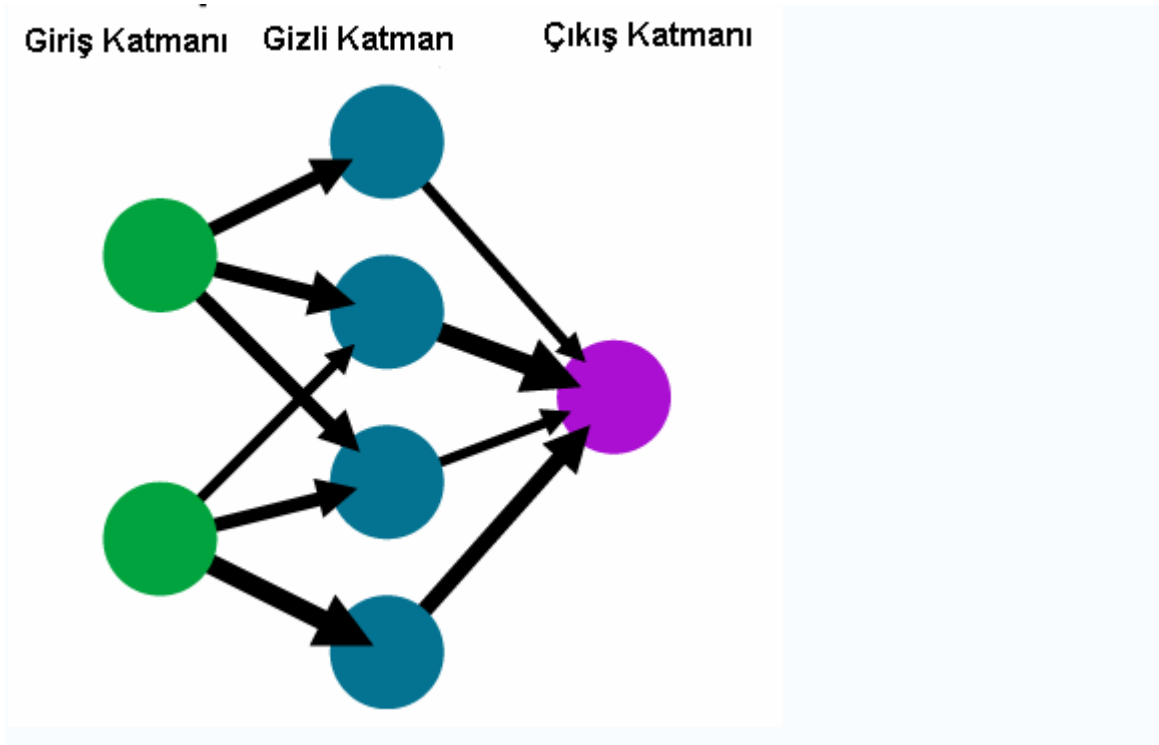
6. Döngü: 2. adıma geri dönülmesi.

Görüldüğü gibi genetik algoritmanın yapısı oldukça geneldir ve herhangi bir probleme uygulanabilir. Kromozomların tanımlanması genellikle ikili düzendeki sayılarla yapılır. Çaprazlama işlemi için kullanılan bireyler iyi bireylerden seçilir.

GA kullanılarak bir problem çözülecekse algoritmanın ne zaman sonlanacağına kullanıcı karar vermektedir. Genetik Algoritmanın belli bir sonlanma kriteri yoktur. Sonucun yeterince iyi olması veya yakınsamanın sağlanması algoritmanın durması için kriter olarak kullanılabilir [10].

### **2.3.7 Yapay Sinir Ağları**

İnsanlığın doğayı araştırma ve taklit etme çabalarının ürünlerinden biri de yapay sinir ağları (YSA) teknolojisidir. YSA, basit biyolojik sinir sisteminin çalışma şekli simüle edilerek tasarlanan programlama yaklaşımıdır. Simüle edilen sinir hücreleri ([nöronlar](#)) içerir ve bu nöronlar çeşitli şekillerde birbirlerine bağlanarak ağı oluştururlar. Bu ağlar öğrenme, hafızaya alma ve veriler arasındaki ilişkiyi ortaya çıkarma kapasitesine sahiptir. Diğer bir ifadeyle, YSA'lar, normalde bir insanın düşünme ve gözlemlemeye yönelik doğal yeteneklerini gerektiren problemlere çözüm üretmektedir. Bir insanın, düşünme ve gözleme yeteneklerini gerektiren problemlere yönelik çözümler üretebilmesinin temel sebebi ise insan beyninin ve dolayısıyla insanın sahip olduğu yaşayarak veya deneyerek öğrenme yeteneğidir.



Şekil 7 . Yapay Sinir Ağı genel yapısı

Biyolojik sistemlerde öğrenme, nöronlar arasındaki [sinaptik](#) (*synaptic*) bağlantıların ayarlanması ile olur. Yani, insanlar doğumlarından itibaren bir yaşayarak öğrenme süreci içerisine girerler. Bu süreç içinde beyin sürekli bir gelişme göstermektedir. Yaşayıp tecrübe ettikçe sinaptik bağlantılar ayarlanır ve hatta yeni bağlantılar oluşur. Bu sayede öğrenme gerçekleşir. Bu durum YSA için de geçerlidir. Öğrenme, eğitime yoluyla örnekler kullanarak olur; başka bir deyişle, gerçekleşme girdi/çıkıtlı verilerinin işlenmesiyle, yani eğitime algoritmasının bu verileri kullanarak bağlantı ağırlıklarını (*weights of the synapses*) bir yakınsama sağlanana kadar, tekrar tekrar ayarlamasıyla olur.

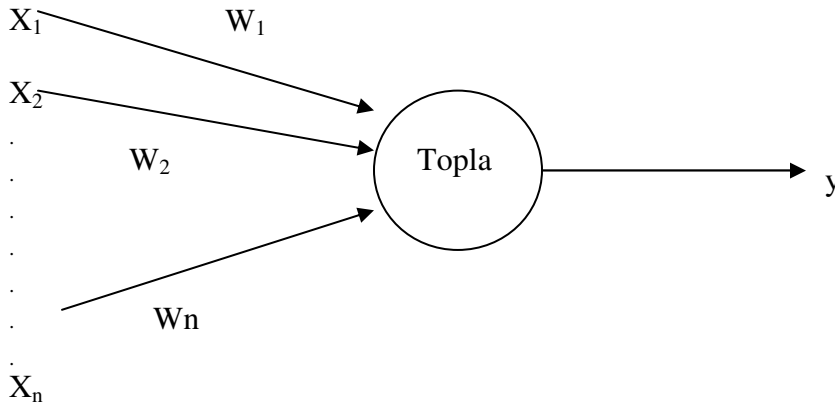
YSA'lar, ağırlıklandırılmış şekilde birbirlerine bağlanmış birçok işlem biriminden (nöronlar) oluşan [matematiksel](#) sistemlerdir. Bir işlem birimi, aslında sık sık transfer fonksiyonu olarak anılan bir denklemdir. Bu işlem birimi, diğer nöronlardan sinyalleri alır, bunları birleştirir, dönüştürür ve sayısal bir sonuç ortaya çıkartır. Genelde, işlem birimleri kabaca gerçek nöronlara karşılık gelir ve bir ağ içinde birbirlerine bağlanır. Bu yapı da sinir ağlarını oluşturmaktadır.



Sinirsel (*neural*) hesaplamanın merkezinde dağıtılmış, adaptif ve doğrusal olmayan işlem kavramları vardır. YSA'lar, geleneksel işlemcilerden farklı şekilde işlem yapmaktadır. Geleneksel işlemcilerde, tek bir merkezi işlem birimi her hareketi sırasıyla gerçekleştirir. YSA'lar ise her biri büyük bir problemin bir parçası ile ilgilenen, çok sayıda basit işlem birimlerinden oluşmaktadır. En basit şekilde, bir işlem birimi, bir girdiyi bir ağırlık kümesi ile ağırlıklandırır, doğrusal olmayan bir şekilde dönüşümünü sağlar ve bir çıktı değeri oluşturur. İlk bakışta işlem birimlerinin çalışma şekli yanıtıcı şekilde basittir. Sinirsel hesaplamanın gücü, toplam işlem yükünü paylaşan işlem birimlerinin birbirleri arasındaki yoğun bağlantı yapısından gelmektedir.

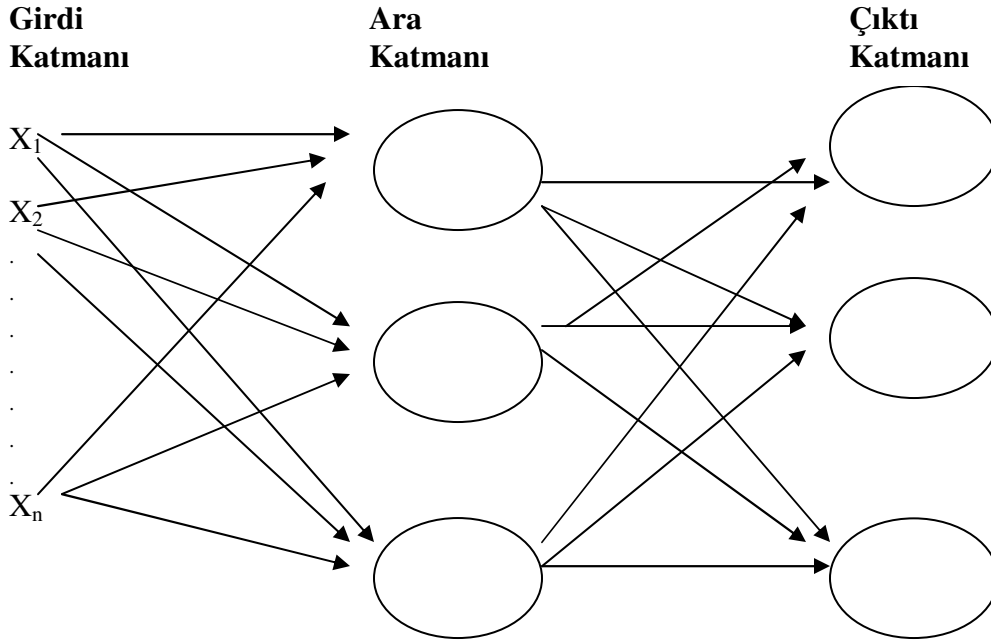
Çoğu YSA'da, benzer karakteristiğe sahip nöronlar tabakalar halinde yapılandırılır ve transfer fonksiyonları eş zamanlı olarak çalıştırılır. Hemen hemen tüm ağlar, veri alan nöronlara ve çıktı üreten nöronlara sahiptir. YSA'nın ana ögesi olan [matematiksel fonksiyon](#), ağın mimarisi tarafından şekillendirilir. Daha açık bir şekilde ifade etmek gerekirse, fonksiyonun temel yapısını ağırlıkların büyüklüğü ve işlem elemanlarının işlem şekli belirler. YSA'ların davranışları, yani girdi veriyi çıktı veriyeye nasıl ilişkilendirdikleri, ilk olarak nöronların transfer fonksiyonlarından, nasıl birbirlerine bağlandıklarından ve bu bağlantıların ağırlıklarından etkilenir [9].

Aşağıdaki şekilde genel yapay sinir ağı yapısı görünmekte olup (x) değerleri girdiler, (w) değerleri ağırlıklar ve (y) değeri de amaç fonksiyon değeri olmaktadır. Burada  $y = \sum(x_i \cdot w_i)$  olmaktadır.



**Şekil 8 . Genel yapay sinir ağı yapısı**

Aşağıdaki şekilde ise hatayı geriye yayma ağı görülmekte olup bu model yapay sinir ağı modelleri arasında en çok kullanılanlardan birisidir. Hatayı geriye yayma ağı, birçok katmandan oluşan tamamıyla bağlantılı ileri beslemeli ağıdır. Başka bir deyişle, bir katmandaki her işlem elemanı alt katmandaki bütün işlem elemanlarına bağlanır. Bununla birlikte, hiçbir geri besleme bağlantısı yoktur. Aynı zamanda, aynı katmandaki işlem elemanları arasında da hiçbir bağlantı yoktur. Ağ, genellikle, üç tip katmandan oluşur: herhangi bir işleme tabi tutulmaksızın girdi örneklerini sadece bir sonraki katmana aktaran girdi katmanı, bir veya birden fazla ara katman ve çıktı katmanı. Girdi katmanı dışındaki katmanlar bilgi işlemekten sorumludur.



Şekil 9. Hatayı geriye yayma ağı

#### 2.4 Çizelgeleme Problemi Çözümünde Daha Önce Yapılan Çalışmalar

Çizelgeleme problemleri, verimliliği esas alarak, kaynakların kullanıcılara belli kısıtlarla tahsis edilmesini hedefler. Bir zaman çizelgeleme problemi olan ders programı hazırlamada (çizelgelemede) hedef, her sınıf için açılacak derslerin

buluşma saatlerinin istenen kısıtlar sağlanarak belirlenmesidir. Örneğin, herhangi bir üniversitede bir bölümümün ders programının belirlenmesi demek açılan bütün derslerin saatlerinin belirlenmesi demektir. Doğal kısıtlardan biri ise aynı dönem derslerinin kesişmemesidir. Benzer şekilde, bir öğretim elemanın verdiği dersler aynı saate atanamaz. Bu tür kısıtlar bu tez çalışmasının 2.1 ve 2.2 bölümlerinde sistematik biçimde ele alınmıştır. Zaman çizelgeleme problemi, zor bir problemdir. Bu tür problemlerinin çözümü için değişik yaklaşımlar kullanılmıştır.

Çizelgeleme probleminin çözümünde, belirlenen kısıtları sağlayabilmesi ve en uygun zaman çizelgesinin elde edilebilmesi için çeşitli araştırma teknikleri kullanılmaktadır. Tavlama benzetimi, tabu arama ve kısıt bazlı mantık programlama bunlara birer örnektir. Zaman çizelgeleme problemleri için geliştirilen modeller genelde birbirinden oldukça farklıdır. Bu sebeple literatürde bulunan sonuçları karşılaştırmak zordur.

Çizelgeleme probleminin çözümüne yaklaşım ilk olarak 60lı yıllara dayanmaktadır. Yapılan bu çalışmada grafik renklendirme(graph coloring) algoritması kullanılmıştır.

Abramson, tavlama benzetim yöntemini kullanarak zaman çizelgeleme problemlerini çözmeye çalışmış ve altı farklı tavlama yöntemini karşılaştırmıştır.

Tavlama algoritmalarından birisi olan Monte Carlo yöntemi ile Abramson en iyi sonuca ulaşmıştır. Bu sonuca ulaşırken 15 öğretmen, 15 derslik kriterlerinden başlayarak 37 öğretmen 34 dersliğe kadar çeşitli kriterlerde algoritmasını çalıştırmıştır. En az öğretmen ve derslikte programlarının 11 saniye sürdüğünü, en çok olan 37 öğretmenli çizelgelemenin de 14 saat sürdüğünü rapor etmişlerdir. Programı geliştirirken Pascal programlama dilini kullanmışlardır[1]. Yapılan bu çalışma şu anda uyguladığımız yarı otomatik ders programı sisteminden çok küçüktür. Çünkü biz ders programımızı oluştururken 100 öğretim elemanı bulunmaktadır. Bununla beraber 20 sınıf vardır.

Hertz ve Schaerf, zaman çizelgeleme problemini tabu arama yöntemi ile çözmeye çalışmışlardır. Gerçek zamana çok uygun olan çizelgeleme problemlerinde bu algoritmayı kullanmışlardır[2;3]. Bu algoritmalarını C++

programlama dili kullanarak yazmışlardır. Bu dilde 2500 satır kod yazmışlar, bu kodun 1000 satırını optimum sonucu bulan fonksiyonun hesaplaması için, 500 satırını arama metotları, 500 satırını çizelgeleme için ve kalan 500 satırını da programın çalışması için gerekenler için kullanmışlardır. Bu programı 3 okulda çalıştırmışlardır. Bu okullar farklı ders sayısı, derslik sayısı ve kriterler içermektedir. Okullardan küçük olanında algoritmanın başarılı sonuçlar verdiğini diğer büyük iki okulda da problem yarattığını rapor etmişlerdir. Problem çıkan okullarda algoritmanın çalışması için bazı parametrelerin değiştiğini belirtmişlerdir[2,3].

Genetik Algoritma'ların zaman çizelgeleme problemlerinde kullanılması ilk olarak Colorni tarafından uygulanmış ve diğer tekniklere göre Genetik Algoritma ile daha iyi çizelgeler elde edildiği rapor edilmiştir. Bu teknikler gezgin satıcı yöntemi ve sağlanabilirlik(satisfiability) yöntemidir. Gezgin satıcı en kısa şekilde gidilecek yerlere ulaşmayı amaçlamaktadır. Sağlanabilirlik ise verilen  $n$  değişkenli bir mantık önermesinin, değişkenlere gerekli değerler atanarak sağlanıp sağlanamayacağına karar vermektir. [5]. 20 tane öğretmen, 10 sınıf ve haftada 30 saat ders olduğunu kabul ederek algoritmalarını kullanmışlardır. Her öğretmen ve her sınıf için yerleştirme yapmışlardır. Bir öğretmenin aynı saatte 2 farklı yerde ve aynı sınıfta aynı saatte iki farklı ders alması engellenmiştir.

Genetik Algoritmayı Erben ve Kepler 1995 yılında kullanmışlardır[6]. C ve prolog programlama dilleri ile yazdıkları program ile ders ataması yapmışlardır. Kendi üniversiteleri olan Konstanz Üniversitesinin gerçek verilerini kullanmışlardır. Algoritmalarının çalıştırılması 2500 nesil sonra durdurmuşlardır. Bu işlem 30 kromozom oluşturulması için hemen hemen yarım saat almıştır. Bu işlemin yinelemeli olarak çalıştırılması da 8,5 saat almıştır. İkinci adım olarak bütün üniversite verilerini kullanmışlardır. Bunlar 300 öğretmen, 80 sınıf, 1500 ders ve yaklaşık 100 derslik ile problemin çözüldüğünü söylemişlerdir. Bununla beraber kullanıcı beğenmediği kısımları kullanıcı arayüzünden değiştirebildiğini söylemişlerdir[6].

Engin ve Fiğlalı 2002 yılında çizelgeleme problemi için Genetik Algoritma çaprazlama operatörünün belirlenmesi amaçlı bir çalışma yapmışlardır. Bu çalışmada tamamlanma zamanlı akış tipi çizelgeleme problemlerinin GA ile çözümünde uygun çaprazlama operatörü belirlenmeye çalışılmıştır. İki farklı grup problem üzerinde yapılan toplam 1950 adet deney sonucunda, işlem süreleri yüksek olan çizelgeleme problemlerinde çok makine-çok iş problemleri için en uygun çaprazlama operatörünün Lineer Sıralı Çaprazlama (LOX) olduğu; iki makine-çok iş problemlerinde ise Sıralı Çaprazlama yönteminin (OX) iyi performans gösterdiği belirlenmiştir. GA'nın kombinatoriyel optimizasyon problemlerinde etkin bir şekilde kullanılabilmesi için GA'da kullanılan diğer parametrelerin optimize edilmesi gereğinin açık olduğu söylenmektedir. Bu yolla çözüm kalitesi ve performansı önemli ölçüde iyileştirilebildiği bildirilmektedir. Yalnızca çaprazlama yönteminin optimizasyonu yapılarak çok daha küçük nesil sayılarında veya daha düşük Cmax değerli çözümlere ulaşılabileceği gözlenmiştir.

2005 yılında akıllı ajanlar ile problemi çözmek için çalışan Dasgupta ve Khazanchi kullanıcı sorgu modülü, ajan yaratma modülü, ajan çalıştırma platformu kullanarak ve ajana öğretim yaptırarak bu problemin çözümünü gerçekleştirmişlerdir. Bu problemde karar desteği kullanan akıllı ajanlar vardır. Bu sistem katmanlı yapısı ile oluşturulmuştur. Bu katmanlar öğrenme motoru, eski verileri kullanarak karar vermeyi geliştirme ve uzman applet tabanlı kullanıcı arayüzü kullanmışlardır. Kompleks çizelgelemeler için dinamik olarak değişen karar verme mekanizması kullanmışlardır.[13].

2006 yılında Lien-Fu Lai, Nien-Lin Hsueh, Liang-Tsung Huang, Tien-Chun Chen yapay zeka ve uzman sistemler ile çizelgeleme probleminin çözümü için uğraşmışlardır. Uzman sistemler kullanarak çizelgelemenin daha esnek yapılabildiği söylenmiştir. Bu çalışma National Changhua University of Education' de Bilgisayar Bilimleri ve Bilgi Mühendisliği Bölümü'nde uygulanmıştır. Bununla beraber nesneye yönelik yazılım mühendisliği metotları kullanılarak geliştirilmiş olup bakımı da kolayca yapıldığını bildirmişlerdir[17].

## **2.5 Yazılım Piyasasındaki Ders Programı Yapan Benzer Yazılımlar**

Piyasada ders programı oluşturma işlevini gören birçok yazılım vardır. Bunlar işlevlerine göre değişik fiyatlar almaktadır. Aşağıda birkaçına örnek verilirse:

### **2.5.1 aSc Timetables**

(<http://www.asctimetables.com/>) Türkçe dil desteği olan başarılı ve kullanımı kolay Ders Dağıtım Programı olduğu iddia edilmektedir. Bütün işlemlerin alışkın olunan Windows pencere ve menüleri içinde sürükle bırak yöntemi ya da fare tıklaması ile halledilebildiği belirtilmiştir. Program 60 Ülkede 7500'ü aşkın okulda kullanıldığı iddia ediliyor ve Türkçe desteği de dahil olmak üzere 30 dil desteği mevcut olduğu belirtiliyor. Fakat 60 ülke ile ilgili hiçbir referans bulunmamaktadır. Farklı ülke ve dil kategorilerine göre sınıflandırılmış hazır ders programlarını incelenebilir olduğu ya da programın öğretici yardım dosyalarını kullanarak programın çalışma mantığını öğrenilebiliyor olduğu belirtiliyor. Ders programı hazırlarken sihirbaz desteğini kullanabilir ya da kullanıcı kendi istediği şekilde ders programını oluşturabildiği belirtiliyor. Bu program bir ders programı hazırlarken aşağıdaki adımlardan geçmektedir:

Öncelikle okul ile ilgili genel bilgiler giriliyor. Ders saatleri, tatil günleri, teneffüsler vb. Daha sonra okulda gösterilen dersler tanımlanıyor. Derslerle ilgili koşullar belirleniyor, sınıflar tanımlanıyor, öğretmenler tanımlanıyor, derslikler tanımlanıyor. Bilgiler girdikten sonra programa Ders Programını oluştur komutu veriliyor. Programın özellikleri olarak aynı sınıfa ve derse dilediğiniz sayıda öğretmen atayabilme, programı sürükle bırak yöntemiyle oluşturabilme, sınıf ve öğretmenler için çok sayıda koşul belirleyebilmenin mümkün olduğu iddia edilmektedir. Sınıflar, dersler, öğretmenler ile ilgili özelleştirilebilir ayrıntılı raporlar alınabildiği, hazırlanan Ders Programının HTML sayfaları ya da Excel formatında kaydedebildiği belirtilmektedir.

Program bütün özellikleriyle süre sınırlaması olmadan çıktılarında Applied Consultants logosu ile kullanılabilir. Tam sürümünün fiyatı ise 270 \$ dır. Bununla beraber girdileri tam olarak analiz eden, hataları bulan, otomatik güncelleme yapan sürümü Premium Edition olarak 500 \$ a satılmaktadır.

Bu yazılım genelde ilk ve orta öğretim kurumlarının ders programını oluşturmak için kullanılmaktadır. Bu nedenle bizim ele aldığımız sorunu çözememektedir.

### **2.5.2 iMagic Timetable Master**

(<http://www.imagictimetablessoftware.com/>) Kâğıt kalem kullanmadan kullanılacak olan program bütün gerekli bilgileri girerek çalıştırılmaktadır. Dışardan dosya alma fonksiyonu avantaj olarak kullanılırken; son versiyonunda Excel ve Web' e de otomatik olarak aktarma da mevcuttur. Çok hızlı bir şekilde ders programını oluşturulduğu belirtilmektedir. Çapraz eşleştirme ile birden fazla ders programını aynı anda oluşturabildiği, birbiri ile bağlantılı olan ders programlarını da beraberce doldurabildiği, yüzlerce ders, derslik oluşturulurken aynı anda veritabanını bir sonraki ders programı oluşturma için hazırlandığı, beğenilmediği takdirde bir ay içerisinde para iadesi mevcut olduğu iddia edilmektedir. Kullanıcı el kitabı mevcut olduğu, ara yüzünün karışık olmadığı belirtilmektedir. Otomatik yönlendirmeli ders programı oluşturması vardır(Wizard). Programın kullanılması için bilgisayarın özellikleri : Intel Pentium işlemci, Microsoft Windows 95 OSR 2.0, Windows 98 SE, Windows Millennium Edition, Windows NT 4.0 with Service Pack 5, Windows 2000 veya Windows XP işletim sistemi, 32MB of RAM, 24MB yeterli disk alanı olması gerekmektedir. Deneme versiyonu bulunmaktadır. Programı iMagicSoftware tarafından oluşturulmuş olup fiyatı sadece tek kullanıcının kullanabilmesi için 108 \$ dır.Liseler ve kolejler için uygundur.

Bu yazılım kendi web sayfalarında belirttiği üzere ilköğretim ve kolej ders programları için kullanılmaktadır. Bununla beraber restoranların çizelgelemesi için de kullanılmaktadır. Bu nedenle bizim ele aldığımız problem için uygun olmamaktadır.

### **2.5.3 Mimosa Scheduling Software**

([www.mimosasoftware.com](http://www.mimosasoftware.com)) Mimosa her çeşit okul ve üniversite için değişik çeşit ve boyutta kapasiteleri için uluslar arası ders programı oluşturma yazılımıdır. Bununla beraber bu yazılım konferans ya da iş yeri için çizelgeleme de yapabilmektedir. Bu sayede Mimosa iş ve endüstri çevresinde de tercih edilen bir programdır. Bu program hızlı, kullanıcı arkadaşı ve çok büyük kapasiteli olduğu

iddia edilmektedir. Bununla beraber program çok zengin eniyileme araçları ve kullanıcı ile beraber yapılabilen birçok çizelgeleme bilgisine de sahip olduğu iddia edilmektedir. Kullanıcı arkadaşı olan program yaygın kullanıcı hatalarını en alta indirmekte olduğu, kullanıcılara birçok ipucu vermekte olduğu ve kullanıcı sorularına gerek makalelerle gerekse direk olarak web sayfasından cevap vermekte olduğu belirtilmektedir. Kullanıcı birçok ders programını otomatik ya da manüel olarak oluşturabilmekte isterse interaktif olarak da oluşturabilmekte olduğu belirtilmektedir. Bu yazılım kullanıcıya kolayca uygulamalarını taşımasını sağlıyor olup isterlerse kopyalama ile başka bir uygulamaya da kolayca aktarabildiği belirtilmektedir. Kullanıcı kendi özel çizelgelemesini, ihtiyaç duyarsa içerdiği şablonlar veya örnek dosyalar ile oluşturabilmektedir. Bu yazılım Mimosa Software Ltd. tarafından geliştirilmiş olup kişi sayısına göre yazılım fiyatı değişmektedir. Yani en az 200 öğrenci için fiyatı480£ iken 1000 öğrenci için 1120 ve maksimum 5000 öğrenci için 4800 £ dur.

Bu uygulamanın her çeşit üniversite, okul ve kurum için kullanımı uygun olduğu söylenmektedir. Fakat fiyatı öğrenci bazlı olduğu için öğrenci sayısı arttıkça fiyatı da çok büyük meblağlara gelmektedir.

#### **2.5.4 Lantiv Timetabler 6**

(<http://www.lantiv.com>) Lantiv Timetabler tahminen en oturmuş organizasyon aracı olup, ilköğretim, lise ya da herhangi bir eğitim kurumu için kullanılabilir. Bu program bütün gerekli verileri (öğretmen, öğrenci, malzeme vb.) veritabanına kaydedip daha sonra hatalardan (çakışmalar) ayıklayarak kullanıcıya göstermektedir. Bütün istenilen bilgileri ara yüzünde gösteriyor olup kullanıcının bulmak istediği bilgilere kolayca ulaşmasını sağlamaktadır. İhtiyaç duyulduğunda iletişim bilgileri vardır. Bununla beraber web sayfası içerisinde bir forum bulunmaktadır. Bu sayede kullanıcıların birbiri ile haberleşmeleri ve sorunlarını kolayca paylaşmaları sağlanmaktadır. Bununla beraber örnek okul ders programı oluşturulma gösterilerek kullanıcıya nasıl kullanıldığını anlatmaktadır. Bu yazılım Lantiv International tarafından geliştirilmiş olup tek kişilik kullanım için 250 \$ dır. Kişi sayısı arttıkça fiyat değişmektedir.



Bu yazılım ilköğretim ve lise programlarını oluşturmak için uygundur. Bu nedenle bizim ele aldığımız problemi çözememektedir.

### **2.5.5 Ders Dağıtım Programı**

(<http://www.evrenselyazilim.com.tr/>) İlk ve ortaöğretim haftalık ders programlarını bilgisayar ortamında pratik bir şekilde yapma imkânı sağlayan bu program; bir derse birden çok öğretmen atama, birden çok sınıfın dersini aynı saate getirip ayrı ayrı öğretmene atama imkânı sağladığı iddia edilmektedir. Ders almamış veya fazla ders verilmiş öğretmenleri tespit edip, dersi boş geçen sınıfların takibinde veya dengesiz ders dağılımında kullanıcıya uyarı verdiği belirtiliyor. Haftalık program yapıldıktan sonra üzerinde istenilen değişiklikler yapılabilmektedir. Laboratuvar, spor salonları ve sınıf planları oluşturulabilmektedir. Program; mazeretli ve izinde olan öğretmenlerin devamına göre yeniden düzenleme imkânıyla yöneticiye kolaylık sağlamaktadır. Programda tasarlanan tüm çalışmalar yazıcıdan çıkarılabilmektedir. Üretici tarafından ihtiyaç duyulan her konuda ücretsiz olarak güncellenme sağlanmaktadır. Bu programın üretici firması Evrensel Yazılımdır. Doğu ve Güneydoğu Anadolu bölgelerine indirim yapmaktadırlar.

Bu uygulama da ilköğretim ve lise ders programlarını çözme yetisindedir. Bu nedenle bizim ele aldığımız problemi çözememektedir.

### **2.5.6 Dersmatik**

(<http://www.dersmatik.com>) Bütün okul, dersane, üniversite ve özel eğitim kurumları kullanılabilir. Türkiye’de ve dış ülkelerde 1700’ü aşkın okulda kullanılan, haftalık ders gün ve saatlerine, öğretmenleri ve dersleri otomatik olarak yerleştirebilen bir programdır. Sağladığı kolaylıklar içerisinde 7 günlük program yapabilme, 24 saatlik program hazırlayabilme, istenilen saate öğle yemeği koyabilme, aynı sınıfa aynı saat için birden fazla öğretmen atayabilme, aynı öğretmeni aynı saat için birden fazla sınıfa atayabilme, teker teker bütün öğretmenlerin istenilen saatlerini kapayabilme, teker teker istenilen sınıfın istenilen saatlerini kapatabilmedir. Bununla beraber sınıfları öğlenci sabahçı şeklinde ayırabilme, istenilen derslerin yerleri kilitleyerek başka saatlere

yerleşmesini engelleyebilme, sürükle bırak yöntemi ile istenilen dersi çakışma olmadan bir başkası ile değiştirebilme, programı tam ekranda görebilme ve kâğıda programın bütün ayrıntılarını çıktı olarak alabilmedir. Programın demo sürümünde 50 ders saatlik deneme yapabiliriz. Bir yıllık sınırsız kullanım ücreti 84 liradır. Bu süre içerisinde %100 ders yerleştirme garantisi ve eğer bir problem olursa %100 geri ödeme garantisi olduğu iddia edilmektedir. Üye olduğu zaman yardım alabileceğiniz telefon numarası, e - posta adresinize gönderildiği belirtilmektedir.

Bu uygulama bütün eğitim kurumları, üniversitede kullanıldığı bildirilmiştir. Fakat kendi web sayfalarında yaptığım araştırmalara göre üniversite için Türkiye’de referans göremeyişim üniversiteler için çokta uygun olmadığını göstergesidir. Buda bize gösteriyor ki bizim ele aldığımız problemin çözümü için uygun değildir.

### **2.5.7 Progmatic Magic**

<http://www.alfabe.biz>

Resmi ve özel okullar ile dershanelerin ders programı oluşturmadır. Kolay kullanımı ile kullanıcının Öğretmen, Sınıf ve Ders tanımlamalarını birkaç dakikaya sığdırdığı iddia edilmektedir. Ders programında istediğiniz dersleri istediğiniz saatlere yerleştirme, sabitleştirme işlemleri yapabildiği ve öğretmenlerin özel şartlarını ( haftanın belirli günlerinde derse girme, belirli saatlere ders verme, vb.) da programa dahil edebildiği belirtilmektedir. Aynı zamanda aynı grup derslerin ( Örneğin Matematik ve Geometri gibi) derslerin aynı güne gelmesini de engelleyebildiği belirtilmektedir. Programın otokontrol sistemi sayesinde aynı dersin aynı gün içerisinde bir sınıfa belirtilen sayıdan (örneğin; Türkçe dersinin aynı sınıfa bir günde 3 saat vermesini ) fazlasının verilmesini engellene bildiği belirtilmektedir. Derslikler oluşturarak, dersliklerin ders programını da yapabileceğiniz bir program olan Progmatic Magic.

ek ders işlemleri menüsünde öğretmenlerin ek dersleriyle ilgili işlemleri de yapabildiği belirtilmektedir. Verilerinizi çok küçük boyutlardaki dosyalarda yedeklenebildiği belirtilmiştir. Öğretmenlerin ders programı, sınıfların ders programı, sınıflarda okutulan dersler, hangi öğretmenin hangi sınıfın hangi

dersine girdiđini gsteren liste, đretmenlerin nbet izelgeleri, nbet yerlerinin listeleri, dersliklerin ders programı ve ek ders iřlemleri ile ilgili ıktıların alınabildiđi iddia edilmektedir.

Bu yazılım da ilköđretim okulları ve dersaneler iin uygundur. Bizim ele aldıđımız problemi özememektedir.

### 3 YARI OTOMATİK DERS PROGRAMI SİSTEMİ (yODep v.1)

Yarı otomatik ders programı sistemi Başkent Üniversitesi Mühendislik Fakültesi verileri taban alınarak yapılmış olup herhangi bir üniversitenin bir fakültesinde uygulanabilir bir sistemdir. Tam otomatik değildir çünkü ders programı oluşturulmaya başlamadan önce manüel olarak Tarih, Türkçe gibi bazı derslerin saatleri doldurulmuştur. Yani saati sabit derslerin gün saat derslik yerleştirilmesi yapılmıştır. Yarı otomatik ders programından söz ediyorsak, yetkilerin bir kısmının da sisteme devredilmesini kabul etmiş oluyoruz. Programın kısa ismi kelimelerin baş harflerinden oluşturulmuş olup ilk versiyonu olduğu için sonunda v.1 bulunmaktadır.

Bu sistem oluşturulmadan önce sistemin paydaşları(aktörleri) için anket yapılmıştır. Anket sonucuna göre sistemin yönleneceği sağlanmıştır. Sistem aktörlerindeki kasıt, ders programında etkilenen insanlardır. Yani öğretim üyesi, öğrenci, bölüm başkanı, dekan... vb. Tasarlanan sistem küçük – orta boyutlu bir fakülte düzeyinde tüm paydaşlar (aktör, akıllı ajan) için eniyileme yapacaktır.

Bu tez çalışması sunumunda 33 tane derslik, 166 tane atanacak ders, 85 tane öğretim elemanı, 40 ders saati vardır. Bu rakamlar ile baktığımızda bu tez çalışmasında ilk atama anında 18.600.000 civarında durum söz konusudur.

(33 x 166 x 85 x 40 = 18.600.000)

Ama bu ihtimalleri yapılan anketler ile azaltılmıştır. Bununla beraber akıllı ajanlar bu kadar durumdan seçim yapmak için değil, bu durum sayısını azaltmak için kullanılmıştır. Bu çizelgeleme problemi polinom zamanda çözülemeyen tam (NP-complete) bir problemdir. Doğal kısıtlardan biri ise, aynı bölümün aynı dönem derslerinin kesişmemesidir. Benzer şekilde, bir öğretim elemanın verdiği dersler aynı saate atanamaz. Diğer tercih nedenleri hazırlamış olduğumuz anket çerçevesinde ve 2.1, 2.2'de açıklanacaktır.

Bu tez çalışmasında 18.600.000 civarında görünen ilk analizde yerleştirme ihtimali aslında bu kadar büyük rakam değildir. Çünkü:

- Bir ders seçilirken bu dersin tercih edilen derslikleri mevcuttur. Bu tercih edilen dersliklere göre bizim 33 olan derslik sayımız otomatik olarak mühendislik fakültesi herhangi bir bölümü için 6 civarına düşmektedir (Çünkü bölümlerin alışkanlıkları vardır ve öğrencileri her iki saatte bir uzak dersliklere gönderme insan eli ile yapıldığında oluşturulmamaktadır).
- Bununla beraber 85 tane öğretim elemanı bütün derslere girebilecek donanımda değildirler. Bir dersin tercih edilen öğretim elemanları mevcuttur. Bu nedenle bir ders atanırken aslında öğretim elemanı 85 değil yaklaşık olarak 2'dir(Başlangıç düzeyi derslerde 5-6 seçenek var iken bazı derslerde 1(bir)'dir.).
- Bununla beraber 166 tane aranacak ders sayısı bölüm başkanının ya da dışarıdaki bölümlerden yapılan otomatik atamalar ile yaklaşık 150 olmaktadır(Türkçe, tarih önceden atanıyor.).
- Bununla beraber 40 saat gün dilimimiz olmasına rağmen hepsine yerleştirme yapamıyoruz. Çünkü arka arkaya gelecek olan ders saatleri için en az 2 saat dilimi olmalıdır. Bu da bize sabah 2 öğleden sonra 3 saat olduğunu göstermektedir(Öğle saati de tercih edilmemektedir.).
- Bununla beraber en kalabalık sınıf ve en sıkışık hocaya baktığımızda aslında ilk seçimde seçilecek atanacak ders 150 değil 10 civarında olduğu görülmektedir.

Yukarıdaki rakamlar ile tekrar ilk ders ataması için olasılık hesapladığımızda:

$$6 \times 2 \times 10 \times 5 \times 5 = 3000$$

olasılık çıkmaktadır. Bu da bize gösteriyor ki yapay zekâ yöntemlerini ilk düzeyde kullanarak, olasılığımızı 6200 kat azalmaktayız. Yinede 3000 seçenek ile başlayan ilk atama çok önemli bir problemdir.

### **3.1 Platform Bağımsız Tasarım Kararları**

**Karar 1.** Sistemin tüm dökümleri A4 sayfa için tasarlanacaktır ve dökümler tercihan daraltılmadan tek tuş ile çıkacaktır.

- Karar 2.** Ders kodları Başkent Üniversitesi gerçeği içinde alınacak, mümkün olan yerlerde küçük harf büyük harf dönüşümü yapılarak kullanıcının ufak hataları tolere edilecektir.
- Karar 3.** Öğretim üyelerinin unvanları isim ile birleştirilmeyecektir.
- Karar 4.** İki basamaklı ders şubesi, üç basamaklı ders kontenjanı ve üç basamaklı sınıf mevcudu olacaktır.
- Karar 5.** Öğretim üyelerinin yönetim sorumlulukları ile ya da başka nedenler ile gelemeyeceği ya da tercih etmeyeceği yerleri program oluşturulmaya başlamadan önce belirtilecektir. Bu bilgi alımı için web ara yüzü kullanılacak olup bu web ara yüzünde Java Applet kullanılacaktır.
- Karar 6.** Kalabalık olan dersler öncelikle yerleştirilecektir, çünkü büyük sınıfın azlığından kaynaklanan önceliği olacaktır.
- Karar 7.** 5 ve 6 numaralı maddeden bir katsayı bulunup program oluşturulmaya başlamadan önce hesaplanacaktır. Yani ön analiz yapılacaktır. Eğer program oluşturulması imkânsız ise gerekli değişiklikler için kullanıcı uyarılacaktır. Bununla beraber yoğunluk oranı %80 civarında ise yine gerekli değişiklikler yapılacaktır.
- Karar 8.** Öğretim üyesinin ders programı otomatik olarak çıkacaktır. Dönem ortasında manüel olarak değişiklik yapıldıysa da ders programı tekrar dökülebilecektir.
- Karar 9.** Derslik ve laboratuvar doluluk tablosu dökülecektir. Bunların altında fakülte yöneticisinin tatmin oranı da verilecektir.
- Karar 10.** Öğrenciler için sınıfların zaman planı dökülecektir ve bunun altında öğrencinin tatmin katsayısı verilecektir.
- Karar 11.** Ders kodu girilince ders adı otomatik olarak gelecektir.
- Karar 12.** Öğretim üyesinin ders programı görüntülenirken farklı kriterlere göre arama tetiklenecektir. Yani ada, soyada, sicil numarasına göre öğretim üyesinin araması yapılacaktır.
- Karar 13.** Öğrenci numarası 8 karakter, öğretim üyesi 4 karakter olarak numaraları girilecektir.
- Karar 14.** Sınıf kodu ile derslik kodu ayrılacaktır.

**Karar 15.** Ön analiz yapılacaktır. Yani kaç ders atanacak, bu derslerin toplam saatleri ne olacaktır hesaplaması atama öncesinde yapılacak ve imkânsıza yakınsa gerekli değişiklikler için kullanıcı uyarılacaktır. Yani sıkışıklığın ne aşamada olduğunu bulmalıyız.

**Karar 16.** Asistanlar kesinlikle gelemeyeceği saatleri işaretleyemeyecektir.

**Karar 17.** Bölüm başkanı ve dekan, öğretim üyelerinin ders programına ve kapattığı saatlere yerlere müdahale edebilecektir.

**Karar 18.** Dersin ders yapısı 2+2 ise bu parçalar farklı güne konulacaktır.

**Karar 19.** Atama öncelikleri dinamik olarak değişecektir. Burada ölçüt doluluk oranlarıdır.

**Karar 20.** En kalabalık olan ders ve zaman planı en sıkışık olan hoca çarpımına göre atamaya başlanacaktır.

**Karar 21.** İlk önce programı düzensiz olmayanlar (yani normal öğrenciler) için sistem optimize edilecektir.

**Karar 22.** Bu yılki dersler girilirken önce bölüm seçilecek ve daha sonra da o bölüme ait öğretim elemanları seçimlik menüde(combobox) üst sıraya gelecektir.

**Karar 23.** Açıklanması gerekenler kullanıcı ara yüzünde (\*) ile açıklanacak.

**Karar 24.** Beklenen düzensiz öğrenci sayısı ilgili ders için sisteme girilecektir.

**Karar 25.** Ekranların kullanımıyla ilgili her şey kullanıcı el kitabı ile verilecek.

**Karar 26.** Kaynak kodun %20 si kadar program içi belgeleme olacaktır.

**Karar 27.** Öğrenci sayısı fazla olduğu için tatmini genel tatminin 1/2'si olacaktır.

**Karar 28.** Atamaya başlamadan önce sıkışıklık oranı %90'dan büyükse atamaya başlanmayacak

**Karar 29.** Öğretim üyeleri sıkışıklık değerlerini girerken toplamda -60 değerinden fazla negatif değer girilirse bu değer oranlanarak toplamda -60 yapılacak ve bütün değerlere yansıtılacaktır.

**Karar 30.** Gerekli olan dersler atanmadan önce manüel olarak atanması gereken dersler atanacaktır.

**Karar 31.** Ders listesi istenildiğinde çıktı olarak alınacak.

**Karar 32.** Gerekirse düzensiz öğrenci tatmin düzeyi azaltılarak hesaba katılacaktır.

**Karar 33.** Tez çalışmasının sınaması için, 1000 tane sahte öğrenci verisi üretilecektir.

**Karar 34.** Lisans, yüksek lisans, kısmi zamanlı öğretim üyesi, tam zamanlı öğretim üyesi, bina sorumlusu, bölüm başkanı, öğrenci dekanı için tercih ve beklentilerinin alınabilmesi için anket yapılacak ve bunlara göre tercihler ayarlanacaktır.

**Karar 35.** Düzenli öğrenciler için, bölüm ve sınıf bazında sabit ders programı dökülecektir.

**Karar 36.** Program oluştururken belli yerlerde durdurulup yarım ders programı alınabilecek, gerekirse manüel değişiklikler yapılacaktır.

**Karar 37.** Programı test ederken 4 farklı veritabanı doluluğu oluşturulacaktır. Bu doluluklar %60 - %70 sınırlarında, %70 - %80 sınırlarında, %80 - %90 sınırlarında ve %90 - %100 sınırlarında olacaktır. Bu veritabanlarına göre sistemin nasıl çalıştığını ve ne gibi tepkiler verdiği izlenecek, tez çalışması sonuçları kendi içerisinde eleştirilecek, kısmen dengelenecektir.

**Karar 38.** Farklı dolulukta oluşturulan veritabanlarına göre yodep sistemi 9. dilimi (saat 17:00 ile 18:00) açacak ya da açmayacak; atamaya başlayacak ya da başlamayacak gibi ön kararlar verecektir. %90'ın üzerindeki doluluklarda sistem atama yapmayacaktır ve kullanıcıyı yeni derslik eklemeye zorlayacaktır. Yeni derslik ya da derslikler eklendiğinde doluluk %90 altına düşerse atama yapabilecektir.

**Karar 39.** Program oluşurken atadığı derslere göre anlık sezgisel fonksiyon değerleri kontrol modunda yazdırılacaktır. Bu işlem normal işletim modunun parçası değildir. Bununla beraber o ana kadar yerleştirilen dersler için sezgisel fonksiyon toplam değerleri normal işletim modunda yazdırılacaktır.

**Karar 40.** Program oluşturulurken atadığı derslik saat dilimi dışında atayabileceği alternatif saat dilimleri de kullanıcıya gösterilecektir. Bununla



beraber bu gösterilen listeler kullanıcı bilgisayarında bir dosya olarak kaydedilecektir.

**Karar 41.** Sınıf içerisinde bulunan materyaller ve sınıfın özellikleri şu şekilde sınıflandırılmıştır:

- Normal sınıf kapasite 25,
- Çok bilgisayarlı sınıf,
- Elektrik gereçleri olan sınıf,
- Biyomedikal gereçleri olan sınıf,
- Endüstri Mühendisliği gereçleri olan sınıf,
- Fizik laboratuvarı,
- Kimya laboratuvarı,
- Makine gereçleri olan sınıf,
- Normal sınıf kapasite 50'ye kadar,
- Normal sınıf kapasite 75'e kadar,
- Datashowlu sınıf kapasite 25,
- Datashowlu sınıf kapasite 50,
- Datashowlu sınıf kapasite 75.

**Karar 42.** Her ders için tercih edilen 3 adet derslik bulunmaktadır.

**Karar 43.** Sınıflandırılan sınıf içeriklerine göre derslerin ataması tercih edilen sınıfa ya da o sınıfta yer yoksa içeriği aynı olan herhangi bir başka sınıfa atama yapılacaktır.

**Karar 44.** Her ders için eğer laboratuvar kredisi mevcut ise 1 – 3 adet tercih edilen laboratuvar bulunmaktadır.

### **3.2 Gerçekleştirilen Sistemin Zekâ Modelleri (Akıllı Ajanlar)**

Akıllı ajan: Her birisi kendi bakış açısından talepleri olan modüler tercih belirtirler. Eğer sistemde birden fazla akıllı ajan modellenmiş ise bunlar arasında dengeyi oluşturan bütünleştirme modülü bulunmalıdır. Her bir akıllı ajan Sezgisel fonksiyonu  $H(S_i)$  alt parçaları olan  $h_1(S_i)$ ,  $h_2(S_i)$ , ...,  $H_n(S_i)$ 'i oluşturur.

Sonuçta  $H(S_i) = c_1 * h_1(S_i) + c_2 * h_2(S_i) + \dots + c_n * h_n(S_i)$  biçiminde modellenir. Bu tez çalışmasında en büyük  $c_j$  katsayısı öğrenci tatmini içindir.

### **3.2.1 Fakülte Dekanı:**

Bu kişi bir fakültenin dekanı olmaktadır. Görevi fakülteadaki bölümlerin yapısı, işleyişi, eksikleri, ihtiyaçları ile ilgilenmektedir. Bu kişi okulun fakültesini yönettiği için aslında okullun genel yapısına uygun şekilde derslerin yerleşmesini ister. Bu nedenle ders yerleşiminin 09 ile 17 saatleri arasında ne kadar yoğun olduğu ile ilgilenir.

Fakülte dekanı için bir ekran bulunmaktadır. Bu ekrandan ders ataması yapılmasından önce İngilizce, tarih, Türkçe gibi derslerin atamasını yapacaktır.

### **3.2.2 Öğrenci Dekanı:**

Bu kişi öğrencilerin okul ile aracısı olarak düşünülebilir. Yani öğrencilerin ne gibi ihtiyaçları olduğu, öğrencilerin nasıl daha huzurlu olacağı, nasıl öğrencilerin motivasyonun daha yüksek seviyelere getirileceği ile ilgilenir. Bu nedenle öğrencilerin çoğunun öğle saatlerinde 2 saat dolaylarında yemek için boş zaman olmasını tercih eder. Bununla beraber öğrencilerin çok fazla dersler arasında boşluk olmasını tercih etmez.

### **3.2.3 Fakülte Yöneticisi ya da Bina sorumlusu:**

Bu kişi bina ile ilgilidir. Yani binanın kalorifer, ışık, temizlik vb. ihtiyaçlarının sorumlusu bu kişidir. Her gün dersliklerin temizliği yapılacağı için, güvenlik açısından ışıkların gece yanmasını en aza indirmeye çalışır. Bu nedenle derslerin 09 ile 17 arasındaki yoğunlaşmasını tercih eder ve derslik ile laboratuvarlardaki son dersin profili ile saatine bakar.

### **3.2.4 Bölüm Başkanı:**

Bu kişi bölümün akademik yöneticisidir. Bölümdeki dersler, öğrencilerin bu derslerden en fazla fayda sağlamasını amaçlar. Bununla beraber hangi öğretim üyesinin hangi derse hangi zaman diliminde olması ile ilgilenir. Bu nedenle öğretim üyesinin ders atamasını istemediği alanları değiştirebilir. Derslerin sabah ya da öğleden sonra olması gerektiğini seçer.

Bölüm başkanı ekranı bulunmaktadır. Bu ekrandan bölüm başkanı kendisi ya da görevlendireceği kişi derslerin girişini, sabah ya da öğleden sonra olması isteğini, öğretim üyesinin sıklıklarını kontrol edebilecektir.

### **3.2.5 Öğretim Üyesi**

#### **3.2.5.1 Tam Zamanlı Öğretim Üyesi:**

Bu kişi dersleri verecek kişidir. Yani ilgili ders ile ilgilenir. Dönem başında istemediği saat dilimlerini web üzerinden kullanabilen ilgili ekrandan kapatılarak o saat dilimlerine ders atmasını engeller. Bu kapatacağı saatleri en fazla 6 saat dilimi olacak şekilde kısıtlanmıştır. Bununla beraber bölüm başkanı bu kapattığı saat dilimlerini değiştirebilir. Tercih ettiği zaman planını program oluşturulmaya başlamadan önce verir. Bu tercihleri ile eşleşme oranında tatmin olur.

Tam zamanlı öğretim üyesi için hazırlanmış olan ekran bulunmaktadır. Bu ekrandan öğretim üyesi sıklık bilgilerini girebilecektir. Yani buradaki kasıt öğretim üyesi gelemeyeceği yerleri kapatılarak bu dilimlere ders konulmasını engelleyeceklerdir.

#### **3.2.5.2 Kısmi Zamanlı Öğretim Üyesi:**

Bu kişi okul mensubu değildir. Dışardan günü birlik ya da derslerin saatine göre gelebilir. Bu kişilerin zaman planlarının web üzerinden girilebilmesi bölüm başkanına zaman kazandırmaktadır. Bölüm başkanı, bazen kapatılabilecek ders saati sayısını (-60)'dan (6\*(-10)) daha fazlaya çıkarabilir. Çünkü tam zamanlı öğretim üyesine benzer istekleri dışında, bazen ders için tek gün isteği olabilir.

Kısmi zamanlı öğretim üyesi için hazırlanmış olan ekran bulunmaktadır. Bu ekrandan öğretim üyesi sıklık bilgilerini girebilecektir. Yani buradaki kasıt öğretim üyesi gelemeyeceği yerleri kapatılarak bu dilimlere ders konulmasını engelleyeceklerdir.

### **3.2.6 Asistan:**

Bu kiři öđretim üyesi ile derslere girebilir. Bazen öđretim üyesi yerine de girebilir. Bu nedenle öđretim üyesinin programının aynısı asistanında da mevcuttur. Her gününün tam dolu olmasını tercih etmez.

### **3.2.7 Öğrenciler:**

Bu kiři derslere girecek olan kiřidir. En fazla tatmin edilmesi gereken kiři budur. Çünkü sayıları fazladır. Sabah 09 ile akşam 17 arasında derslerinin yerleşmesini ister. Öğle yemek saatinin olmasını ister. Dersler arasında 1den fazla saat boşluk olmasını istemez. Bununla beraber biyolojik saatleri nedeni ile 12:00 – 14:00 arasında bir yemek aralarının olmasını, çok erken saatte ve çok geç saatte derslerinin olmaması gibi geçerli beklentileri vardır.

Bu beklentiler varsayımlar yapmak yerine ekte verilmiş olan bir anket ile tespit edilmiş ve öğrencileri temsil eden “akıllı ajan (intelligent agent)” bu ankete göre programlanmıştır.

### **3.2.8 Yarı Otomatik Ders Programı Sistemi İçin Yapılan Anket ve Sonuçları**

Bu ankete üniversitede üçüncü ve dördüncü sınıfta okuyan öğrenciler, yüksek lisans öğrencileri ve öđretim üyeleri katılmıştır. 21 üçüncü sınıf öğrencisi, 26 dördüncü sınıf öğrencisi, 7 yüksek lisans öğrencisi ve 3 öđretim görevlisi katılmıştır. Anket dört puanlık skala üzerinden değerlendirilmeleri istenmiştir.

Skala şöyledir:

A: Tamamen katılırim

B: Kısmen katılırim

C: Kararsızım

D: Katılmıyorum

Anket soruları ekte verilmiştir.

Anketin sonuçlarına değerlendirirken 3.sınıf öğrencilerinin seçtiđi seçeneklere bakıldığında zor derslerin sabah olmasını, dersinin ilk saat diliminde başlamasını, ders yükünün 4 güne dağılmasını, akşam ders bitişinin de erken olmasını daha

çok sayıda öğrenci istemektedir. Sosyal derslerin ve İngilizce dersinin sabah olmasını istememektedirler.

4. sınıf öğrencilerinin seçtiği seçeneklere bakıldığında akşam bitişinin erken olmasını, zor derslerin sabah olmasını, dersinin 2. saat diliminde başlamasını ve ders yükünün 4 güne dağılmasını en çok istemişlerdir. Sosyal derslerin ve İngilizce dersinin sabah olmasını istememektedirler.

Yüksek lisans öğrencilerinin seçtiği seçeneklere bakıldığında öğrencilerin hepsi Cuma günü 18:00 da başlayan dersi kabul ettiği görülmektedir. Bununla beraber kampus içerisinde çalışan öğrenciler 17:00 da yüksek lisans dersinin başlamasını isterken, kampus dışarısında çalışan öğrenciler bunu istememektedir. Cumartesi ve Pazar günü ders istenmemektedir.

Öğretim elemanlarına baktığımızda hepsi dersinin sabah ilk saat diliminde başlamasını istemektedir. Haftanın 5 gününe dağılan ders istememektedirler.

### **3.3 Platform Bağımlı Tasarım Kararları**

#### **3.3.1 Tasarlanan Sistemin Önerilen İnsan Bilgisayar Etkileşimi**

yoDep Sistemi'nin Java programlama dili ile yazılmasına karar verilmiştir. İstenirse uygulama yazılımı gelecekte çok katmanlı mimari şekline dönüştürülebilecektir. Yazılım kullanıcı(client) ve sunucu(server) olarak iki katmanlı olarak yazılmıştır. İstenirse araya uygulama sunucusu (Application Server [Oracle Application Server (oc4j)]) kurulup çok katmanlı mimariye dönüştürülecektir. Bu yazılımın geliştirilmesi için Oracle JDeveloper 10.1.3.0.0 ara yüzü kullanılmıştır. Veritabanı olarak ORACLE Database 10g Express Edition ilişkisel veritabanı kullanılmıştır. Bununla beraber, Java kodu ile veri tabanı işlemlerini yapabilmek için nesne/ilişkisel eşleme aracı olan Hibernate teknolojisi kullanılmıştır.

#### **3.3.2 Sistem Mimarisi**

Bu projede ara yüzlerin bulunacağı bir kullanıcı tarafın (client) bulunacağı java Project(jpr); bir de veri tabanı işlemlerinin yapılacağı yani veri tabanı bağlantıları, SQL cümlecikleri, sorgu sonucunda gelenlerin bir yere kaydedilmesi(nesnelere)

gibi kısımların olacağı yönetici sınıfı(manager) ile veri aktarma nesnelere(data transfer object [DTO]) nin bulunacağı server kısmından oluşmuştur. DTO'lar olarak kullanacağımız nesnelere bilgilerin ya da insan bilgilerinin tutulacağı,içerisinde getirici ve yükleyici (get ve set) metotlarının bulunacağı nesnelere dir.

Veritabanı işlemleri Hibernate ile yapılacağı için veritabanı bilgilerinin veritabanına tanıtılması gerekmektedir. Bu bilgilerin içerisinde veritabanı adresi, veritabanı hizmet numarası, kullanıcı adı ve şifre bulunmaktadır. Bu işlem için;

```
<propertyname=  
"hibernate.connection.driver_class">oracle.jdbc.driver.OracleDriver  
</property>  
  <propertyname=  
"hibernate.connection.url">jdbc:oracle:thin:@localhost:1521:xe  
</property>  
<property name="hibernate.connection.username">tez</property>  
<property name="hibernate.connection.password">tez</property>  
<propertyname=  
"hibernate.dialect">org.hibernate.dialect.Oracle9Dialect  
</property>  
<property name="show_sql">>false</property>
```

komutları kullanılmıştır. Bu komutlar hibernate.config.xml dosyasında tutulmaktadır. Veritabanından bağlantı (connection) alınacağı zaman bu xml dosyası çağırılmaktadır.

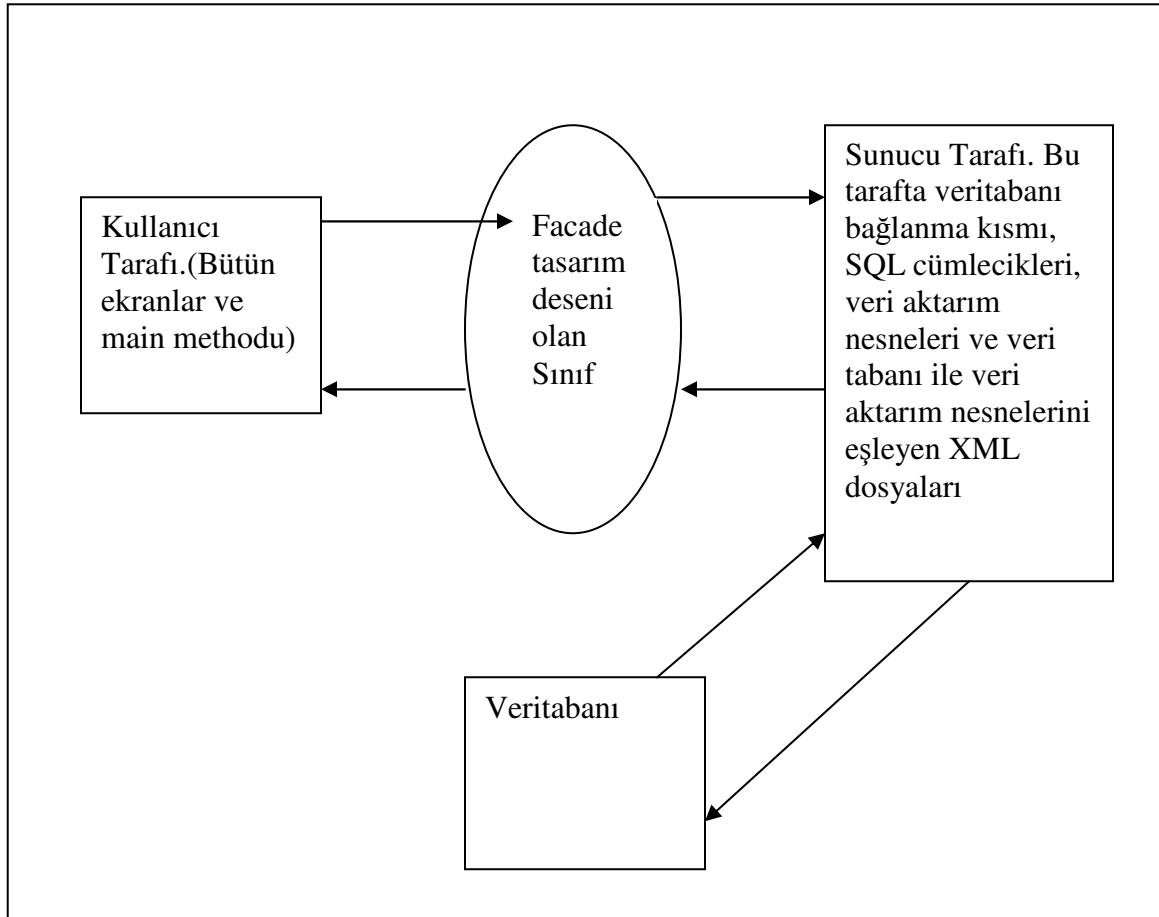
Sistem veritabanına veri girişi (insert) yaparken birincil anahtar (primary key) seçimi için bir veritabanında sequence oluşturulmuştur. Döngüsüz oluşturulan bu sequence 0 dan başlayıp birer birer artacaktır. Hibernate ile birincil anahtarın değerini almak için ;

```
<generator class="sequence">  
  <param name="sequence">SQ_TEZ</param>  
</generator>
```

komutları kullanılmıştır.

Bununla beraber Öğretim elemanı seçeneklerinin girilebilmesi için JAVA APPLET yapılmıştır. Bu Applet'in uygulanabilmesi için çeşitli güvenlik aşamalarından geçilmiştir. Bunlardan ilki Applet imzalamadır. Applet yazılımcı tarafından jar şekline getirilerek imzalanmıştır. Ayrıca Applet'in çalışabilmesi için araya uygulama sunucusu katmanı konulmuştur.

Kullanıcı ile sunucu tarafı normal masaüstü bilgisayar için Facade tasarım deseni kullanılarak bağlanmıştır.



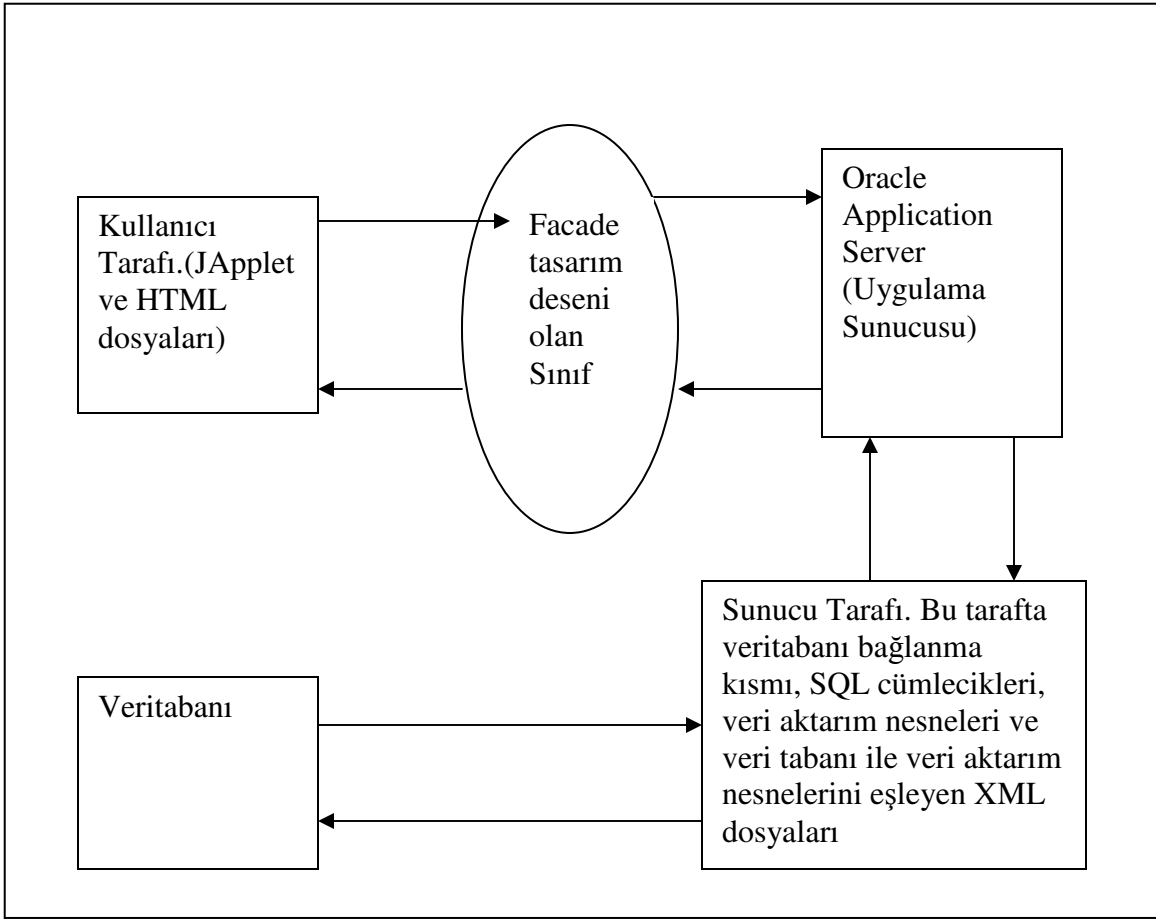
Şekil 10. Yazılım normal masaüstü bilgisayar için tasarımı

İnternet'ten öğretim elemanlarının kullanması için hazırlanan JApplet'te ise kullanıcı ile sunucu Oracle Application Server ile bağlanmıştır. Kullanıcıdan yine Facade deseni olan sınıf aracılığı ile Session Bean' e bağlanıp Session Bean da sunucu tarafına gitmektedir. Bu bağlantı işlemini gösteren fonksiyona bakarsak;

```
private static Context getInitialContext()
    throws NamingException
{
    Hashtable env = new Hashtable();
    // Standalone Oracle Application Server (OC4J) connection details
    env.put(Context.INITIAL_CONTEXT_FACTORY,
        "oracle.j2ee.rmi.RMIInitialContextFactory" );
    env.put( Context.SECURITY_PRINCIPAL, "oc4jadmin" );
    env.put( Context.SECURITY_CREDENTIALS, "admin" );
    env.put(Context.PROVIDER_URL, "ormi://88.249.4.109:23791/tezserver");
    return new InitialContext( env );
}
```

Yukarıdaki fonksiyonda uygulama sunucusunun adresi, sunucuya bağlanmak için kullanılan kullanıcı adı ve şifre ve bağlanmak için hangi JAVA kütüphanesi kullanılacaksa onun adresi bulunmaktadır.





**Şekil 11. Yazılım internetten kullanım için tasarımı**

Server kısmında temel olarak, [sql](#) kullanılarak bir [veritabanı](#) sunucusu üzerinden yapılan işlemlerin java fonksiyonları ve kendine özgü [hql](#) (Hibernate query language) ile objeler üzerinden, veritabanı sunucusundan bağımsız olarak yapılması mantığı ile ortaya çıkmış olan HİBERNATE teknolojisi kullanılacaktır. Yazılım belgeleme yöntemi olarak da kolay belgeleme yöntemi[22] kullanılmıştır.

### 3.3.3 Sistemde kullanılacak Olan Ekranlar

#### 3.3.3.1 Tercih ve Yönetim Ekranı

**Şekil 12 . Yarı Otomatik Ders Programı Sistemi Ana Ekran**

Yukarıdaki ekran ana ekran olarak kullanılacaktır. Kullanıcı adı ve şifresi ile sisteme girecek olan kullanıcı yapacağı işlemi seçip onunla ilgili butona basacaktır. Her kullanıcı için bütün butonlar açık değildir. Yani yönetici olarak sisteme girecek olan kullanıcı yeni kullanıcı da ekleyebilecektir ve aşağıdaki bütün butonlardaki işlemi yapacaktır.

Veri Giriş butonuna basıldığında derslerin, öğretim görevlilerinin, dersliklerin girişlerinin yapılacağı ekran açılacaktır.

Bölüm başkanı ekranına bastığımızda bu dönem için yerleştirilecek derslerin ilk girişinin yapılacağı ekran açılacaktır.

Program görüntüleme ekranına basıldığında derslik, öğretim görevlisi veya sınıf seçimliklerini seçerek istenilen program görüntülenecektir.

Liste görüntüleme butonuna basıldığında öğretim görevlileri, derslikler ve dersler listesi görülecektir.

Sistem tatmin ekranı, sistemin ne kadar başarılı atama yapmakta olduğunu işletim zamanında görüntüleyecek olan ekrandır.

Öğretim elemanı zaman tercih ekranı, öğretim elemanının istemediği yerleri kapatacağı ekrandır. Bu ekranın ayrıca Internet ortamından girilebileceği muadili(eşdeğeri) bir ekran da vardır. Bu da öğretim elemanının okul dışında olduğu zamanlar veya evde bulunduğu zaman, zaman tercihini girmesini sağlayacaktır.

### 3.3.3.2 Veri Giriş Ekranı: Derslik – Yeni Öğretim Elemanı – Yeni Ders

#### Derslik - Yeni Öğretim Elemanı - Yeni Ders Giriş Ekranı

Ders Girişi

Ders Kodu  T  P  KR  KE  Ders Adı

Ön Koşul

**İçeriği**  
 Projeksiyon ve Tek Bilgisayar  Beyaz Tahta  Tüm Bilgisayar

**Kaydet** **Güncelle** **Sil**

\* Ders Kodu kısmı 6 karakter olacaktır ve ilk 3 karakteri harf diğer 3 karakteri rakam olacaktır.  
\* T kısmında ders ile ilgili Teorik, P kısmında dersle ilgili Pratik, KR kısmında Kredi, KE kısmında KE kredisi bulunacaktır.  
\* Ders Adı kısmında dersin tam adı olmalıdır.  
\* Dersin içeriği kısmında dersin istenileri olacaktır.Bunlardan teklî ya da çoklu seçim yapılabilmektedir.  
\* Ön Koşul kısmında ders alabilecek öğrenci için ön koşul olmalıdır.  
\* Daha sonra Dersi Kaydet butonuna basılmalıdır ve ders kaydedilmelidir

Derslik ve Lab Girişi

Derslik No  Kapasitesi  Sıra Tipi  Kolçaklı  Sıralı

**İçeriği**  
 Projeksiyon ve Tek Bilgisayar  Beyaz Tahta  Tüm Bilgisayar

**Kaydet** **Güncelle** **Sil**

\* Derslik No kısmı ilk karakteri harf diğer karakterleri de rakam olmalıdır.  
\* Derslik kapasinde yazan numara olmalıdır.  
\* Kapasitesi kısmında dersliğin kaç öğrenci kapasitesi olduğu belirtilmelidir.  
\* Sıra tipi kısmında teklî seçim yapılacaktır ve kolçaklı ya da sıralı şekilde seçimler yapılmaktadır.  
\* Dersliğin içeriği kısmında çoklu seçim yapılabilecektir.İstenilen içerik seçilebilir.  
\* Daha sonra Derslik Kaydet butonuna basılıp derslik veri tabanına kaydedilmelidir.

Öğretim Elemanı Girişi

Sicil No  Soyadı  Adı  Ünvanı  Kadrosu Hangi Bölümde

66666  asistan  Bilgisayar Mühendisliği

**Kaydet** **Güncelle** **Sil**

\* Öğretim elemanının sicil numarası 4 rakamlı olmalıdır.  
\* Daha sonra Soyadı, adı, ünvanı girilmelidir ve Kadrosu seçildikten sonra Öğretim Elemanı Kaydet butonuna basılıp Öğretim elemanı kaydedilmelidir.

Şekil 13 . Yarı Otomatik Ders Programı Sistemi Derslik - Yeni Öğretim Elemanı - Yeni Ders Giriş Ekranı

Yukarıda görünen ekran bütün veri girişler için kullanılmaktadır. Ders Girişi, Derslik ve Lab Girişi ile Öğretim Elemanı Girişi kısımlarından oluşmaktadır.

Derslik Girişi kısmında Ders Kodu kısmı 6 karakter olacaktır ve ilk 3 karakteri harf, diğer 3 karakteri rakam olacaktır. T kısmında ders ile ilgili Teorik, P kısmında dersle ilgili Pratik, KR kısmında ABD stili Kredi, KE kısmında Avrupa stili ECTS kredisi bulunacaktır. Ders Adı kısmında dersin tam adı olmalıdır. Dersin içeriği kısmında dersin gerekleri olacaktır. Bunlardan tekli ya da çoklu seçim yapılabilmektedir. Ön Koşul kısmında dersi alabilecek öğrenci için ön koşul olmalıdır. Daha sonra Dersi Kaydet butonuna basılmalıdır ve ders kaydedilmelidir.

Derslik ve Lab Girişi kısmında Derslik No kısmı ilk karakteri harf, diğer karakterleri de rakam olmalıdır. Derslik no, kısmında yazan numara olmalıdır. Kapasitesi kısmında dersliğin normal oturma düzeninde kaç öğrenci kapasitesi olduğu belirtilmelidir. Bu sayı sınav düzenindeki kapasite değildir. Sıra tipi kısmında tekli seçim yapılacaktır ve kolçaklı ya da sıralı şekilde seçimler yapılmaktadır. Dersliğin içeriği kısmında çoklu seçim yapılabilecektir. İstenilen içerik seçilebilir. Daha sonra Derslik Kaydet butonuna basılıp derslik veri tabanına kaydedilmelidir.

Öğretim Elemanı Girişi Kısmında da öğretim elemanının sicil numarası 4 rakamlı olmalıdır. Daha sonra Soyadı, adı, unvanı girilmelidir ve kadrosu seçildikten sonra Öğretim Elemanı Kaydet butonuna basılıp öğretim elemanı kaydedilmelidir.

### 3.3.3.3 Bölüm Başkanı Yarıyıl Yönetim Ekranı

**Bölüm Başkanı Yarıyıl Yönetim Ekranı**

Bu yıldı Dersterle İlgili Girişler

Bölüm:  Üçüncü Öğretim Elemanı:

Ders Kodu:  İkinci Öğretim Elemanı:

Şube No:  Birinci Öğretim Elemanı:

Kontenjan:  Parça:  Tercih Ed. Derslik:

Tercih Yeri:  Terim:

\* Ders Kodu alanı 6 karakter ve ilk 3 karakteri harf diğerleri rakam olacak şekilde; Şube No 2 karakter ve rakam olacaktır.  
\* Birinci, ikinci ve üçüncü öğretim elemanı yerlerinde gerekli seçimler yapılacaktır.  
\* Dersin kontenjanı belirtilip kaç parça olarak ders saatinin bölünmesi eklenecektir.  
\* Tercih yeri istenilen seçilecek, beklenen düzensiz öğrenci sayısı ve tercih edilen derslik gerekli olan kısma yazılacaktır.  
\* Bütün bilgiler girdikten sonra kaydet butonuna basılınca o dönem açılacak olan ders ile ilgili bilgiler veri tabanına kaydedilmiş olacaktır.

Şekil 14 . Yarı Otomatik Ders Programı Sistemi Bölüm Başkanı Yarıyıl Yönetim Ekranı

Yukarıda Bölüm Başkanı kısmı yönetim ekranı bulunmaktadır. Bu ekrandan o dönem açılacak derslerle ilgili girişler yapılacaktır. Ders Kodu alanı 6 karakter ve ilk 3 karakteri harf diğerleri rakam olacak şekilde; Şube No 2 karakter ve rakam; birinci, ikinci ve üçüncü öğretim elemanı yerlerinde gerekli seçimler yapılacaktır. Dersin kontenjanı belirtilip kaç parça olarak ders saatinin bölünmesi eklenecektir. Bununla beraber tercih yeri istenilen seçilecek ve beklenen düzensiz öğrenci

sayısı gerekli olan kısma yazılacaktır. Bütün bilgiler girildikten sonra Kaydet butonuna basılınca o dönem açılacak olan ders ile ilgili bilgiler veri tabanına kaydedilmiş olacaktır.

### 3.3.3.4 Ders Programı Görüntüleme Ekranı

The screenshot shows a software interface for displaying a course program. At the top, there is a search section with the following fields: 'Sicil No', 'Soyadı', 'Adı', 'Derslik', 'Sınıf', 'Şube', and 'Bölümü'. The 'Sınıf' field is set to '1' and 'Şube' is set to '1'. The 'Bölümü' dropdown menu is set to 'Bilgisayar Mühendisliği'. There are 'Ara' (Search) and 'Temizle' (Clear) buttons. Below the search fields, there are two asterisked notes: '\* İstenilen alan girildikten sonra ara butonuna basıldığında istenilen ders programı aşağıda görüntülenecektir.' and '\* Öğretim Elemanı için sicil, ad veya soyad; sınıf için bölüm ve sınıf; derslik için derslik alanı doldurulmalıdır.' There is also a radio button for 'Çoklu Yazdır' (Print Multiple) and a 'Yazdır' (Print) button.

	Pazartesi	Salı	Çarşamba	Perşembe	Cuma
09-10	ENG100- (001) B311	EDE100- () A305	MAT152- (001) C304	BIL106- (001) B311	FIZ104- (001) C401
10-11	ENG100- (001) B311	EDE100- () A305	MAT152- (001) C304	BIL106- (001) B311	FIZ104- (001) C401
11-12	ENG100- (001) B311	FIZ110- (001) A305	BIL102- (001) C304	FIZ110- (001) A305	BIL162- (001) C304
12-13		FIZ110- (001) A305	BIL102- (001) C304	FIZ110- (001) A305	BIL162- (001) C304
13-14					
14-15	FIZ104- (001) C401	MAT152- (001) C304	BIL106- (001) B311	BIL102- (001) C304	
15-16	FIZ104- (001) C401	MAT152- (001) C304	BIL106- (001) B311	BIL102- (001) C304	
16-17		BIL162- (001) C304			
17-18		BIL162- (001) C304			
18+					

Şekil 15 . Yarı Otomatik Ders Programı Sistemi Ders Programı Görüntüleme Ekranı

Yukarıdaki ekranda değişik seçeneklerle ders programı görüntülenecektir.

- Öğretim üyesine (sicil no, adı, soyadı) göre,
- Bir bölüme göre(örneğin Bilgisayar Mühendisliği Bölümü),
- Bir bölümün bir sınıfına göre(örneğin Bilgisayar Mühendisliği Bölümü, 1. sınıf),
- Bir bölümün bir sınıfı içinde şubeye göre(örneğin Bilgisayar Mühendisliği Bölümü, 1. sınıf 1. şube),

- Derslięe gre

İstenilen kriterler girildikten sonra Ara butonuna basılacaktır. Ara butonuna basıldıktan sonra ders programı aŐağıdaki tabloda grünecektir.

Temizle butonunun iŐlevi ekranı ve aŐağıdaki tabloyu temizleyerek yeni aramaya hazırlamaktır.

Yazdır butonunun iki iŐlevi vardır. İlk iŐlevi aŐağıdaki tabloda grnen ders programını pdf dokmanına yazmaktır. Dięer iŐlevi oklu yazdır butonu seęildikten sonra Yazdır'a basılırsa sınıfların ve Őubelerin ders programını yazmaktır.



### 3.3.3.5 Listeleme Ekranı

Akademisyen Listesi     Ders Listesi     Sınıf Listesi

Bölüm

Bilgisayar Mühendisliği    Listele

\* Akademisyen Listesi seçildiğinde alttaki Bölüm seçim kısmından gerekli olan bölüm seçilip Listele butonuna basıldığında istenilen bölümle ilgili akademisyenlerin listesi gelecektir.

\* Ders Listesi kısmi isaretlendiğinde alttaki seçimden bölüm seçilir ve Listele butonuna basıldığında istenilen bölüm ile ilgili derslerin listesi gelmektedir.

\* Sınıf listesi isaretlendiğinde gerekli olan bölüm, Ders Kodu ve Subesi alanlar doldurulduktan sonra Listele butonuna basıldığında istenilen sınıf ile ilgili öğrencilerin listesi gelecektir.

Adı	Soyadı	Ünvanı	Bölümü
HAKAN	ÇAĞLAR	doç. dr.	Bilgisayar Mühendisliği
HASMET	GÜRÇAY	doç. dr.	Bilgisayar Mühendisliği
NIZAMI	GASSILOV	doç. dr.	Bilgisayar Mühendisliği
SERDAR	MÜLDÜR	doç. dr.	Bilgisayar Mühendisliği
SEREF	SAGIROĞLU	doç. dr.	Bilgisayar Mühendisliği
BÜLENT GÜRSEL	EMIROĞLU	ögr. gör.	Bilgisayar Mühendisliği
GÜVEN	KÖSE	ögr. gör.	Bilgisayar Mühendisliği
EMRE	SÜMER	ögr. gör.	Bilgisayar Mühendisliği
HALUK	NALBANTOĞLU	ögr. gör.	Bilgisayar Mühendisliği
İBRAHİM BARAN	USLU	ögr. gör.	Bilgisayar Mühendisliği
KRASSIMIR	IANKOV	ögr. gör.	Bilgisayar Mühendisliği
MUHAMMET	YORULMAZ	ögr. gör.	Bilgisayar Mühendisliği
NESLİŞAH	TOROSDAGLI	ögr. gör.	Bilgisayar Mühendisliği
OGUL	GÖÇMEN	ögr. gör.	Bilgisayar Mühendisliği
MEHMET ÜMIT	KARAKAS	prof. dr.	Bilgisayar Mühendisliği
TOFIK	MAMEDOV	prof. dr.	Bilgisayar Mühendisliği
FATİH VEHBI	ÇELEBI	ydoc. dr.	Bilgisayar Mühendisliği
MUSTAFA	SERT	ydoc. dr.	Bilgisayar Mühendisliği
BAHAETTİN	CENGİZ	prof. dr.	Bilgisayar Mühendisliği
BELGİN	OZKUL	ögr. gör.	Bilgisayar Mühendisliği
KAYA	KILAN	ögr. gör.	Bilgisayar Mühendisliği
MÜJDAT	KAYA	ögr. gör.	Bilgisayar Mühendisliği

Şekil 16 . Yarı Otomatik Ders Programı Sistemi Listeleme Ekranı Akademisyen Listesi

Yukarıdaki ekran istenilen listelerin oluşturulmasını sağlayacak ekrandır. Akademisyen Listesi, Ders Listesi ve Sınıf Listesi şeklindeki listeleri bize alt tabloda farklı kolon adları ile birlikte sunacaktır. Örneğin;

**Listeleme Ekranı**

Akademisyen Listesi
  Ders Listesi
  Sınıf Listesi

Bölüm

Bilgisayar Mühendisliği

Listele

\* Akademisyen Listesi seçildiğinde alttaki Bölüm seçim kısmından gerekli olan bölüm seçilip Listele butonuna basıldığında istenilen bölümle ilgili akademisyenlerin listesi gelecektir.

\* Ders Listesi kısmi işaretlendiğinde alttaki seçimden bölüm seçilir ve Listele butonuna basıldığında istenilen bölüm ile ilgili derslerin listesi gelmektedir.

\* Sınıf listesi işaretlendiğinde gerekli olan bölüm, Ders Kodu ve Subesi alanlar doldurulduktan sonra Listele butonuna basıldığında istenilen sınıf ile ilgili öğrencilerin listesi gelecektir.

Ders Kodu	Ders Adı	Kredi	Öğretim Üyesi
BIL102	BILGISAYAR YAZILIMI II	3	HASAN OGUL
BIL102	BILGISAYAR YAZILIMI II	3	OGUL GÖÇMEN
BIL106	PROGRAMLAMA LABORA...	1	KAYA KILAN
BIL106	PROGRAMLAMA LABORA...	1	OGUL GÖÇMEN
BIL162	BETIK DILLER	2	KAYA KILAN
BIL162	BETIK DILLER	2	EMRE SÜMER
BIL232	ELEKTRIK DEVRELERI	4	FATİH VEHBİ ÇELEBİ
BIL232	ELEKTRIK DEVRELERI	4	İBRAHİM BARAN USLU
BIL240	PROGRAMLAMA DILLERI	3	NESLİŞAH TOROSDAGLI
BIL240	PROGRAMLAMA DILLERI	3	NESLİŞAH TOROSDAGLI
BIL252	VERİ YÖNETİMİ VE DOSY...	3	MUSTAFA SERT
BIL252	VERİ YÖNETİMİ VE DOSY...	3	MUSTAFA SERT
BIL256	VERİ İLETİŞİMİ	3	KRASSİMİR İANKOV
BIL256	VERİ İLETİŞİMİ	3	KRASSİMİR İANKOV
BIL284	ÇEVİRİCİ DİLLER	3	EMRE SÜMER
BIL284	ÇEVİRİCİ DİLLER	3	EMRE SÜMER
BIL334	İŞLETİM SİSTEMLERİ	3	GÜRSEL EMİROĞLU
BIL352	VERİTABANI YÖNETİM SI...	3	MUSTAFA SERT
BIL364	OTOMATA TEORİSİ	3	NİZAMİ GASSİLOV
BIL375	BILGISAYAR ORGANİZA...	3	HALUK NALBANTOĞLU
BIL375	BILGISAYAR ORGANİZA...	3	KRASSİMİR İANKOV
BIL382	YAZILIM MÜHENDİSLİĞİN...	3	MEHMET ÜMİT KARAKAS

**Şekil 17 : Yarı Otomatik Ders Programı Sistemi Listeleme Ekranı Ders Listesi**

Akademisyen Listesi seçildiğinde alttaki Bölüm Seçim kısmından gerekli olan bölüm seçilip Listele butonuna basıldığında istenilen bölümle ilgili akademisyenlerin listesi gelecektir.

Aynı şekilde Ders Listesi kısmi işaretlendiğinde alttaki seçimden bölüm seçilir ve Listele butonuna basıldığında istenilen bölüm ile ilgili derslerin listesi gelir.

Son olarak Sınıf Listesi işaretlendiğinde gerekli olan bölüm, Ders Kodu ve Şubesi alanlarının doldurulması gerekmektedir. O alanlar doldurulduktan sonra da Listele butonuna basıldığında istenilen sınıf ile ilgili öğrencilerin listesi gelecektir.

### 3.3.3.6 Akıllı Ajanlar Tatmin Düzeyi İzleme Ekranı

**Akıllı Ajanlar Tatmin Düzeyi İzleme Ekranı**

Atama No: 172

Başlat Devam Et Durdur

Öğrenciler: 0.6428

Öğretim Üyeleri: 1.0

Dekan: 0.9621 Öğrenci Dekanı: 0.9789

Bölüm Başkanı: 1.0

Ortam Hizmet Sorumlusu: 0.9621

Alternatifleri Göster

Toplam: 0.8117

0.528 || 48 --> f(x) = 0.528 ||  
END202 --> B307 derslikli : 11 --> f(x) = 0.581 || 21 --> f(x) = 0.581 || 41 --> f(x) = 0.581 || 12 --> f(x) = 0.572 || 22 --> f(x) = 0.572 || 42 --> f(x) = 0.572 || 13 --> f(x) = 0.563 || 23 --> f(x) = 0.563 || 43 --> f(x) = 0.563 ||  
END202 --> B307 derslikli : 11 --> f(x) = 0.581 || 21 --> f(x) = 0.581 || 41 --> f(x) = 0.581 || 12 --> f(x) = 0.572 || 22 --> f(x) = 0.572 || 42 --> f(x) = 0.572 || 13 --> f(x) = 0.563 || 23 --> f(x) = 0.563 || 43 --> f(x) = 0.563 ||  
END202 --> B307 derslikli : 21 --> f(x) = 0.612 || 41 --> f(x) = 0.612 || 22 --> f(x) = 0.600 || 42 --> f(x) = 0.600 || 23 --> f(x) = 0.587 || 43 --> f(x) = 0.587 ||  
END202 --> B307 derslikli : 21 --> f(x) = 0.612 || 41 --> f(x) = 0.612 || 22 --> f(x) = 0.600 || 42 --> f(x) = 0.600 || 23 --> f(x) = 0.587 || 43 --> f(x) = 0.587 ||  
BIL375 --> B308 derslikli : 37 --> f(x) = 0.575 || 47 --> f(x) = 0.575 ||  
BIL375 --> B308 derslikli : 37 --> f(x) = 0.575 || 47 --> f(x) = 0.575 ||  
BIL375 --> B308 derslikli : 47 --> f(x) = 0.542 || 8 --> f(x) = 0.528 || 18 --> f(x) = 0.528 || 28 --> f(x) = 0.528

\* İlk olarak programın oluşturmaya başlaması gerekmektedir yani Başlat butonuna basılmalıdır.  
\* Ön analizden çıkan sonuçlara göre algoritmanın devam edip etmeyeceğine yoDep karar verilecektir.  
\* Tatmin düzeyleri istenilen düzeyde ise Devam Et butonuna basılacaktır.  
\* Eğer ki Tatmin Düzeyleri beğenilmezse algoritmanın çalışması durdurmak için Durdur butonuna basılmalıdır.

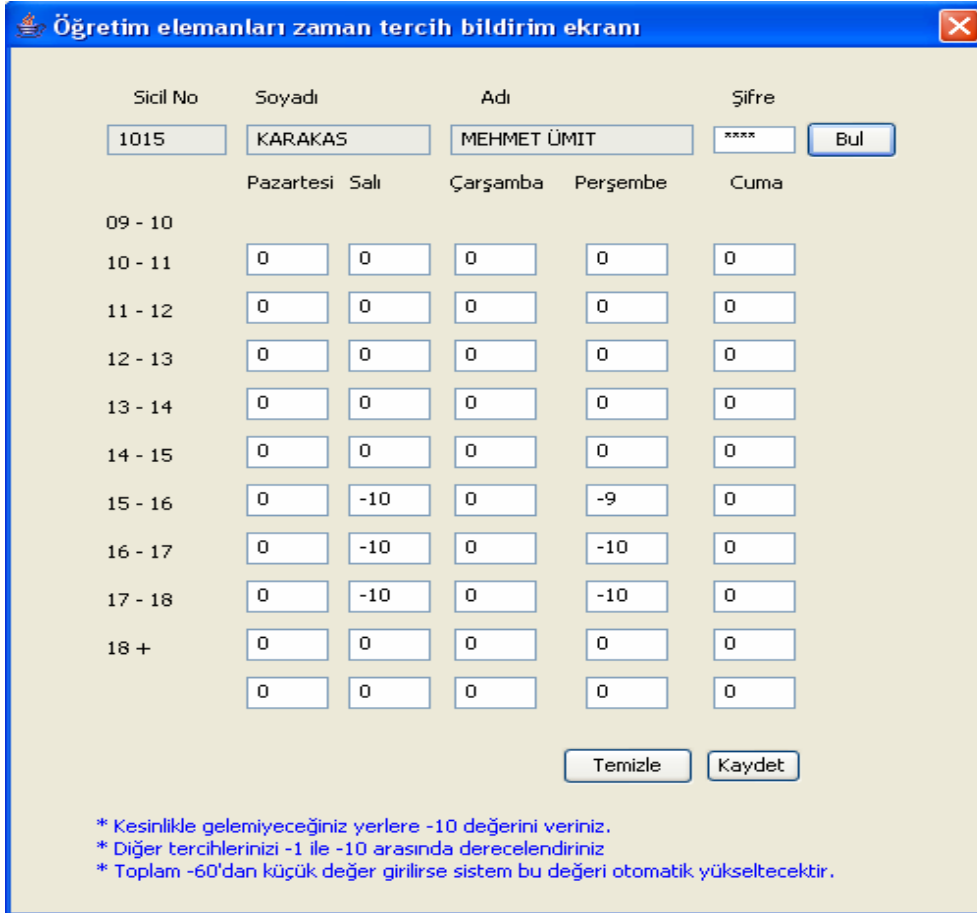
Şekil 18 . Yarı Otomatik Ders Programı Sistemi Akıllı Ajanlar Tatmin Düzeyi İzleme Ekranı

Yukarıdaki ekran, sistemin program oluşturmaya başladığında, oluşturulacak olan tatmin düzeylerinin atama numaraları ile beraber verilmesidir. İlk olarak programın oluşturmaya başlaması gerekmektedir. Sistem, anlık Öğrenciler,

Öğretim Üyeleri, Dekan, Bölüm Başkanları ve Ortam Hizmet Sorumlusu'nun tatmin düzeyleri tek tek verecek ve toplam tatmin düzeyini de bu bulunduğu değerlerden hesaplayacaktır. Buradan çıkan sonuçlara göre algoritmanın devam edip etmeyeceğine karar verilecektir. Tatmin düzeyleri istenilen düzeyde ise Durdur butonuna basılmasına gerek görülmeyecektir. Eğer ki Tatmin Düzeyleri beğenilmezse algoritmanın çalışması durdurulacaktır ve veritabanında gerekli değişiklikler yapılarak Devam Et butonuna basılacaktır.

Bununla beraber alt taraftaki alanda amaç fonksiyon değeri ve alternatif derslik saat dilimi bulunmaktadır. Ders programı oluşturulduktan sonra istenilirse fonksiyon değerine bakılarak alternatif yerlere dersler konulabilir.

### 3.3.3.7 Öğretim Elemanları Zaman Tercih Bildirim Ekranı (İnternetsiz kullanım)



The screenshot shows a window titled "Öğretim elemanları zaman tercih bildirim ekranı" with a close button. The form contains the following fields and options:

Sicil No	Soyadı	Adı	Şifre		
<input type="text" value="1015"/>	<input type="text" value="KARAKAS"/>	<input type="text" value="MEHMET ÜMIT"/>	<input type="text" value="****"/>	<input type="button" value="Bul"/>	
	Pazartesi	Salı	Çarşamba	Perşembe	Cuma
09 - 10					
10 - 11	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>
11 - 12	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>
12 - 13	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>
13 - 14	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>
14 - 15	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>
15 - 16	<input type="text" value="0"/>	<input type="text" value="-10"/>	<input type="text" value="0"/>	<input type="text" value="-9"/>	<input type="text" value="0"/>
16 - 17	<input type="text" value="0"/>	<input type="text" value="-10"/>	<input type="text" value="0"/>	<input type="text" value="-10"/>	<input type="text" value="0"/>
17 - 18	<input type="text" value="0"/>	<input type="text" value="-10"/>	<input type="text" value="0"/>	<input type="text" value="-10"/>	<input type="text" value="0"/>
18 +	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>
	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>

At the bottom of the form, there are two buttons: "Temizle" and "Kaydet". Below the buttons, there are three asterisks indicating instructions:

- \* Kesinlikle gelemediğiniz yerlere -10 değerini veriniz.
- \* Diğer tercihlerinizi -1 ile -10 arasında derecelendiriniz
- \* Toplam -60'dan küçük değer girilirse sistem bu değeri otomatik yükseltecektir.

Şekil 19 . Yarı Otomatik Ders Programı Sistemi Öğretim Elemanı Zaman Tercih Bildirim Ekranı

Yukarıdaki ekrandan öğretim üyeleri sıkışık olduğu saatleri belirteceklerdir. Bu ekran sadece okul içerisinde kullanılabilir. Öğretim üyesi sicil numarasına ya da adına soyadına göre Bul butonunu tetikleyecektir. Kesinlikle gelemeyeceği durumlarda -10 değeri verilecektir. -10 dan 0 a doğru istediği değeri vererek istediği yerleri belirtebilirler. Eğer Bölüm Başkanı ilgili kişinin limitini arttırmaz ise ve -60 dan fazla değer verilir ise sistem otomatik olarak bu değeri -60'a oranlayacaktır.

### 3.3.3.8 Öğretim Elemanları Zaman Tercih Bildirim Ekranı (İnternetli kullanım)

Sicil No	Soyadı	Adı	Şifre				
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="button" value="Bul"/>			
	Pazartesi	Salı	Çarşamba	Perşembe	Cuma		
09 - 10	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>		
10 - 11	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>		
11 - 12	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>		
12 - 13	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>		
13 - 14	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>		
14 - 15	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>		
15 - 16	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>		
16 - 17	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>		
17 - 18	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>		
18 +	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>		
	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>		

\* Kesinlikle gelemeyeceğiniz yerlere -10 değerini veriniz.  
\* Diğer tercihlerinizi -1 ile -10 arasında derecelendiriniz  
\* Toplam -60'dan küçük değer girilirse sistem bu değeri otomatik yükseltecektir.

Şekil 20 . Yarı Otomatik Ders Programı Sistemi Öğretim Elemanı Zaman Tercih Bildirim Ekranı (İnternet'li kullanım)

Yukarıdaki ekrandan öğretim üyeleri sıkışık olduğu saatleri belirteceklerdir. Bu ekran Internet olduğu her yerde kullanılabilir. Öğretim üyesi sicil numarasına ya da adına soyadına göre Bul butonunu tetikleyecektir. Kesinlikle

gelemeyeceği durumlarda -10 değeri verilecektir. -10 dan 0 a doğru istediği değeri vererek istediği yerleri belirtebilirler. -60 dan fazla değer verildiğinde sistem otomatik olarak bu değeri -60'a oranlayacaktır.

Bu ekranı dışardan kullanabilmek için ilk olarak Oracle Uygulama Sunucusu olan bir makine kurulmuştur. Bu makineye statik ip alınarak makinenin kapanması durumunda ipsinin değişmemesi sağlanmıştır. Uygulama, sunucuya kurulduktan sonra dışarıdan erişimi sağlanmıştır. Burada Java Applet kullanılmıştır. Çünkü internetsiz kullanım ile benzer ekran görüntüsü olması istenmektedir. Java Applet'in internetten kullanımını sağlamak için applet Gökhan MEMİŞ Tez adına imzalanmıştır. Daha sonra dışarıdan kullanımı test edilmiştir. Test edilirken kullanacak olan öğretim üyeleri adına işlem yapabilmek için şifre verilmiştir.

### 3.3.3.9 Ön Analiz Ekranı

Ön Analiz Ekranı

Atanacak Miktar: 561.0 Göster

Mevcut Miktar: 643.0 Genel Oran : 0.8724727838258165

Derslik Türü	Toplam	Istenen	Bu türde Oran	Dikkat
NORMAL SINIF KAPASITE 25	840	368	0.438	
KIMYA LABARATUARI	35	28	0.800	X
MAKINE GERECLERİ OLAN SINIF	70	17	0.242	
NORMAL SINIF KAPASITE 75 E KADAR	35	28	0.800	X
BIYOMEDİKAL GERECLERİ OLAN SINIF	70	8	0.114	
COK BILGISAYARLI SINIF	70	42	0.600	
ELEKTRİK GERECLERİ OLAN SINIF	70	37	0.528	
ENDUSTRI MUH GERECLERİ OLAN SINIF	35	16	0.457	
NORMAL SINIF KAPASITE 50'YE KADAR	35	13	0.371	

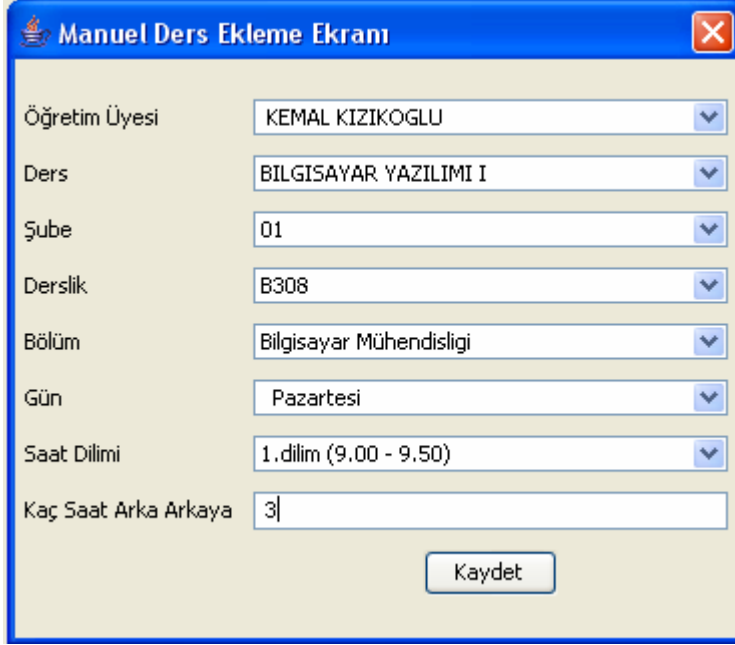
Durdur Devam

Şekil 21 . Yarı Otomatik Ders Programı Sistemi Ön Analiz Ekranı

Bu ekran ders programı oluşturma işlemleri başlamadan önce sistemin ne kadar dolu olduğunu kontrol etmek için tasarlanmıştır. Ön analiz sırasında sistem, imkânsız ya da ders programını oluşturması çok zor olduğunu önkestirirse işletim

iptal edilebilir ya da gerekli deęişiklikler yapılabilir. Bu ekranda dersliklerin atamaya başlamadan önce ne kadar sıkışık olduęu görölmektedir.

### 3.3.3.10 Manüel Ders Ekleme Ekranı



Öğretim Üyesi	KEMAL KIZIKOGLU
Ders	BILGISAYAR YAZILIMI I
Şube	01
Derslik	B308
Bölüm	Bilgisayar Mühendisligi
Gün	Pazartesi
Saat Dilimi	1. dilim (9.00 - 9.50)
Kaç Saat Arka Arkaya	3

Kaydet

Şekil 22 : Yarı Otomatik Ders Programı Sistemi Manüel Ders Ekleme Ekranı

Bu ekran ders oluşturma başlamadan önce manüel olarak atanacak derslerin girilmesini sağlamaktadır. Dönem başlarında Tarih, Edebiyat, İngilizce gibi derslerin atanması gerekmektedir. Çünkü bu tip derslerin yerleri sabittir ve o yerlere atanması gerekmektedir. Bu tip derslerin atanması da bu ekrandan yapılacaktır.

### 3.3.3.11 Yeni Kullanıcı Ekleme Ekranı



The screenshot shows a window titled "Yeni Kullanıcı Ekleme Ekranı" with a close button in the top right corner. The window contains the following fields:

- Kullanıcı Adı: gmemis
- Şifre: \*\*\*\*
- Şifre Tekrar: \*\*\*\*
- Yetkisi: Yonetici (dropdown menu)

A "Kaydet" button is located at the bottom center of the form.

Şekil 23 . Yarı Otomatik Ders Programı Sistemi Yeni Kullanıcı Ekleme Ekranı

Bu ekrandan sistemi kullanacak olan kullanıcıların girişi yapılacaktır. Kullanıcı adı, şifre bilgileri ve yetki bilgileri buradan girilebilir. Bu ekranı sadece yönetici kullanıcısı ile sisteme girmiş olan kullanıcılar kullanabileceklerdir. Yönetici, bir başka yöneticiye ve kullanıcıya yetki verebilir. Yönetici olmayan normal kullanıcılar yeni bir kullanıcı yaratamazlar.



### 3.3.3.12 Oluşturulmuş Olan Ders Programının PDF görüntüsü

	Pazartesi	Salı	Çarşamba	Perşembe	Cuma
1. Dilim	ENG100- (001) B311	EDE100- 0 A305	MAT152- (001) C304	BIL106- (001) B311	FIZ104- (001) C401
2. Dilim	ENG100- (001) B311	EDE100- 0 A305	MAT152- (001) C304	BIL106- (001) B311	FIZ104- (001) C401
3. Dilim	ENG100- (001) B311	FIZ110- (001) A305	BIL102- (001) C304	FIZ110- (001) A305	BIL162- (001) C304
4. Dilim		FIZ110- (001) A305	BIL102- (001) C304	FIZ110- (001) A305	BIL162- (001) C304
5. Dilim					
6. Dilim	FIZ104- (001) C401	MAT152- (001) C304	BIL106- (001) B311	BIL102- (001) C304	
7. Dilim	FIZ104- (001) C401	MAT152- (001) C304	BIL106- (001) B311	BIL102- (001) C304	
8. Dilim		BIL162- (001) C304			
9. Dilim		BIL162- (001) C304			

Şekil 24 . Yarı Otomatik Ders Programı Sistemi Oluşmuş Olan Programın PDF Görüntüsü

Burada örneklenen pdf dosyası, yoDep v1. programı görüntüleme ekranında yazdır butonuna basıldıktan sonra otomatik olarak oluşmakta, dökülmekte ve programın çalıştığı bilgisayarın C klasörü altına kaydedilmektedir.

### 3.3.4 yoDep v1.0 Kolay Belgele Tablosu

Kolay-BELGELE Küçük ve Orta boy yazılımlarda , tipik bilgi işlem merkezleri ve küçük yazılım firmalarının rahatça kullanabileceği Ümit KARAKAŞ tarafından geliştirilmiş belgeleme yöntemidir.[22]. Aşağıda, yoDep V1 yazılımında kullanılan bazı JAVA sınıflarının kolay belgele tabloları bulunmaktadır.

**Tablo 3. TatminManager sınıfı için kolay belgele tablosu**

<b>Proje Adı :</b>		Yarı Otomatik Ders Programı Sistemi				<b>Programlama Dili :</b>	JAVA
<b>Bu kestirim sırasında durumu :</b>		Programlama Başladı				<b>Tarih :</b>	03.02.2008
<b>Sınıf Adı :</b>		TatminManager					
Yordam No	Altsistem ya da Yordam adı	Güçlük Kodu (1)	Kim Yazdı / Yazacak	Belgeleme Komut Sayısı (2)	Önerilen Belgeleme Yüzdesi / Oluşan Satır (3)	Emek Normalize Değeri (4)	Tahmini Süre / Oluşan Süre ( gün ya da saat) (5)
1	calculateSatisfaction	3	Gökhan	33	4	101,8	1 saat
2	calculateFacultyDeanSatisfaction	2	Gökhan	2	0	4	0.1 saat
3	calculateStudentDeanSatisfaction	2	Gökhan	2	0	4	0.1 saat
4	calculateFacultyManagerSatisfaction	2	Gökhan	2	0	4	0.1 saat
5	calculateDepartmentPresidentSatisfaction	2	Gökhan	2	0	4	0.1 saat
6	calculateInstructorSatisfaction	2	Gökhan	2	0	4	0.1 saat
7	calculateStudentSatisfaction	2	Gökhan	3	0	6	0.1 saat
8	calculateTotalLecture	2	Gökhan	1	0	2	0.1 saat
	Toplam			47			1.7 saat

**Tablo 4. CreateProgram sınıfı için kolay belgele tablosu**

<b>Proje Adı :</b>		Yarı Otomatik Ders Programı Sistemi				<b>Programlama Dili :</b>	JAVA
<b>Bu kestirim sırasında durumu :</b>		Programlama Başladı				<b>Tarih :</b>	03.02.2008
<b>Sınıf Adı :</b>		CreateProgram					
Yordam No	Altsistem ya da Yordam adı	Güçlük Kodu (1)	Kim Yazdı / Yazacak	Belgeleme Komut Sayısı (2)	Önerilen Belgeleme Yüzdesi / Oluşan Satır (3)	Emek Normalize Değeri (4)	Tahmini Süre / Oluşan Süre ( gün ya da saat) (5)
1	createProgram	3	Gökhan	96	10	295	5 saat
2	generateGunAndSaat	3	Gökhan	44	5	135,5	4 saat
3	selectGunSaat	2	Gökhan	34	5	71,5	4 saat
4	selectionSqlForDoubleOrTriple	2	Gökhan	10	1	20,7	1 saat
5	saveProgram	2	Gökhan	23	4	48,8	2 saat
6	deleteLectureBecauseOfInfiniteLoop	2	Gökhan	32	5	67,5	3 saat
	Toplam			239			19 saat

**Tablo 5. TezManager sınıfı için kolay belgele tablosu**

<b>Proje Adı :</b>		Yarı Otomatik Ders Programı Sistemi				<b>Programlar:</b> JAVA	
<b>Bu kestirim sırasında durumu :</b>		Programlama Başladı				<b>Tarih :</b> 03.02.2008	
<b>Sınıf Adı :</b>		TezManager					
Yordam No	Altsistem ya da Yordam adı	Güçlük Kodu (1)	Kim Yazdı / Yazacak	Belgelemesiz Komut Sayısı (2)	Önerilen Belgeleme Yüzdesi / Oluşan Satır (3)	Emek Normalize Değeri (4)	Tahmini Süre / Oluşan Süre ( gün ya da saat) (5)
1	insertThisSemesterCourse	2	Gökhan	20	2	41,4	0,25 saat
2	getAllDepartment	2	Gökhan	13	2	27,4	0,1 saat
3	getAllInstructor	2	Gökhan	14	2	29,4	0,1 saat
4	getAllClassroom	2	Gökhan	14	2	29,4	0,1 saat
5	saveInstructorSelection	3,5	Gökhan	24	3	86,1	0,1 saat
6	getInstructorSelectionList	2	Gökhan	5	1	10,7	0,1 saat
7	deleteInstructorSelection	2	Gökhan	17	2	35,4	0,1 saat
8	getAllthisSemesterLecture	2	Gökhan	20	2	41,4	0,1 saat
9	getLecture	2	Gökhan	1	0	2	0,1 saat
10	getFirstAnalysisValues	2	Gökhan	34	5	71,5	0,1 saat
11	getTotalCount	2	Gökhan	1	0	2	0,1 saat
12	getInstructorClosenessCount	2	Gökhan	1	0	2	0,1 saat
13	saveLecture	2	Gökhan	22	3	46,1	0,1 saat
14	getFirstAnalysisValuesForClass	2	Gökhan	35	5	73,5	0,25 saat
15	getInstructorProgram	2	Gökhan	47	5	97,5	0,25 saat
16	preReport	2	Gökhan	40	5	83,5	0,25 saat
17	getClassProgram	2	Gökhan	44	5	91,5	0,25 saat
18	getClassRoomProgram	2	Gökhan	37	5	77,5	0,25 saat
19	getInstructorWithDepartment	2	Gökhan	36	5	75,5	0,25 saat
20	getLectureWithDepartment	2	Gökhan	45	5	93,5	0,25 saat
21	getClassroomList	2	Gökhan	29	4	60,8	0,25 saat
22	getAllInstructorProgram	2	Gökhan	59	8	123,6	0,5 saat

**Tablo 6. HibernateManager sınıfı için kolay belgele tablosu**

<b>Proje Adı :</b>		Yarı Otomatik Ders Programı Sistemi				<b>Programlar:</b> JAVA	
<b>Bu kestirim sırasında durumu :</b>		Programlama Başladı				<b>Tarih :</b> 03.02.2008	
<b>Sınıf Adı :</b>		HibernateManager					
Yordam No	Altsistem ya da Yordam adı	Güçlük Kodu (1)	Kim Yazdı / Yazacak	Belgelemesiz Komut Sayısı (2)	Önerilen Belgeleme Yüzdesi / Oluşan Satır (3)	Emek Normalize Değeri (4)	Tahmini Süre / Oluşan Süre ( gün ya da saat) (5)
1	generateSessionFactory	2	Gökhan	6	1	12,7	0,25 saat
	Toplam			6			0,25 saat

**Tablo 7. SatisfactionFrame sınıfı için kolay belgele tablosu**

<b>Proje Adı :</b>	Yarı Otomatik Ders Programı Sistemi				<b>Programı:</b>	JAVA	
<b>Bu kestirim sırasında durumu :</b>	Programlama Başladı				<b>Tarih :</b>	03.02.2008	
<b>Sınıf Adı :</b>	SatisfactionFrame						
Yordam No	Altsistem ya da Yordam adı	Güçlük Kodu (1)	Kim Yazdı / Yazacak	Belgelem esiz Komut Sayısı (2)	Önerilen Belgeleme Yüzdesi / Oluşan Satır (3)	Emek Normalize Değeri (4)	Tahmini Süre / Oluşan Süre ( gün ya da saat) (5)
1	jblnit	2	Gökhan	177	10	361	0,5 saat
2	buttonStart_actionPerformed	2	Gökhan	3	1	6,7	0,25 saat
	Toplam			180			0,75 saat

**Tablo 8. ProgramViewerFrame sınıfı için kolay belgele tablosu**

<b>Proje Adı :</b>	Yarı Otomatik Ders Programı Sistemi				<b>Programı:</b>	JAVA	
<b>Bu kestirim sırasında durumu :</b>	Programlama Başladı				<b>Tarih :</b>	03.02.2008	
<b>Sınıf Adı :</b>	ProgramViewerFrame						
Yordam No	Altsistem ya da Yordam adı	Güçlük Kodu (1)	Kim Yazdı / Yazacak	Belgelem esiz Komut Sayısı (2)	Önerilen Belgeleme Yüzdesi / Oluşan Satır (3)	Emek Normalize Değeri (4)	Tahmini Süre / Oluşan Süre ( gün ya da saat) (5)
1	jblnit	3,5	Gökhan	124	10	441	0,5 saat
2	buttonFind_actionPerformed	3	Gökhan	57	5	174,5	0,25 saat
3	buttonClear_actionPerformed	1	Gökhan	1	0	1	0,01 saat
4	clearScreen	2	Gökhan	13	1	26,7	0,01 saat
5	buttonPrint_actionPerformed	3	Gökhan	30	2	91,4	0,25 saat
	Toplam			225			1,2 saat

**Tablo 9. MainFrame sınıfı için kolay belgele tablosu**

Proje Adı :		Yarı Otomatik Ders Programı Sistemi				Programı:	JAVA
Bu kestirim sırasında durumu :		Programlama Başladı				Tarih :	03.02.2008
Sınıf Adı :		MainFrame					
Yordam No	Altsistem ya da Yordam adı	Güçlük Kodu (1)	Kim Yazdı / Yazacak	Belgelem esiz Komut Sayısı (2)	Önerilen Belgeleme Yüzdesi / Oluşan Satır (3)	Emek Normalize Değeri (4)	Tahmini Süre / Oluşan Süre ( gün ya da saat) (5)
1	jblnit	3,5	Gökhan	93	3	327,6	0,5 saat
2	buttonPresident_actionPerformed	2	Gökhan	3	1	6,7	0,1 saat
3	buttonDataEntry_actionPerformed	2	Gökhan	3	2	7,4	0,1 saat
4	buttonProgramViewer_actionPerformed	2	Gökhan	3	3	8,1	0,1 saat
5	buttonList_actionPerformed	2	Gökhan	3	1	6,7	0,1 saat
6	buttonSatisfaction_actionPerformed	2	Gökhan	3	1	6,7	0,1 saat
7	buttonInstructorSelection_actionPerformed	2	Gökhan	3	1	6,7	0,1 saat
8	buttonPreAnalysis_actionPerformed	2	Gökhan	3	1	6,7	0,1 saat
	Toplam			114			1,2 saat

**Tablo 10. InstructorSelectionFrame sınıfı için kolay belgele tablosu**

Proje Adı :		Yarı Otomatik Ders Programı Sistemi				Programı:	JAVA
Bu kestirim sırasında durumu :		Programlama Başladı				Tarih :	03.02.2008
Sınıf Adı :		InstructorSelectionFrame					
Yordam No	Altsistem ya da Yordam adı	Güçlük Kodu (1)	Kim Yazdı / Yazacak	Belgelem esiz Komut Sayısı (2)	Önerilen Belgeleme Yüzdesi / Oluşan Satır (3)	Emek Normalize Değeri (4)	Tahmini Süre / Oluşan Süre ( gün ya da saat) (5)
1	jblnit	3,5	Gökhan	100	3	352,1	0,5 saat
2	buttonSave_actionPerformed	3	Gökhan	34	1	102,7	0,1 saat
3	clearScreen	2	Gökhan	12	2	25,4	0,1 saat
4	buttonFind_actionPerformed	3	Gökhan	27	3	83,1	0,1 saat
5	textFieldRegistryNo_actionPerformed	3,5	Gökhan	1	1	4,2	0,05 saat
	Toplam			174			0,85 saat

**Tablo 11. FirstAnalyses sınıfı için kolay belgele tablosu**

Proje Adı :		Yarı Otomatik Ders Programı Sistemi				Programı:	JAVA
Bu kestirim sırasında durumu :		Programlama Başladı				Tarih :	03.02.2008
Sınıf Adı :		FirstAnalyses					
Yordam No	Altsistem ya da Yordam adı	Güçlük Kodu (1)	Kim Yazdı / Yazacak	Belgelem esiz Komut Sayısı (2)	Önerilen Belgeleme Yüzdesi / Oluşan Satır (3)	Emek Normalize Değeri (4)	Tahmini Süre / Oluşan Süre ( gün ya da saat) (5)
1	jblnit	2	Gökhan	28	3	58,1	0,1 saat
2	buttonView_actionPerformed	3	Gökhan	8	1	24,7	0,05 saat
	Toplam			36			0,15 saat

**Tablo 12. DepartmentPresidentFrame sınıfı için kolay belgele tablosu**

<b>Proje Adı :</b>	Yarı Otomatik Ders				<b>Programl:</b>	JAVA	
<b>Bu kestirim sırasında durumu :</b>	Programlama Başla				<b>Tarih :</b>	03.02.2008	
<b>Sınıf Adı :</b>	DepartmentPresidentFrame						
Yordam No	Altsistem ya da Yordam adı	Güçlük Kodu (1)	Kim Yazdı / Yazacak	Belgelem esiz Komut Sayısı (2)	Önerilen Belgeleme Yüzdesi / Oluşan Satır (3)	Emek Normalize Değeri (4)	Tahmini Süre / Oluşan Süre ( gün ya da saat) (5)
1	jbinit	3	Gökhan	117	10	358	0,5 saat
2	buttonSaveYearClass_actionPerformed	3	Gökhan	20	2	61,4	0,05 saat
	Toplam			137			0,55 saat

**Tablo 13. SystemUtility sınıfı için kolay belgele tablosu**

<b>Proje Adı :</b>	Yarı Otomatik Ders Programı Sistemi				<b>Programl:</b>	JAVA	
<b>Bu kestirim sırasında durumu :</b>	Programlama Başladı				<b>Tarih :</b>	03.02.2008	
<b>Sınıf Adı :</b>	SystemUtility						
Yordam No	Altsistem ya da Yordam adı	Güçlük Kodu (1)	Kim Yazdı / Yazacak	Belgelem esiz Komut Sayısı (2)	Önerilen Belgeleme Yüzdesi / Oluşan Satır (3)	Emek Normalize Değeri (4)	Tahmini Süre / Oluşan Süre ( gün ya da saat) (5)
1	showAtCenter	2	Gökhan	10	2	21,4	0,1 saat
	Toplam			10			0,1 saat

**Tablo 14. PDFConverter sınıfı için kolay belgele tablosu**

<b>Proje Adı :</b>		Yarı Otomatik Ders				<b>Programl:</b>	JAVA
<b>Bu kestirim sırasında durum</b>		Programlama Başla				<b>Tarih :</b>	03.02.2008
<b>Sınıf Adı :</b>		PDFConverter					
Yordam No	Altsistem ya da Yordam adı	Güçlük Kodu (1)	Kim Yazdı / Yazacak	Belgelem esiz Komut Sayısı (2)	Önerilen Belgeleme Yüzdese / Oluşan Satır (3)	Emek Normalize Değeri (4)	Tahmini Süre / Oluşan Süre ( gün ya da saat) (5)
1	createPDF	3	Gökhan	20	4	62,8	0,1 saat
2	createProgramPdf	3,5	Gökhan	125	10	444,5	2 saat
3	createMultipleProgramPdf	3,5	Gökhan	123	10	437,5	2 saat
4	countClass	2	Gökhan	12	2	25,4	0,1 saat
5	getCell	2	Gökhan	3	1	6,7	0,05 saat
6	setColspan	2	Gökhan	2	1	4,7	0,05 saat
7	setHorisantalAlignment	2	Gökhan	2	1	4,7	0,05 saat
8	setHeight	2	Gökhan	2	1	4,7	0,05 saat
	Toplam			289			4,4 saat

**Tablo 15. ClientHandler sınıfı için kolay belgele tablosu**

<b>Proje Adı :</b>		Yarı Otomatik Ders				<b>Programlama</b>	JAVA
<b>Bu kestirim sırasında durumu :</b>		Programlama Başladı				<b>Dili :</b>	JAVA
<b>Sınıf Adı :</b>		ClientHandler				<b>Tarih :</b>	03.02.2008
Yordam No	Altsistem ya da Yordam adı	Güçlük Kodu (1)	Kim Yazdı / Yazacak	Belgelem esiz Komut Sayısı (2)	Önerilen Belgeleme Yüzdese / Oluşan Satır (3)	Emek Normalize Değeri (4)	Tahmini Süre / Oluşan Süre ( gün ya da saat) (5)
1	insertThisSemesterCourse	1,3	Gökhan	1	1	2	0,05 saat
2	getAllDepartment	1,3	Gökhan	1	1	2	0,05 saat
3	getAllInstructor	1,3	Gökhan	1	1	2	0,05 saat
4	getAllClassroom	1,3	Gökhan	1	1	2	0,05 saat
5	saveInstructorSelection	1,3	Gökhan	1	1	2	0,05 saat
6	deleteInstructorSelection	1,3	Gökhan	1	1	2	0,05 saat
7	getAllthisSemesterLecture	1,3	Gökhan	1	1	2	0,05 saat
8	getFirstAnalysisValues	1,3	Gökhan	1	1	2	0,05 saat
9	getFirstAnalysisValuesForClass	1,3	Gökhan	1	1	2	0,05 saat
10	getInstructorProgram	1,3	Gökhan	1	1	2	0,05 saat
11	getClassProgram	1,3	Gökhan	1	1	2	0,05 saat
12	getClassRoomProgram	1,3	Gökhan	1	1	2	0,05 saat
13	getInstructorWithDepartment	1,3	Gökhan	1	1	2	0,05 saat
14	getLectureWithDepartment	1,3	Gökhan	1	1	2	0,05 saat
15	getClassroomList	1,3	Gökhan	1	1	2	0,05 saat
16	getAllInstructorProgram	1,3	Gökhan	1	1	2	0,05 saat
17	getInstructorSelectionList	1,3	Gökhan	1	1	2	0,05 saat
18	getInitialContext	2	Gökhan	6	2	13,4	0,25 saat
19	invokeMethod	2	Gökhan	15	4	32,8	0,25 saat
	Toplam			38			0,9 saat

Özetle, tez çalışması 2006 Kasım ayında başlamış olup, Kasım 2006 ile Ocak 2007 arasında tez için gerekli bilgiler toplanmış, 2007 Ocak ile 2007 Mayıs ayları arasında tezin tasarım kararları alınmıştır. Programlama çalışması genellikle 2007 Mayıs ile 2007 Temmuz ve 2007 Eylül ve 2008 Ocak arasında akşam evde çalışma ile yapılmıştır. İşin programlama kesiminde yaklaşık 7600 satır kod üretilmiş, yaklaşık 50 saat programcı zamanı kullanılmıştır.

Tez çalışmasının, kaynak araştırması, tez programının mimarı ve tasarımı, tez metninin defalarca yazılması işi de en az 320 saat almıştır.

### 3.3.5 Hibernate

[ab.org.tr/ab06/bildiri/46.pdf](http://ab.org.tr/ab06/bildiri/46.pdf)

Kalıcı verilerin yönetim şekli bütün yazılım projeleri için önemli bir anahtar nokta olmuştur. İlkel sorgulama işlemlerini(create, insert, update,delete) SQL ve JDBC de el ile mi yazmalıyız yoksa bunlar otomatik olarak mı oluşturulmalı sorusu her zaman bulunmaktadır. Normalde Java'da kalıcılık verinin ilişkisel veri tabanına SQL ile kaydedilmesidir. Ama burada sorun bu işi nesneye dayalı uygulamalarda en başarılı şekilde nasıl yapacağımızdır.

Bu sorunların cevapları tartışılırken, son zamanlarda Nesne İlişkisel Eşleme (ORM(Object/Relational Mapping)) büyük kabul gördü. Nesne İlişkisel Eşleme, İlişkisel Veri tabanı ile Nesnelere arasında çevirme yapma ve bağlantı kurma işlemidir. Küçük nesnelere bu işlemi yapmak kolay iken nesnelere, verilerin ve birleştirmemiz gereken tabloların sayısı arttıkça, bu işleminin yapılması oldukça zor bir hal alır. Karmaşıklaşan nesne modeli ve ilişkileri karşımıza sorun olarak çıkar. Hibernate bu nokta da bu işlemleri en başarılı şekilde yerine getirmemizde bize büyük kolaylık, rahatlık ve başarımlar sağlar. Hibernate Java'da kalıcı veri yönetimine bütün bir çözüm getiren bir projedir. İyi bir Nesne/İlişkisel Eşleme aracıdır. Uygulamaların, ilişkisel veri tabanı ile etkileşimine aracılık eder. Basit bir Java nesnesini kalıcı hale gelmesini ve kaydedilmiş kalıcı nesneyi geri



yüklememizi basit komutlarla sağlar. Bu da geliştiricinin sadece iş mantığına odaklanmasını sağlar. Takip edilmesini gerektiren belli katı kuralları yoktur. Bu sayede hem yeni hem de var olan projelere herhangi bir değişiklik gerektirmeden başarılı bir şekilde uyum sağlar. Java dünyasında ilişkisel veri tabanlarına ilk erişim yöntemi JDBC olmuştur. Fakat burada basit sorgulama komutlarını dahi elle yazmamız gerekiyordu. Bu da geliştiriciye büyük yük getiriyordu. Bu yüzden yazılan uygulama kodları da veri yapısına bağımlıydı. Yazdığımız nesneye dayalı kodlamaya rağmen veriye erişim nesnesel olmayan bir şekilde yapılıyordu. Nesneye Dayalı Programlama ile uğraşan geliştiriciler, veriye de yine nesnelere ulaşmak ister. Bütün bu ihtiyaçları en iyi yerine getiren Hibernate oldu. Kalıcı nesnelere, başka bir nesnedan türetilmeyen basit Java nesnelere POJO(Plain Old Java Object) olmalıydı. Belirli katı standartları olmamalı ve güçlü bir sorgulama dilini desteklemeliydi.

Hibernate HQL adında SQL'e benzeyen bir sorgulama dilini destekler. Basit sorgulama cümleleri elle yazılmak yerine Hibernate tarafından otomatik olarak üretilmektedir. Zorlu sorgulamalar nesnesel bir yapıya sahip olan HQL tarafından gerçekleştirilebilir. HQL'e yollanan parametre ve geri dönen değerler bundan böyle nesne yapısında olabilir.

Hibernate'in genel mimarisi aşağıdaki şekilde gösterilmiştir.



Şekil 25. Hibernate genel yapısı

Hibernate önceden açılan bağlantıları tekrar hızlı bir şekilde kullanmayı sağlayan “Bağlantı Havuzlarını” ve okunan verinin yazılan veriden fazla olduğu durumda büyük performans sağlayan “Önbellekleme” 'yi destekler. C3P0, Proxool, Apache DBCP gibi bağlantı havuzlarını destekler. Hibernate'te iki şekilde önbellekleme yapılır. Session üzerinde tutulan önbellek olan Birinci Seviye Önbellek ile asıl zor olan, değiştirilebilir önbellek olan İkinci Seviye Önbellek'tir.

### **3.3.6 Uygulama Sunucusu**

J2EE (Java 2 Enterprise Edition) teknolojilerini kullanarak geliştirilen uygulamaları, standartlara (J2EE belirtimine) uygun olarak çalıştıran yazılımlardır. Kullanıcı arayüzü ile veri tabanı (veya bu görevi gören sistemler) arasında yer alırlar. Çok katlı bir mimari düşünülürse, uygulama sunucusu, mimarinin orta katlarını içinde bulundurur. İşte bu yüzden, uygulama sunucuları, orta kat yazılımı (Middleware) sayılırlar ve orta kat yazılım teknolojilerini kullanırlar. Genellikle, kullanıcı arayüzüyle bilgi işlem servisleri arasında veya veri işlem katıyla veri tabanı arasında ya da dağıtık nesnelere ölçeklenebilirlik servisleri arasında bu teknolojilerden yararlanırlar.

#### **3.3.6.1 Oracle Uygulama Sunucusu**

Oracle Application Server (Uygulama Sunucusu)yazılımı, Oracle' ın Internet platformu için çok önemli bir bileşenidir. Bundan yıllar önce ilk veritabanlarının var olmasının nedeni yazılımları veriden bağımsız kılmak iken, şimdi eğilim verilerin arasındaki ilişkiyi veritabanından bağımsız kılma yönündedir. Oracle' ın bu amaçla geliştirilmiş yazılımı Application Server çok daha az kompleks modüller içeren sunucular ve daha iyi yazılımların ortaya çıkması amaçlamaktadır. (<http://www.umutgokbayrak.com/page0/page0.html>)

Oracle Application Server;

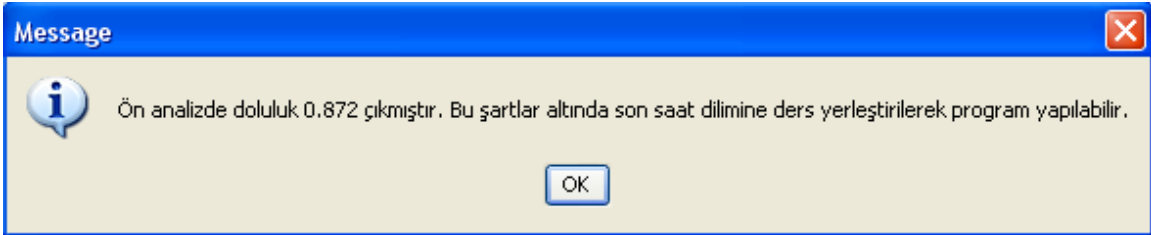
- Java ve daha pek çok programlama diline destek vermekte olup, bir yazılım geliştirirken çoğunlukla en çok vakit harcanan yerler olan ağ

işlemleri, entegrasyon, güvenilirlik, geliştirilebilirlik gibi ihtiyaçlara hazır çözümler sunmaktadır.

- Bütün ağ modellerine uygun yapısı ile HTML, Java, CORBA, COM gibi yapılara tam destek vermektedir. Diğer benzeri yazılımların aksine Oracle dağıtılabilir bir mimaridedir. Üzerinde çalışan süreçlerin (proseslerin) yükünü azaltmak için çok sayıda sunucuya görev dağılımı yapabilir ve sonuçları tek bir sunucu üzerinden yayınlabilir.
- Oracle veritabanları sadece Oracle veritabanları ile iletişim içinde olmak zorunda değildir. ODBC/JDBC arayüzleri sayesinde herhangi bir veritabanı ile rahatlıkla iletişime geçebilir.
- Bütün bir Oracle Application Server sitesi, tek bir makine veya yüzlerce makineden oluşabilir, web tarayıcı ile tek bir konsoldan yönetilebilir. Yakın gelecekte Oracle Application Server, Oracle Enterprise Manager ile entegre bir şekilde sunulacak olup, bütün bir Oracle platformunun tek bir konsoldan yönetimi mümkün kılınacaktır.

### 3.3.7 Ders Programı Oluşturma

Akıllı ajanlar tatmin düzeyi izleme ekranından ders programı oluşturmaktadır. Bu ekranda ilk olarak ön analizdeki doluluk bilgisi verilmektedir.



**Şekil 26 : Ön analiz uyarısı**

Bu ekranı gördükten sonra eğer ki doluluk oranı %90'dan büyük ise ders programı oluşturmaya sistem başlamamaktadır.

Doluluk oranı %90dan küçük ise ön analiz ekranı açılıp son kontrol olarak dersliklerin dolulukları görülmektedir.

**Ön Analiz Ekranı**

Atanacak Miktar: 561.0

Mevcut Miktar: 643.0 Genel Oran : 0.8724727838258165

Derslik Türü	Toplam	Istene	Bu türde Oran	Dikkat
NORMAL SINIF KAPASITE 25	840	368	0.438	
KIMYA LABARATUARI	35	28	0.800	X
MAKINE GERECLERİ OLAN SINIF	70	17	0.242	
NORMAL SINIF KAPASITE 75 E KADAR	35	28	0.800	X
BIYOMEDİKAL GERECLERİ OLAN SINIF	70	8	0.114	
COK BILGISAYARLI SINIF	70	42	0.600	
ELEKTRİK GERECLERİ OLAN SINIF	70	37	0.528	
ENDUSTRI MUH GERECLERİ OLAN SINIF	35	16	0.457	
NORMAL SINIF KAPASITE 50'YE KADAR	35	13	0.371	

**Şekil 27 : Program oluşturulmadan önceki ön analiz**

Bu ekranda doluluklar %80 ve daha büyük olduğunda yanına (x) işareti bulunmaktadır. Bu sayede istenilirse doluluk olan derslik türlerinden diğerlerine kaydırma yapılabilir. Bu sayede kalabalık ya da sistemin sıkıntıya düşebileceği yerler daha önceden çözülebilecektir.

Sistem oluşturmaya başladıktan sonra ise anlık tatminler, amaç fonksiyon değerleri ve alternatif derslik saat dilimi gösterilmektedir.

**Akıllı Ajanlar Tatmin Düzeyi İzleme Ekranı**

Atama No:

Öğrenciler:

Öğretim Üyeleri:

Dekan:  Öğrenci Dekanı:

Bölüm Başkanı:

Ortam Hizmet Sorumlusu:

Toplam:

0.528 || 48 --> f(x) = 0.528 ||  
 END202 --> B307 derslikli : 11 --> f(x) = 0.581 || 21 --> f(x) = 0.581 || 41 --> f(x) = 0.581 || 12 --> f(x) = 0.572 || 22 --> f(x) = 0.572 || 42 --> f(x) = 0.572 || 13 --> f(x) = 0.563 || 23 --> f(x) = 0.563 || 43 --> f(x) = 0.563 ||  
 END202 --> B307 derslikli : 11 --> f(x) = 0.581 || 21 --> f(x) = 0.581 || 41 --> f(x) = 0.581 || 12 --> f(x) = 0.572 || 22 --> f(x) = 0.572 || 42 --> f(x) = 0.572 || 13 --> f(x) = 0.563 || 23 --> f(x) = 0.563 || 43 --> f(x) = 0.563 ||  
 END202 --> B307 derslikli : 21 --> f(x) = 0.612 || 41 --> f(x) = 0.612 || 22 --> f(x) = 0.600 || 42 --> f(x) = 0.600 || 23 --> f(x) = 0.587 || 43 --> f(x) = 0.587 ||  
 END202 --> B307 derslikli : 21 --> f(x) = 0.612 || 41 --> f(x) = 0.612 || 22 --> f(x) = 0.600 || 42 --> f(x) = 0.600 || 23 --> f(x) = 0.587 || 43 --> f(x) = 0.587 ||  
 BIL375 --> B308 derslikli : 37 --> f(x) = 0.575 || 47 --> f(x) = 0.575 ||  
 BIL375 --> B308 derslikli : 37 --> f(x) = 0.575 || 47 --> f(x) = 0.575 ||  
 BIL375 --> B308 derslikli : 47 --> f(x) = 0.542 || 8 --> f(x) = 0.528 || 18 --> f(x) = 0.528 || 28 --> f(x) = 0.528

\* İlk olarak programın oluşturmaya başlaması gerekmektedir yani Başlat butonuna basılmalıdır.  
 \* Ön analizden çıkan sonuçlara göre algoritmanın devam edip etmeyeceğine yöDep karar verilecektir.  
 \* Tatmin düzeyleri istenilen düzeyde ise Devam Et butonuna basılacaktır.  
 \* Eğer ki Tatmin Düzeyleri beğenilmezse algoritmanın çalışması durdurmak için Durdur butonuna basılmalıdır.

**Şekil 28 : Program oluşturma**

### 3.3.8 Veri Tabanı Tabloları

#### 3.3.8.1 Atanacak Ders Durumu Tablosu

Bu tablo bu dönem için ataması gerçekleşecek olan derslerin verilerinin bulunduğu tablodur. Bu tablodaki kolonlar şu şekildedir:

#	Adı	Veri Tipi	Nulls?
1	ID	NUMBER	N
2	DERSID	VARCHAR2(11)	N

3	SUBENO	VARCHAR2(3)	N
4	TERCIHBOLGESI	VARCHAR2(3)	Y
5	BIRINCIOGRETIMUYESIID	NUMBER	N
6	IKINCIOGRETIMUYESIID	NUMBER	Y
7	UCUNCUOGRETIMUYESIID	NUMBER	Y
8	BOLUMID	NUMBER	N
9	KONTENJAN	VARCHAR2(3)	N
10	PARCA	VARCHAR2(3)	Y
11	DUZENSIZOGRSAYISI	VARCHAR2(3)	Y
12	DERSLIKID	NUMBER	Y

### 3.3.8.2 Bölüm Tablosu

Bu tablo sistemde bulunan bölümlerin tutulduğu tablodur. Bu bölümler öğretim elemanı, ders, atanacak ders gibi tablolarla bağlantılıdır. Bu tablodaki kolonlar şu şekildedir:

# Adı	Veri Tipi	Nulls?
1 BOLUMID	NUMBER	N
2 ADI	VARCHAR2(60)	N

Bölüm adı 60 karakter ayrılmıştır. Çünkü değişik bölüm isimlerine baktığımızda bu kadar uzun bolum adı ile karşılaşılmıştır. Örnek: Sahne Sanatları Bölümü  
Opera – Koro ve Popüler Müzik Şarkıcılığı Ana sanat Dalı

### 3.3.8.3 Ders Tablosu

Bu tablo sistemdeki bütün derslerin tutulduğu tablodur. Bu dersler bu dönem atanacak ya da atanmayacak olabilir. Bu tablodaki kolonlar şu şekildedir:

# Adı	Veri Tipi	Nulls?
1 DERSID	NUMBER	N
2 DERSKODU	VARCHAR2(10)	N

3	TEORIK	VARCHAR2(2)	N
4	PRATIK	VARCHAR2(2)	N
5	KREDI	VARCHAR2(2)	N
6	ECTS	VARCHAR2(2)	Y
7	DERSADI	VARCHAR2(50)	Y
8	SAAT	VARCHAR2(2)	Y
9	ISTERLERI	VARCHAR2(3)	Y
10	TERCIH_EDILEN_DERSLIK1	NUMBER	N
11	TERCIH_EDILEN_DERSLIK2	NUMBER	N
12	TERCIH_EDILEN_DERSLIK3	NUMBER	N
13	TERCIH_EDILEN_LAB1	NUMBER	N
14	TERCIH_EDILEN_LAB2	NUMBER	N
15	TERCIH_EDILEN_LAB3	NUMBER	N

Ders adi kısmı 50 karakter ayrılmıştır. Çünkü değişik bölüm isimlerine baktığımızda bu kadar uzun ders adı ile karşılaşılımıştır. Örneğin, Isıtma Soğutma İklimlendirme Sistem Tasarımı dersi 44 karakter olmaktadır.

#### 3.3.8.4 Derslik Tablosu

Bu tablo sistemin yerleştirme yapacağı dersliklerin tutulduğu tablodur. Bu derslikler dolu ise muadili derslikid'si de tutularak o dersliğe ataması sağlanır. Bu tablodaki kolonlar şu şekildedir:

# Adı	Veri Tipi	Nulls?
1 DERSLIKID	NUMBER	N
2 DERSLIKNO	VARCHAR2(11)	N
3 KAPASITESI	NUMBER	N
4 SIRATIPI	NUMBER	N
5 ICERIGI	NUMBER	N
6 MUADILI_DERSLIK_ID	NUMBER	Y

### 3.3.8.5 Derslik Programı Tablosu

Bu tablo atama gerçekleştikten sonra derslikte olan bilgilerin tutulduğu tablodur. Dersliğin programını görüntüleme bu tablodan sağlanmaktadır. Bu tablodaki kolonlar şu şekildedir:

# Adı	Veri Tipi	Nulls?
1 GUNSAATID	NUMBER	N
2 DERSID	NUMBER	N
3 DERSLIKPROGRAMIID	NUMBER	N
4 DERSLIKID	NUMBER	N
5 ATANACAKDERSDURUMUID	NUMBER	N

### 3.3.8.6 Gün ve saat tablosu

Bu tablo haftanın gün ve saatlerinin birleştiği tablodur. Her gün için her saat için bir id vardır. Bunlara göre atama gerçekleşir. Bu tablodaki kolonlar şu şekildedir:

# Adı	Veri Tipi	Nulls?
1 GUNSAATID	NUMBER	N
2 GUN	VARCHAR2(9)	N
3 SAAT	VARCHAR2(10)	N

### 3.3.8.7 Kişi Tablosu

Bu tablo sistemdeki bütün öğrencilerin ve öğretim elemanlarının bulunduğu tablodur. Öğretim elemanın sicil numarası numara kolonuna gelmektedir. Öğrencinin numarası da numara tablosuna gelmektedir. Bir kayıtın öğrenci mi öğretim elemanı mı olduğunu anlamak için numara kolonuna bakılır. Bu kolondaki rakam uzunluğu 4 ise öğretim elemanıdır 8 ise öğrencidir. Bu tablodaki kolonlar şu şekildedir:

# Adı	Veri Tipi	Nulls?
1 NUMARASI	NUMBER	N
2 ADI	VARCHAR2(15)	N



3	SOYADI	VARCHAR2(15)	N
4	UNVANI	NUMBER	Y
5	BOLUMU	NUMBER	Y

### 3.3.8.8 Logon Tablosu

Bu tablo sistemi kullanacak olan kullanıcıların tutulduğu tablodur. İlk ekrandan seçenekleri seçip seçememesinin belirlenmesi bu tabloya göre olur. Çünkü sistemi kullanacak olan kullanıcıların yetkileri bu tabloda tutulmaktadır. Bu tablodaki kolonlar şu şekildedir:

# Adı	Veri Tipi	Nulls?
1 ID	NUMBER	N
2 KULLANICIADI	VARCHAR2(19)	N
3 SIFRE	VARCHAR2(10)	N
4 YETKISI	NUMBER	N

Bu tabloda Yetkisi alanında number veri tipi bulunmaktadır. Burada kullanıcıların yetkisi “yönetici, kullanıcı” şeklinde ayrılmıştır. Yönetici olana 1 değeri, kullanıcı olana da 0 değeri konulmaktadır.

### 3.3.8.9 Öğretim Üyesi Programı Tablosu

Bu tablo ders programı oluşturulurken anlık olarak dolmaktadır. Bu tablo derslik dışındaki ders programlarının görüntülenmesi için kullanılmaktadır. Öğretim elemanı, sınıf ve şube programları bu tabloda bulunmaktadır. Bu tablodaki kolonlar şu şekildedir:

# Adı	Veri Tipi	Nulls?
1 ID	NUMBER	N
2 SICILNO	NUMBER	N
3 GUNSAATID	NUMBER	N
4 DERSID	NUMBER	N
5 DERSLIKID	NUMBER	N

6	SUBEKODU	VARCHAR2(3)	Y
7	BOLUMID	NUMBER	Y
8	ATANACAKDERSDURUMUID	NUMBER	Y

### 3.3.8.10 Öğretim Üyesi Sıkışıklığı Tablosu

Bu tablo öğretim elemanının sıkışıklıklarının bulunduğu tablodur. Normal olarak uygulamadan girilecek olan ya da internet üzerinden girilecek olan sıkışıklıklar bu tabloda bulunacaktır. Bu tablodaki kolonlar şu şekildedir:

# Adı	Veri Tipi	Nulls?
1 ID	NUMBER	N
2 OGRETIMUYESID	NUMBER	N
3 GUNSAATID	NUMBER	Y
4 SINIRLILIGI	VARCHAR2(3)	Y
5 DURUM	VARCHAR2(1)	Y

### 3.3.8.11 Sınıf İçerikleri Listesi Tablosu

Bu tablo sınıflarda ne gibi gereçlerin bulunduğunu, bununla beraber sınıfların ne ihtiyaç için oluşturulduğu bulunmaktadır. Yani sınıflarda projeksiyon, kimya gereçleri, fizik gereçleri gibi özelliklerin tutulduğu tablodur. Bu tablodaki kolonlar şu şekildedir:

# Adı	Veri Tipi	Nulls?
1 SINIF_ICERIKLERI_ID	NUMBER	N
2 ACIKLAMA	VARCHAR2(63)	N
5 SIRASI	VARCHAR2(3)	N

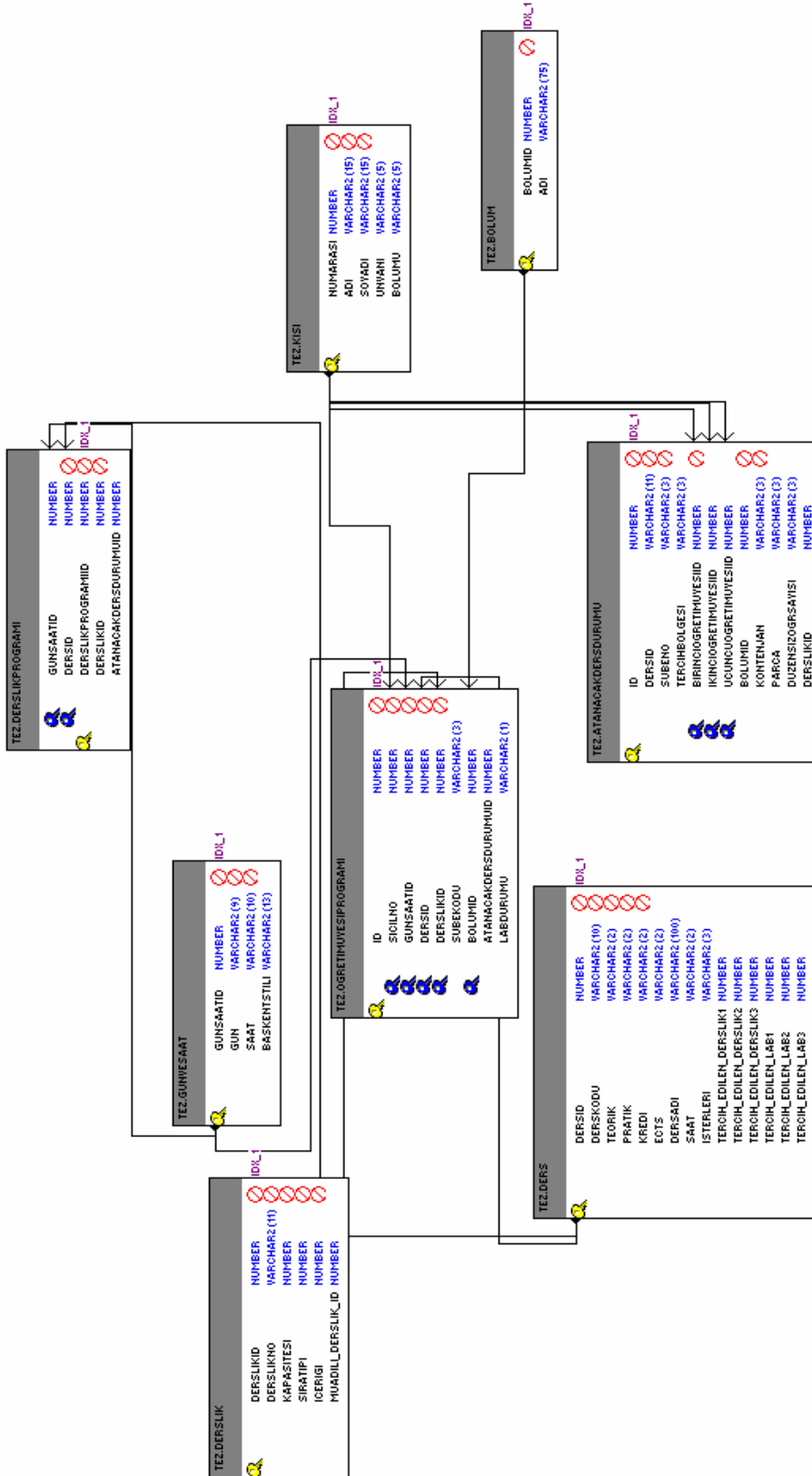
### 3.3.8.12 Unvan Tablosu

Bu tablo öğretim üyelerinin unvanlarının bulunduğu tablodur. Bütün unvanlar bu tabloda bulunuyor olup öğretim elemanın kullanıldığı yerlerde bu unvanlar da gelecektir. Bu tablodaki kolonlar şu şekildedir:

#	Adı	Veri Tipi	Nulls?
1	UNVANID	NUMBER	N
2	ADI	VARCHAR2(12)	Y

### 3.3.9 Veri Tabanı Tablolarının Birbirleri ile Bağlantıları

Veri tabanı tasarımı yapılırken ilişkisel tasarım kullanılmıştır. Yani tablolar birbirlerine constraint ile bağlanmıştır. Bir tablonun birincil anahtarı diğer bir tablonun yabancı anahtarı olarak bulunmaktadır.



Şekil 29 . Veri Tabanı Tabloları Bağlantıları

Yukarıdaki görüntüde veritabanı tablolarının birbirleri ile bağlantılı olduğu tabloları göstermektedir.

### **3.4 Tez Çalışmasının İşletilmesinden Sonraki Değerlendirmeler**

Tez çalışması çerçevesinde yazılan program tamamlandığında, tez çalışması Başkent Üniversitesi Mühendislik Fakültesi'nin verileri ile daha önce belirtildiği gibi dört senaryo (doluluk > %90, %90 > doluluk > %80 yoğun, %80 > doluluk %70 tipik ve %60 kolay modları ) çerçevesinde çalıştırılmıştır.

- A) Bu modların oluşturulmasında ders eksiltme değil, derslik eksiltme ya da fazlalaştırma kullanılmıştır.
- B) Şekil 26'da görüleceği gibi yoDep V1 ön işlem modülü, durumu incelemiş ve işin başlangıcında kullanıcıyı mükemmelen yönlendirmiştir.
- C) Örnek olarak öğretim üyeleri için atama öncesinde tercih edilmeyen zamanlar doldurulmuş idi, yoDep V1 ataması sırasında sistem bu kısıtlara mükemmelen uymuştur. Örneğin;

Öğretim elemanları zaman tercih bildirim ekranı

Sicil No	Soyadı	Adı	Şifre		
1015	KARAKAS	MEHMET ÜMIT	****	Bul	
	Pazartesi	Salı	Çarşamba	Perşembe	Cuma
09 - 10					
10 - 11	0	0	0	0	0
11 - 12	0	0	0	0	0
12 - 13	0	0	0	0	0
13 - 14	0	0	0	0	0
14 - 15	0	0	0	0	0
15 - 16	0	-10	0	-9	0
16 - 17	0	-10	0	-10	0
17 - 18	0	-10	0	-10	0
18 +	0	0	0	0	0
	0	0	0	0	0

Temizle Kaydet

\* Kesinlikle gelemeyeceğiniz yerlere -10 değerini veriniz.  
\* Diğer tercihlerinizi -1 ile -10 arasında derecelendiriniz  
\* Toplam -60'dan küçük değer girilirse sistem bu değeri otomatik yükseltecektir.

Şekil 30. Örnek öğretim üyesi tercih edilmeyen zaman

prof. dr. KARAKAS, MEHMET ÜMIT DERS PROGRAMI

	Pazartesi	Salı	Çarsamba	Persembe	Cuma
1. Dilim				BIL382- () B303	
2. Dilim				BIL382- () B303	BIL382- () B303
3. Dilim				BIL382- () B303	
4. Dilim					
5. Dilim					
6. Dilim	BIL474- () B308		BIL474- () B308		BIL474- () B308
7. Dilim	BIL474- () B308				
8. Dilim					
9. Dilim					
10. Dilim					

**Şekil 31. Örnek öğretim üyesi ders programı**

Yukarıda Ümit Karakaş için Salı öğleden sonra ve Perşembe öğleden sonra kapatılmıştır. Daha sonra program oluşturulduktan sonra Ümit Karakaş'a kapattığı yerlere ders ataması yapılmadığı görülmüştür.

Bununla beraber ders programı çok tatmin edici görünmeyen sınıfın ders programını inceleyerek;

## Makina Mühendisliği 4. Sınıf . Şube DERS PROGRAMI

	Pazartesi	Salı	Çarşamba	Perşembe	Cuma
1. Dilim	MAK496-0 B305	MAK440-0 B305	MAK431-0 B305	MAK440-0 B305	
2. Dilim	MAK496-0 B305	MAK440-0 B305	MAK431-0 B305	MAK431-0 B305	
3. Dilim					
4. Dilim			MAK496-0 B305		
5. Dilim					
6. Dilim	MAK402-0 B305	MAK402-0 B305	MAK402-0 B305		ENG400- (002) C304
7. Dilim		MAK402-0 B305	MAK402-0 B305		ENG400- (002) C304
8. Dilim					ENG400- (002) C304
9. Dilim					
10. Dilim					

### Şekil 32. Makine Mühendisliği 4. sınıf ders programı

Burada Salı günü içerisindeki boşluklar ve Çarşamba günü dersler arasındaki boşlukların neden öyle olduğunu derslik ders programından görebilmekteyiz.



## B305 DERS PROGRAMI

	Pazartesi	Sali	Çarsamba	Persembe	Cuma
1. Dilim	MAK496-0 B305	MAK440-0 B305	MAK431-0 B305	MAK440-0 B305	MAK208- (002) B305
2. Dilim	MAK496-0 B305	MAK440-0 B305	MAK431-0 B305	MAK431-0 B305	MAK208- (002) B305
3. Dilim		END409-0 B305	MAK303-0 B305	MAK208- (002) B305	MAK208- (002) B305
4. Dilim	END409-0 B305	END409-0 B305	MAK496-0 B305	END409-0 B305	BIL412-0 B305
5. Dilim					
6. Dilim	MAK402-0 B305	MAK402-0 B305	MAK402-0 B305	MAK206- (002) B305	END204- (001) B305
7. Dilim	MAK206- (002) B305	MAK402-0 B305	MAK402-0 B305	MAK206- (002) B305	END204- (001) B305
8. Dilim	END204- (001) B305	BIL401-0 B305		END204- (001) B305	
9. Dilim		BIL401-0 B305			
10. Dilim					

### Şekil 32. B305 dersliđinin haftalık ders programı

Yukarıdaki derslik programında Sali sabah ve Çarşamba sabahın tamamıyla dolu olduđu görölmektedir.

D) Asıl kritik olan “yođun moda (doluluk %70 ile %80 arasında)” Bařkent Üniversitesi'nin Güz 2007 – 2008 gerçek verileri kullanılarak yapılan atama ařađıda adı belirtilen öđretim üyelerine iřletim zamanında izlettirilmiřtir. yoDep V1 iřletimi bařarı ile tamamlanmıřtır.

D1 ) Yazılım, yođun moda, verileri alıp atamaya bařladıktan sonra Pentium 4 iřlemci, 2048 MB bellek ve 20GB'dan büyük disk ortamında İstanbul Kültür Üniversitesi'nde, bir öđretim elemanının gözetiminde 1 dakika da Bařkent Üniversitesi'nin ders programı atamasını yapmıřtır.

D2 ) Yazılım, yođun moda, verileri alıp atamaya bařladıktan sonra Bařkent Üniversitesi'nde Centrino Duo iřlemcili, 1024 MB bellek ve 20GB

dan büyük disk ortamında iki öğretim elemanının gözetiminde yaklaşık 2 dakikada Başkent Üniversitesi'nin ders programı atamasını yapmıştır.

- E) Ders atamaları yapıldıktan ve diske yazıldıktan sonra tek tuş hareketi ile öğretim üyesi (dr ya da asistan), derslik / laboratuvar ve 20 bölüm – sınıf ders programı 2 dakikada dökülmüştür.
- F) Ders programını yapan öğretim üyesi her yarıyıl yaklaşık 2 iş günü kullandığını, yoDep V1 'in verilerin yüklenmesi için gerekli 2 günü düşüldüğünde her yarıyıl 2 işgünü bölüm başkan yardımcısına zaman kazandırdığı tespit edilmiştir.
- G) Son ve en önemli olan nokta ise oluşturulan ders programında tarafların tatminidir. yoDep sistemi atamaları bitirdiğinde tarafların tatmin ortalamalarının yaklaşık olarak 0,92 olduğu işaret edilmekteydi. Bu çalışmaya zaman ayıran öğretim üyelerine (Prof. Dr. Ümit KARAKAŞ, Yrd.Doç.Dr.Hasan OĞUL, Yrd.Doç.Dr. Ergün Eraslan, Yrd. Doç. Dr. Kemal YÜKSEK, Öğr.Gör. Yusuf ALTUNEL) “yoDep'in görünür akıllı ajan düzeyini beş puanlık skala üzerinde değerlendirir misiniz?” sorusu yöneltilmiştir. Skala şöyledir:
- 5: Mükemmel / Çok zeki  
4: Oldukça zeki  
3: Orta zekâda  
2: Kötü bir performans  
1: Çok kötü

G maddesindeki değerlendirme sonuçları şöyledir:

Yusuf ALTUNEL dışındaki öğretim üyeleri 4 cevabını vermiştir. Burada neden 4 diye sorulduğunda daha iyi sonuçların çıkabileceği söylenmiştir. Yusuf ALTUNEL 3 cevabını vermiştir. Neden sorusunu sordumuzda ise şu an için kendisinin işini görmeyeceğini belirtmiştir.

#### 4 SONUÇ

Bu çalışmada kuramsal ön çalışması, tasarımı, programlanması ve işletilmesi ile oluşturulan yoDep V1(Yarı Otomatik Ders Programı Oluşturma Sistemi, Sürüm 1) başarı ile tamamlanmıştır.

Buradaki teknik problem atanacak 166 ders (genellikle her biri haftanın iki ayrı saatine), 85 öğretim üyesinin ve 33 derslik/laboratuvarın haftalık programıdır. Sınırlamalar ve tercih anketleri sonuçları ile daraltıldıktan sonra dahi başlangıç noktasında 3000 ataması olan ağaç yapısı bulunmaktadır. Oluşturulan sezgisel fonksiyon kullanılarak dört senaryo ile çalıştırılmıştır. Yaklaşık 166 çift atama yapılmış ve çok az geriye dönüş (backtracking) ile atama tamamlanmıştır. Atama yaklaşık olarak 1 ile 2 dakika arasında gerçekleşmiştir. Ders programlarının dökülmesi için çok kısa zaman harcanmıştır.

Yapılan otomatik atamanın zeka düzeyi beş öğretim üyesi, tarafından beşlik skala (5 : yoDep çok zeki) üzerinden ortalama olarak değerlendirilmiştir. 4 öğretim elemanı 4 değerini ve 1 öğretim elemanı 3 değerini vermişlerdir. Yani ortalama 3,8 ile orta zekâ ile oldukça zeki arasında çıkmıştır; ama oldukça zekiye daha yakın çıkmıştır.

Bunun yanında, sistemin kullanılmasının bölüm başkan yardımcısına her yarıyıl 2 işgünü, bölüm sekreterine ise 5 işgünü kazandıracak kestirilmiştir.

Bununla beraber piyasadaki çizelgeleme yazılımlarının lisans fiyatlarının yüksek olması ve genellikle ilköğretim ile orta öğretim ders programlarını yapması yoDep'in avantajlarıdır.

Bundan sonra yapılacak işlere baktığımızda farklı dil destekleri olan yazılımlar mevcut olduğu için farklı dil seçeneği eklenebilir.

Öğrenciler için tek gün ya da herhangi bir zaman dilimi kapatılabilir. Buradaki amacımız son sınıf öğrencilerinin bir ya da daha fazla gününü kapatarak çalışmalarına olanak sağlamaktır. Bununla beraber seminer ya da toplantı için de saatler kapatılabilir.

Aynı dersin farklı şubelerini aynı saat dilimine koymak da bundan sonra yapılacak işler arasında olabilir. Çünkü yazılımı inceleyen birkaç öğretim elemanı

tarafından bu türlü atamanın olması istenmiştir. Bu kısıtın sağlanması için derslerin aynı öğretim elemanı tarafından veriliyor olmaması gerekmektedir. Düzensiz öğrenciler için (farklı dönemlerde ders almak zorunda olan öğrenciler) çapraz müfredat dönemlerinde derslerin çakışması engellenebilir.

## 5 Kaynaklar Listesi

- 1) Abramson, D. (1991). "Constructing School Timetables Using Simulated Annealing: Sequential and Parallel Algorithms", *Management Science*, 37(1):98-113.
- 2) Hertz, A. (1992). "Finding a feasible course schedule using a tabu search", *Discrete Applied Mathematics*, 35: 255-270.
- 3) Schaerf, A. (1996). "Tabu Search Techniques for Large High-School Timetabling Problems", *Proc. of the Fourteenth National Conference on AI*, August, 363- 368.
- 4) Russell, S., Norvig, P., "Artificial Intelligence: A Modern Approach", Prentice-Hall, 1995
- 5) Colomi, A., Dorigo, M., Maniezzo, V., (1990), "Genetic Algorithm and Highly Constrained Problems: The Time-Table Case", *Proceeding of the First International Workshop on Parallel Problem solving from Nature*, 55-59.
- 6) Erben, W., Keppler, J. (1995). "A Genetic Algorithm Solving a Weekly Course - Timetabling Problem", *Proc. of the First Int. Conf. on the Practice and Theory of Automated Timetabling (ICPTAT)*, 21-32.
- 7) Cladeira, J.P., Rosa, A.C., (1997). "School Timetabling using Genetic Search", *The 2nd International Conference for the Practice and Theory of Automated Timetabling (PATAT'97)*, 115-122.
- 8) Glover, F., 1989, "Tabu Search - Part I", *ORSA Journal on Computing*, Summer, Vol.1, No.3, pp.190-206.

- 9) Reich Y., Barai S.V., (2000), "A methodology for building neural networks models from empirical engineering data", Engineering Applications of Artificial Intelligence, Volume 13, Number 6, pp. 685-694(10)
- 10) Sinriech D. ve Samakh E. (1999), "A Genetic Approach to the Pickup/Delivery Station Location Problem in Segmented Flow Based Material Handling Systems", Journal of Manufacturing Systems, Volume: 18, Number: 2, s. 81-99.
- 11) BEASLEY, D., BULL, D.R., and MARTIN, R.R., 1993a. An Overview of Genetic Algorithms: Part 1, Fundamentals. University Computing, Vol.15(2), pp. 58-69, UK.
- 12) BEASLEY, D., BULL, D.R., and MARTIN, R.R., 1993b. An Overview of Genetic Algorithms: Part 2, Research Topics .University Computing, Vol. 15(4), pp. 170-181, UK.
- 13) Dasgupta, P., & Khazanchi, D. (2005). Adaptive decision support for academic course scheduling using intelligent software agents. International Journal of Technology in Teaching and Learning,1(2), 63-78
- 14) E. Ozcan, A. Alkan, (2002) Çok Nüfuslu Kararlı Hal Genetik Algoritması Kullanarak Otomatik Çizelgeleme, 19. Bilisim Kurultayı Bildiriler Kitabı, s. 149-154.
- 15) <http://themis.hilmibaycan.com/>
- 16) O. Engin, A. Figlali, (2002) "Akış Tipi Çizelgeleme Problemlerinin Genetik Algoritma Yardımı ile Çözümünde Uygun Çaprazlama Operatörünün Belirlenmesi", Dogus Üniversitesi Dergisi 2002/6, s.27-35

- 17) Lien-Fu Lai, Nien-Lin Hsueh, Liang-Tsung Huang, Tien-Chun Chen, (2006) "An Artificial Intelligence Approach to Course Timetabling" ictai, 18th IEEE International Conference on Tools with Artificial Intelligence (ICTAI'06), pp. 389-396
- 18) Maes, P. (1994), "Agents that Reduce Work and Information Overload", Communications of the ACM 37 (7), 31-40
- 19) S. Kirkpatrick and C. D. Gelatt and M. P. Vecchi, 1983, Optimization by Simulated Annealing, Science, Vol 220, Number 4598, pages 671-680.
- 20) Jeffrey Bagdis, 2007, A Machine-Learning Approach to Computer Go.
- 21) Peterson, I., 1996 "The Soul of a Chess Machine: Lessons Learned from a Contest Pitting Man against Computer." Sci. News 149, 200-201, Mar. 30.
- 22) Karakas, U., 1995 "Yazılım Ön Kestirim Yöntemleri" Bilişim 95 TBD 12. Ulusal Bilişim Kurultayı İstanbul.

## 6 Ekler Listesi

### EK 1 ANKET SORULARI

- | Soru No | Soru   |
|---------|--|
| 1       | Dersim ilk saat diliminde (8:40, 8:50, 9:00) başlamalıdır.   |
| 2       | Dersim ikinci saat diliminde (9:40, 9:50, 10:00) başlamalıdır  |
| 3       | Yemek arasını erken ( 12:00, 13:00 ) tercih ederim.  |
| 4       | Yemek arasını geç ( 13:00, 14:00 ) tercih ederim.  |
| 5       | Erken ya da geç mutlaka bir yemek arası gereklidir   |
| 6       | Öğle saatlerinde iki saat sosyal etkinlik arası istiyorum.   |
| 7       | Ders yüküm haftanın beş gününe eşit dağılmalıdır.  |
| 8       | Ders yüküm haftanın dört buçuk gününe eşit dağılmalıdır.   |
| 9       | Ders yüküm haftanın dört gününe eşit dağılmalıdır.   |
| 10      | Ders yüküm haftanın iki gününe eşit dağılmalıdır.  |
| 11      | Ders yüküm haftanın bir gününe yoğunlaşmalıdır.  |
| 12      | Derslerimin akşam bitiş saati mümkün olduğunca erken olmalıdır.  |
| 13      | Teknik yönden zor dersler mümkün olduğunca erken başlamalıdır.   |
| 14      | Sosyal dersler, tarih ve Türkçe erken saatte başlamalıdır.   |
| 15      | İngilizce dersleri erken saatte başlamalıdır.  |
| 16      | Seçmeli dersler dördüncü saat diliminde ( 11:40, 11:50, 12:00 ) başlamalıdır.  |
| 17      | Seçmeli dersler beşinci saat diliminde ( 12:40, 12:50, 13:00 ) başlamalıdır.   |
| 18      | YL dersim 9. saat diliminde ( 16:40, 16:50, 17:00 ) başlamalıdır.  |
| 19      | YL dersim 10. saat diliminde ( 17:40, 17:50, 18:00 ) başlamalıdır.   |
| 20      | Cumartesi sabahı 10:00 da ders kabul ederim.   |
| 21      | İşgünü 18:00 da başlayan YL desini yerine Pazar günü ders isterim.   |
| 22      | Cuma günü 18:00 da başlayan bir ders benim için kabuldür.  |
| 23      | "İlk ders 8:50de başlamalıdır, ilk dört dersten sonra 5. ders saati 12:40 - 13:30 da başlamalıdır. Altıncı - dokuzuncu ders saati 13:30 da başlamalıdır. |



Dokuzuncu ders saati bitiři 17:20 olacaktır. Beřinci ders saati sadece 10:50 de bařlayan üç saatlik blok ders, 12:40 da bařlayan üç saatlik blok ders vb biçiminde kullanılmalıdır. ( var olan durumda 9:00 - 17:00 arasında sekiz ders saati vardır ve ders programı yapılmasında güçlükler bulunmaktadır )"

## EK 2 BOLUM LISTESI

- 1 Bilgisayar Mühendisliđi
- 2 Biyomedikal Mühendisliđi
- 3 Elektrik Elektronik Mühendisliđi
- 4 Endüstri Mühendisliđi
- 5 Makine Mühendisliđi

## EK 3 DERSLER LİSTESİ

BIL101	BİLGİSAYAR YAZILIMI I
BIL102	BİLGİSAYAR YAZILIMI II
BIL105	PROGRAMLAMA LABORATUVARI I
BIL106	PROGRAMLAMA LABORATUVARI II
BIL162	BETİK DİLLER
BIL232	ELEKTRİK DEVRELERİ
BIL240	PROGRAMLAMA DİLLERİ
BIL252	VERİ YÖNETİMİ VE DOSYA YAPILARI
BIL256	VERİ İLETİŞİMİ
BIL284	ÇEVİRİCİ DİLLER
BIL300	STAJ I
BIL334	İŞLETİM SİSTEMLERİ
BIL352	VERİTABANI YÖNETİM SİSTEMLERİ
BIL364	OTOMATA TEORİSİ
BIL375	BİLGİSAYAR ORGANİZASYONU
BIL382	YAZILIM MÜHENDİSLİĞİNE GİRİŞ
BIL400	STAJ II
BIL401	HESAPLAMALI GRAFİK
BIL412	İMGE İŞLEME
BIL413	KRİPTOGRAFI VE GÜVENLİK
BIL441	ÇİZGE KURAMI
BIL474	BİLGİSAYAR GRAFİĞİNE GİRİŞ
BIL483	YÖNETİM BİLİŞİM SİSTEMLERİ
BIL491	BITİRME PROJESİ I
BIL492	BITİRME PROJESİ II
BME204	İNSAN ANATOMİSİ VE FİZYOLOJİSİ
BME212	BIYOMALZEME
BME222	ELEKTRONİK
BME234	ELEKTROMANYETİK TEORİ
BME300	STAJ I
BME308	MÜHENDİSLER İÇİN FİZYOLOJİ II
BME314	BIYOMEDİKAL ENSTRÜMANTASYON I
BME332	BIYOMEDİKAL MÜH.DE BİLGİSAYAR UYGULAMALARI
BME400	STAJ II
BME402	BIYOMEDİKAL İŞARET İŞLEME II

BME414 TIBBİ TEKNOLOJİ YÖNETİMİ  
BIYOMEDİKAL MÜHENDİSLİĞİNDE SEÇME KONULAR

BME440 II

BME445 BIYOMEDİKAL GÖRÜNTÜ İŞLEME

BME491 BITİRME PROJESİ I

BME492 BITİRME PROJESİ 2

EDE100 EDEBİYAT

EEM202 DEVRE TEORİSİ II

EEM214 ELEKTRONİK I

EEM224 ELEKTROMANYETİK I

EEM300 STAJ I

EEM302 SINYALLER VE SİSTEMLER II

EEM312 SAYISAL ELEKTRONİK

EEM322 DENETİM SİSTEMLERİ

EEM332 MIKROİŞLEMCİLER

EEM400 STAJ II

EEM423 ANTENLER VE YAYILIM

EEM438 VERİ İLETİSİMİ

EEM442 HABERLEŞME II

EEM462 SAYISAL DENETİM SİSTEMLERİ

EEM474 GÜÇ ELEKTRONİĞİ

EEM491 BITİRME PROJESİ 1

EEM492 BITİRME PROJESİ 2

END202 YÖNEYLEM ARAŞTIRMASI I

END204 OLASILIK VE İSTATİSTİKTE GİRİŞ

END232 ÜRETİM YÖNTEMLERİ

END300 STAJ I

END304 ÜRETİM SİSTEMLERİ VE TESİS PLANLAMASI

END306 RASSAL MODELLER

END308 BENZETİM

END310 ÜRETİM PLANLAMA VE KONTROLÜ I

END312 İŞBİLİM

END400 STAJ II

END409 DENEY TASARIMI

END415 TAHMİN TEKNİKLERİ

END450 MESLEKİ YETERLİLİK

END491 BITİRME PROJESİ 1

END492 BITİRME PROJESİ 2  
ENG100 İNGİLİZCE  
ENG200 İNGİLİZCE 2  
ENG300 İNGİLİZCE 3  
ENG400 İNGİLİZCE4  
FİZ104 FİZİK 1  
FİZ110 FİZİK 2  
KİM110 KİMYA 1  
KİM114 KİMYA 2  
KİM122 KİMYA 3  
MAK104 TEKNİK RESİM II  
MAK200 ATÖLYE STAJI  
MAK202 MUKAVEMET  
MAK204 TERMODİNAMİK II  
MAK206 DİNAMİK  
MAK208 ÜRETİM TEKNOLOJİLERİ  
MAK212 SAYISAL ANALİZ  
MAK300 STAJ I  
MAK302 ELEKTRONİK  
MAK303 ÜRETİM MÜHENDİSLİĞİ  
MAK306 MAKİNE ELEMANLARI II  
MAK308 AKIŞKANLAR MEKANİĞİ II  
MAK312 SİSTEM DİNAMİĞİ VE KONTROLÜ  
MAK400 STAJ II  
MAK402 MAKİNE MÜHENDİSLİĞİ LABORATUVARI  
MAK431 OTOMOTİV MÜHENDİSLİĞİ  
ISITMA SOĞUTMA İKLİMLENDİRME SİSTEM  
MAK440 TASARIMI  
MAK492 BITİRME PROJESİ II  
MAK496 MEKATRONİK SİSTEM TASARIMI  
MAT101 MATEMATİK 1  
MAT105 MATEMATİK 2  
MAT151 MATEMATİK 3  
MAT152 MATEMATİK 4  
MAT210 DİFERANSİYEL 1  
MAT250 DİFERANSİYEL 2  
SOS204 SOSYOLOJİ 1

SOS222 SOSYOLOJI 2  
TAR200 INKILAP TARIHI

## EK 4 UNVAN LİSTESİ

1	prof. dr.
2	doç. dr.
3	ydoc. dr.
4	dr.
5	ögr. gör.
6	asistan
7	bursiyer
8	okutman
9	öğrenci
10	yl. öğrenci

## EK 5 ÖĞRETİM ELEMANLARI LİSTESİ

- 1 HAKAN ÇAGLAR
- 2 HASMET GÜRÇAY
- 3 NIZAMI GASSILOV
- 4 SERDAR MÜLDÜR
- 5 SEREF SAGIROĞLU
- 6 BÜLENT GÜRSEL EMİROĞLU
- 7 GÜVEN KÖSE
- 8 EMRE SÜMER
- 9 HALUK NALBANTOĞLU
- 10 İBRAHİM BARAN USLU
- 11 KRASSİMİR İANKOV
- 12 MUHAMMET YORULMAZ
- 13 NESLİŞAH TOROSDAGLI
- 14 OGUL GÖÇMEN
- 15 MEHMET ÜMİT KARAKAS
- 16 TOFIK MAMEDOV
- 17 FATİH VEHBİ ÇELEBİ
- 18 MUSTAFA SERT
- 19 OSMAN EROGUL
- 20 ARIF KOÇOĞLU
- 21 AYKUT ERDAMAR
- 22 DİLEK ÇÖKELİLER
- 23 FAZİL DUMAN
- 24 METİN YILDIZ
- 25 KEMAL SERBETÇİ
- 26 HÜSEYİN AKÇAY
- 27 İNAN GÜLER
- 28 BÜLENT YILMAZ
- 29 YURİY ALYEKSYEYENKOV
- 30 BİRSEN SAKA TANATAR
- 31 İSMAIL ENİS UNGAN
- 32 ADNAN KÖKSAL
- 33 ALPER URAZ
- 34 MURAT EMİN AKATA
- 35 TURHAN ÇİFTÇİBASİ
- 36 AYSEL SAFAK
- 37 HAMİT ERDEM
- 38 İSMAIL AVCİBAS
- 39 MUSTAFA DOĞAN
- 40 FULYA ALTIPARMAK



41 FAHRI BUĞRA ÇAMLICA  
42 PELİN TOKTAS  
43 BERNA DENGİZ  
44 EKREM PAKDEMİRLİ  
45 FATMA ZEHRA MÜLÜK  
46 İMDAT KARA  
47 İSMAİL ERDEM  
48 SERPİL EROL  
49 ERGÜN ERASLAN  
50 FATMA PAKDİL  
51 MUZAFFER KAPANOGLU  
52 İBRAHİM USLAN  
53 MEHMET AKÇAY  
54 ANDAÇ TÖRE SAMİLOGLU  
55 BESİM BARANOGLU  
56 LEVENT ÇOLAK  
57 BEDRİ TUÇ  
58 CAN ÇOGUN  
59 NEVZAT ONUR  
60 ÖMER FARUK ELALDI  
61 AHMET HAKAN ARGESO  
62 AHMET ONUR  
63 BEDI CENK BALÇIK  
64 BAHATTİN CENGİZ  
65 BELGİN ÖZKUL  
66 KAYA KILAN  
67 MÜJDAT KAYA  
68 NECMETTİN TANRIÖVER  
69 GÜRSEL EMİROGLU  
70 ELİF MEDETOGULLARI  
71 GÜLER ERGÜN EFE  
72 NURAY ÇELEBİ  
73 MUSTAFA AZREG  
74 BAHAR ÖZDOĞAN  
75 CENGİZ KOÇUM  
76 SİTKİ ÇAĞDAS İNAM  
77 EROL TÜMER  
78 CEVDET TEZCAN  
79 İNGİLİZCE1 İNGİLİZCE1  
80 İNGİLİZCE2 İNGİLİZCE2  
81 İNGİLİZCE3 İNGİLİZCE3  
82 İNGİLİZCE4 İNGİLİZCE4  
83 İSMET GÖRGÜLÜ

84  
85

TÜRKÇE TÜRKÇE  
HASAN OGUL

EK 6 Oluşturulan Ders Programları

**Bilgisayar Mühendisliği 1. Sınıf 001. Şube DERS PROGRAMI**

	Pazartesi	Salı	Çarşamba	Perşembe	Cuma
1. Dilim	ENG100- (001) B311	EDE100- 0 A305	BIL162 (LAB) - (001) LAB2	BIL102- (001) C306	FIZ110- (001) B310
2. Dilim	ENG100- (001) B311	EDE100- 0 A305	BIL162 (LAB) - (001) LAB2	FIZ110- (001) B310	FIZ110- (001) B310
3. Dilim	ENG100- (001) B311	FIZ104- (001) C407		FIZ104- (001) C407	BIL102- (001) C306
4. Dilim		FIZ104- (001) C407			BIL102- (001) C306
5. Dilim					
6. Dilim		MAT152- (001) C304	MAT152- (001) C303	BIL106- (001) C407	BIL162- (001) C303
7. Dilim		MAT152- (001) C304	BIL106- (001) C407		BIL162- (001) C303
8. Dilim					
9. Dilim					
10. Dilim					

## Bilgisayar Mühendisliği 1. Sınıf 002. Şube DERS PROGRAMI

	Pazartesi	Sali	Çarsamba	Persembe	Cuma
1. Dilim	FIZ104- (002) C407	EDE100- 0 A305	FIZ110- (002) B310	FIZ104- (002) C407	BIL102- (002) C306
2. Dilim	FIZ104- (002) C407	EDE100- 0 A305	FIZ110- (002) B310	BIL102- (002) C306	BIL102- (002) C306
3. Dilim	FIZ110- (002) B310	BIL162 (LAB) - (002) LAB2			
4. Dilim		BIL162 (LAB) - (002) LAB2			
5. Dilim					
6. Dilim	ENG100- (002) B311		BIL106- (002) C407		
7. Dilim	ENG100- (002) B311			BIL162- (002) C303	
8. Dilim	ENG100- (002) B311			BIL162- (002) C303	BIL106- (002) C407
9. Dilim					
10. Dilim					

## Bilgisayar Mühendisliği 2. Sınıf 001. Şube DERS PROGRAMI

	Pazartesi	Salı	Çarsamba	Perembe	Cuma
1. Dilim	ENG200- (001) C303	MAT210- (001) C309	TAR200- 0 C310	MAT250- (001) B312	BIL240- (001) B312
2. Dilim	ENG200- (001) C303	MAT210- (001) C309	TAR200- 0 C310	MAT250- (001) B312	BIL240- (001) B312
3. Dilim	ENG200- (001) C303	MAT250- (001) B312	TAR200- 0 C310	BIL256- (001) B406	BIL284- (001) A310
4. Dilim	BIL284- (001) A310	BIL284- (001) A310	MAT210- (001) C309	BIL256- (001) B406	BIL284- (001) A310
5. Dilim					
6. Dilim	BIL240 (LAB) - (001) LAB2	BIL252 (LAB) - (001) LAB2	BIL232- (001) B304	BIL232- (001) B304	
7. Dilim	BIL240 (LAB) - (001) LAB2	BIL252 (LAB) - (001) LAB2	BIL232- (001) B304	BIL232- (001) B304	BIL252- (001) B307
8. Dilim					BIL252- (001) B307
9. Dilim					
10. Dilim					

## Bilgisayar Mühendisliği 2. Sınıf 002. Şube DERS PROGRAMI

	Pazartesi	Salı	Çarşamba	Perşembe	Cuma
1. Dilim	MAT210- (002) C309	MAT250- (002) B312	TAR200- () C310	BIL232 (LAB) - (002) LAB6	BIL256- (002) B406
2. Dilim	MAT210- (002) C309	MAT250- (002) B312	TAR200- () C310	BIL232 (LAB) - (002) LAB6	BIL256- (002) B406
3. Dilim	BIL240- (002) B312		TAR200- () C310	MAT250- (002) B312	BIL240- (002) B312
4. Dilim	BIL240- (002) B312			MAT210- (002) C309	BIL240- (002) B312
5. Dilim					
6. Dilim	ENG200- (002) C303	BIL232- (002) B312	BIL252- (002) B312	BIL252- (002) B312	
7. Dilim	ENG200- (002) C303	BIL232- (002) B312	BIL252- (002) B312	BIL252- (002) B312	
8. Dilim	ENG200- (002) C303	BIL284- (002) A309	BIL284- (002) A309		BIL284- (002) A309
9. Dilim		BIL284- (002) A309			
10. Dilim					

### Bilgisayar Mühendisliği 3. Sınıf 001. Şube DERS PROGRAMI

	Pazartesi	Salı	Çarsamba	Perembe	Cuma
1. Dilim	ENG300- (001) C306	BIL352- () B308	BIL364- () B311	BIL382- () B303	BIL352- () B308
2. Dilim	ENG300- (001) C306	BIL352- () B308	BIL364- () B311	BIL382- () B303	BIL382- () B303
3. Dilim	ENG300- (001) C306	BIL352- () B308	BIL364- () B311	BIL382- () B303	
4. Dilim	BIL364- () B311				
5. Dilim					
6. Dilim	BIL334- () B307	BIL334- () B307	BIL334- () B307		
7. Dilim	BIL334- () B307			BIL375- (001) B308	BIL375- (001) B308
8. Dilim	BIL375- (001) B308			BIL375- (001) B308	
9. Dilim					
10. Dilim					

### Bilgisayar Mühendisliği 3. Sınıf 002. Şube DERS PROGRAMI

	Pazartesi	Salı	Çarsamba	Perembe	Cuma
1. Dilim		BIL352- () B308	BIL364- () B311	BIL382- () B303	BIL352- () B308
2. Dilim		BIL352- () B308	BIL364- () B311	BIL382- () B303	BIL382- () B303
3. Dilim		BIL352- () B308	BIL364- () B311	BIL382- () B303	
4. Dilim	BIL364- () B311				
5. Dilim					
6. Dilim			BIL334- () B307	BIL375- (002) B308	
7. Dilim		BIL375- (002) B308	BIL375- (002) B308		
8. Dilim	ENG300- (002) C306				
9. Dilim					
10. Dilim					



## Bilgisayar Mühendisliği 4. Sınıf . Şube DERS PROGRAMI

	Pazartesi	Salı	Çarsamba	Perembe	Cuma
1. Dilim	ENG400- (001) B406			BIL412- 0 B307	
2. Dilim	ENG400- (001) B406			BIL412- 0 B307	
3. Dilim	ENG400- (001) B406		BIL483- 0 C307	BIL412 (LAB) - 0 LAB2	BIL483- 0 C307
4. Dilim	BIL401- 0 B307		BIL483- 0 C307	BIL412 (LAB) - 0 LAB2	BIL412- 0 B305
5. Dilim					
6. Dilim	BIL474- 0 B308	BIL413 (LAB) - 0 LAB15	BIL474- 0 B308	BIL413 (LAB) - 0 LAB15	BIL474- 0 B308
7. Dilim	BIL474- 0 B308	BIL413 (LAB) - 0 LAB15			
8. Dilim		BIL401- 0 B305	BIL441- 0 A307		BIL441- 0 A307
9. Dilim		BIL401- 0 B305	BIL441- 0 A307		
10. Dilim					

## Bilgisayar Mühendisliği 4. Sınıf . Şube DERS PROGRAMI

	Pazartesi	Salı	Çarsamba	Perembe	Cuma
1. Dilim				BIL412- () B307	
2. Dilim				BIL412- () B307	
3. Dilim			BIL483- () C307	BIL412 (LAB) - () LAB2	BIL483- () C307
4. Dilim	BIL401- () B307		BIL483- () C307	BIL412 (LAB) - () LAB2	BIL412- () B305
5. Dilim					
6. Dilim		BIL413 (LAB) - () LAB15	BIL474- () B308	BIL413 (LAB) - () LAB15	BIL474- () B308
7. Dilim		BIL413 (LAB) - () LAB15			
8. Dilim	ENG400- (002) B406	BIL401- () B305	BIL441- () A307		BIL441- () A307
9. Dilim		BIL401- () B305	BIL441- () A307		
10. Dilim					

## Biyomedikal Mühendisliği 1. Sınıf 001. Şube DERS PROGRAMI

	Pazartesi	Sali	Çarsamba	Persembe	Cuma
1. Dilim	FIZ110- (001) B310	ENG100- (001) B311	EDE100- 0 A305	FIZ110- (001) B310	FIZ104- (001) C407
2. Dilim	FIZ110- (001) B310	ENG100- (001) B311	EDE100- 0 A305	FIZ104- (001) C407	FIZ104- (001) C407
3. Dilim	MAT151- (001) A308	ENG100- (001) B311		MAT151- (001) A308	
4. Dilim	MAT151- (001) A308				
5. Dilim					
6. Dilim	KIM114- (001) C307		BIL101- 0 B311	BIL101- 0 B311	KIM114- (001) C307
7. Dilim	KIM114- (001) C307		BIL101- 0 B311	BIL101- 0 B311	KIM114- (001) C307
8. Dilim	BIL105- (001) C407		BIL105- (001) C407		
9. Dilim					
10. Dilim					

## Biyomedikal Mühendisliği 1. Sınıf 002. Şube DERS PROGRAMI

	Pazartesi	Salı	Çarsamba	Perembe	Cuma
1. Dilim		FIZ110- (002) B310	EDE100- () A305	KIM114 (LAB) - (002) LAB15	
2. Dilim		FIZ110- (002) B310	EDE100- () A305	KIM114 (LAB) - (002) LAB15	MAT151- (002) A308
3. Dilim	KIM114- (002) C309	MAT151- (002) A308	BIL105- (002) C407		FIZ110- (002) B310
4. Dilim	KIM114- (002) C309	MAT151- (002) A308		BIL105- (002) C407	
5. Dilim					
6. Dilim	FIZ104- (002) B304	ENG100- (002) B311	BIL101- () B311	BIL101- () B311	MAT152- (002) C310
7. Dilim	FIZ104- (002) B304	ENG100- (002) B311	BIL101- () B311	BIL101- () B311	MAT152- (002) C310
8. Dilim		ENG100- (002) B311	FIZ104- (002) B304	MAT152- (002) C310	
9. Dilim					
10. Dilim					

## Biyomedikal Mühendisliği 2. Sınıf 001. Şube DERS PROGRAMI

	Pazartesi	Salı	Çarsamba	Perembe	Cuma
1. Dilim	BME234- (001) A310	ENG200- (001) C303		TAR200- 0 C310	
2. Dilim	BME234- (001) A310	ENG200- (001) C303		TAR200- 0 C310	
3. Dilim	SOS204- (001) A309	ENG200- (001) C303		TAR200- 0 C310	
4. Dilim	SOS204- (001) A309				
5. Dilim					
6. Dilim	BME212- (001) A307	BME204- (001) A310	BME204- (001) A310	BME212- (001) A307	BME222 (LAB) - (001) LAB5
7. Dilim	BME212- (001) A307	BME204- (001) A310		BME222- (001) A310	BME222 (LAB) - (001) LAB5
8. Dilim				BME222- (001) A310	
9. Dilim					
10. Dilim					

## Biyomedikal Mühendisliği 2. Sınıf 002. Şube DERS PROGRAMI

	Pazartesi	Salı	Çarsamba	Perembe	Cuma
1. Dilim			BME222 (LAB) - (002) LAB6	TAR200- 0 C310	
2. Dilim			BME222 (LAB) - (002) LAB6	TAR200- 0 C310	
3. Dilim		SOS204- (002) A309	BME234- (002) A310	TAR200- 0 C310	
4. Dilim		SOS204- (002) A309	BME234- (002) A310		
5. Dilim					
6. Dilim	BME222- (002) A310	ENG200- (002) C303			BME204- (002) A310
7. Dilim	BME222- (002) A310	ENG200- (002) C303	BME204- (002) A310	BME212- (002) A307	BME204- (002) A310
8. Dilim	BME212- (002) A307	ENG200- (002) C303		BME212- (002) A307	
9. Dilim					
10. Dilim					

### Biyomedikal Mühendisliği 3. Sınıf 001. Şube DERS PROGRAMI

	Pazartesi	Salı	Çarsamba	Perşembe	Cuma
1. Dilim		ENG300- (001) C306		BME314- (001) A307	BME314- (001) A307
2. Dilim		ENG300- (001) C306	BME314- (001) A307	BME314- (001) A307	
3. Dilim		ENG300- (001) C306	BME314- (001) A307		
4. Dilim					
5. Dilim					
6. Dilim	BME308- (001) A304	BME332- (001) A307	BME332- (001) A307		BME308- (001) A304
7. Dilim	BME308- (001) A304	BME332- (001) A307			
8. Dilim		BME332- (001) A307			
9. Dilim					
10. Dilim					

### Biyomedikal Mühendisliği 3. Sınıf 002. Şube DERS PROGRAMI

	Pazartesi	Salı	Çarşamba	Perşembe	Cuma
1. Dilim	BME314- (002) A307	BME314- (002) A307	BME314- (002) A307		
2. Dilim	BME314- (002) A307	BME314- (002) A307			
3. Dilim					
4. Dilim					
5. Dilim					
6. Dilim	BME332- (002) A308	ENG300- (002) C306		BME332- (002) A308	BME308- (002) A309
7. Dilim	BME332- (002) A308	ENG300- (002) C306	BME308- (002) A309		BME308- (002) A309
8. Dilim	BME332- (002) A308	ENG300- (002) C306			
9. Dilim					
10. Dilim					



## Biyomedikal Mühendisliği 4. Sınıf . Şube DERS PROGRAMI

	Pazartesi	Salı	Çarşamba	Perşembe	Cuma
1. Dilim	BME440- () A309	ENG400- (001) B406	BME440- () A309	BME445- () A309	BME445- () A309
2. Dilim	BME440- () A309	ENG400- (001) B406		BME445- () A309	BME445- () A309
3. Dilim		ENG400- (001) B406			
4. Dilim					
5. Dilim					
6. Dilim	BME414- () A309	BME402- () A309	BME414- () A309	BME402- () A309	
7. Dilim	BME414- () A309			BME402- () A309	
8. Dilim					
9. Dilim					
10. Dilim					

## Biyomedikal Mühendisliği 4. Sınıf . Şube DERS PROGRAMI

	Pazartesi	Salı	Çarşamba	Perşembe	Cuma
1. Dilim	BME440- () A309		BME440- () A309	BME445- () A309	BME445- () A309
2. Dilim	BME440- () A309			BME445- () A309	BME445- () A309
3. Dilim					
4. Dilim					
5. Dilim					
6. Dilim	BME414- () A309		BME414- () A309	BME402- () A309	
7. Dilim	BME414- () A309	ENG400- (002) B406		BME402- () A309	
8. Dilim		ENG400- (002) B406			
9. Dilim					
10. Dilim					

## Elektrik Elektronik Mühendisliği 1. Sınıf 001. Şube DERS PROGRAMI

	Pazartesi	Salı	Çarşamba	Perşembe	Cuma
1. Dilim	MAT101-0 A308	MAT151-(001) A308	ENG100-(001) A304	EDE100-0 A305	MAT101-0 A308
2. Dilim	MAT101-0 A308	MAT151-(001) A308	ENG100-(001) A304	EDE100-0 A305	
3. Dilim	BIL106-(001) C407	FIZ110-(001) B310	ENG100-(001) A304	FIZ110-(001) B310	BIL106-(001) C407
4. Dilim		FIZ110-(001) B310	MAT151-(001) A308		
5. Dilim					
6. Dilim		BIL102-(001) C307		MAT152-(001) C303	MAT105-(001) C407
7. Dilim	MAT105-(001) C407	BIL102-(001) C307	MAT152-(001) C303	FIZ104-(001) C407	MAT105-(001) C407
8. Dilim	BIL102-(001) C307	FIZ104-(001) C407	MAT152-(001) C303	FIZ104-(001) C407	
9. Dilim					
10. Dilim					

## Elektrik Elektronik Mühendisliği 1. Sınıf 002. Şube DERS PROGRAMI

	Pazartesi	Salı	Çarsamba	Perembe	Cuma
1. Dilim	MAT101-0 A308	BIL102-(002) C307	BIL102-(002) C307	EDE100-0 A305	MAT101-0 A308
2. Dilim	MAT101-0 A308	BIL102-(002) C307	BIL106-(002) C407	EDE100-0 A305	MAT151-(002) A311
3. Dilim		MAT151-(002) A311	FIZ110-(002) B310		MAT151-(002) A311
4. Dilim	FIZ110-(002) B310		FIZ110-(002) B310		BIL106-(002) C407
5. Dilim					
6. Dilim	MAT105-(002) C407	MAT105-(002) C407	ENG100-(002) A304	FIZ104-(002) C310	
7. Dilim	FIZ104-(002) C310	MAT105-(002) C407	ENG100-(002) A304	FIZ104-(002) C310	
8. Dilim			ENG100-(002) A304		
9. Dilim					
10. Dilim					

## Elektrik Elektronik Mühendisliği 2. Sınıf 001. Şube DERS PROGRAMI

	Pazartesi	Salı	Çarsamba	Perembe	Cuma
1. Dilim	EEM224- 0 A303	EEM202- (001) A303	ENG200- (001) C303	EEM224- 0 A303	TAR200- 0 C310
2. Dilim	EEM224- 0 A303	EEM202- (001) A303	ENG200- (001) C303	EEM202- (001) A303	TAR200- 0 C310
3. Dilim	EEM202- (001) A303	EEM202- (001) A303	ENG200- (001) C303	EEM214- (001) A303	TAR200- 0 C310
4. Dilim	EEM224- 0 A303	EEM202- (001) A303		EEM214- (001) A303	EEM214- (001) A303
5. Dilim					
6. Dilim	MAT210- (001) C309	EEM214 (LAB) - (001) LAB6	MAT210- (001) C306		EEM214 (LAB) - (001) LAB6
7. Dilim	MAT210- (001) C309	EEM214 (LAB) - (001) LAB6			
8. Dilim					
9. Dilim					
10. Dilim					

## Elektrik Elektronik Mühendisliği 2. Sınıf 002. Şube DERS PROGRAMI

	Pazartesi	Salı	Çarsamba	Perembe	Cuma
1. Dilim	EEM224- 0 A303	EEM202 (LAB) - (002) LAB6	EEM214- (002) A303	EEM224- 0 A303	TAR200- 0 C310
2. Dilim	EEM224- 0 A303	EEM202 (LAB) - (002) LAB6	EEM214- (002) A303	EEM214 (LAB) - (002) LAB5	TAR200- 0 C310
3. Dilim	EEM202 (LAB) - (002) LAB6	EEM214- (002) A304	EEM214 (LAB) - (002) LAB5	EEM214 (LAB) - (002) LAB5	TAR200- 0 C310
4. Dilim	EEM224- 0 A303				
5. Dilim					
6. Dilim	EEM202- (002) A303		ENG200- (002) B407	EEM202- (002) A303	
7. Dilim	EEM202- (002) A303	MAT210- (002) C309	ENG200- (002) B407		
8. Dilim	MAT210- (002) C309	MAT210- (002) C309	ENG200- (002) B407		
9. Dilim					
10. Dilim					

### Elektrik Elektronik Mühendisliği 3. Sınıf 001. Şube DERS PROGRAMI

	Pazartesi	Salı	Çarsamba	Perembe	Cuma
1. Dilim			ENG300- (001) C306		
2. Dilim			ENG300- (001) C306		
3. Dilim	EEM302- (001) A306	EEM322- (001) A306	ENG300- (001) C306	EEM322- (001) A306	EEM302- (001) A306
4. Dilim	EEM302- (001) A306	EEM322- (001) A306	EEM322- (001) A306		
5. Dilim					
6. Dilim	EEM312 (LAB) - (001) LAB5	EEM332- (001) A308	EEM312 (LAB) - (001) LAB5		EEM312- (001) A307
7. Dilim	EEM312 (LAB) - (001) LAB5	EEM332- (001) A308	EEM312- (001) A307	EEM332- (001) A308	EEM312- (001) A307
8. Dilim		EEM332- (001) A308		EEM332- (001) A308	
9. Dilim					
10. Dilim					

### Elektrik Elektronik Mühendisliği 3. Sınıf 002. Şube DERS PROGRAMI

	Pazartesi	Salı	Çarşamba	Perşembe	Cuma
1. Dilim	EEM332 (LAB) - (002) LAB5		EEM312- (002) A308	EEM312- (002) A308	EEM332 (LAB) - (002) LAB5
2. Dilim	EEM332 (LAB) - (002) LAB5		EEM312- (002) A308		EEM322- (002) A307
3. Dilim	EEM322- (002) A307	EEM322- (002) A307	EEM312- (002) A308	EEM332- (002) A307	EEM312- (002) A308
4. Dilim	EEM322- (002) A307			EEM332- (002) A307	EEM312- (002) A308
5. Dilim					
6. Dilim			ENG300- (002) C309		
7. Dilim			ENG300- (002) C309		
8. Dilim		EEM302- (002) A306	ENG300- (002) C309	EEM302- (002) A306	
9. Dilim		EEM302- (002) A306			
10. Dilim					



## Elektrik Elektronik Mühendisliği 4. Sınıf . Şube DERS PROGRAMI

	Pazartesi	Salı	Çarşamba	Perşembe	Cuma
1. Dilim	EEM442-0 A306	EEM474-0 A306	ENG400- (001) B406	EEM474-0 A306	EEM474-0 A306
2. Dilim	EEM442-0 A306	EEM474-0 A306	ENG400- (001) B406	EEM442-0 A306	EEM442-0 A306
3. Dilim			ENG400- (001) B406		
4. Dilim					
5. Dilim					
6. Dilim	EEM438-0 A306	EEM438-0 A306	EEM438-0 A306	EEM462-0 A306	EEM423-0 A306
7. Dilim	EEM438-0 A306	EEM462-0 A306	EEM423-0 A306	EEM462-0 A306	EEM423-0 A306
8. Dilim	EEM423-0 A306				
9. Dilim					
10. Dilim					

## Elektrik Elektronik Mühendisliği 4. Sınıf . Şube DERS PROGRAMI

	Pazartesi	Sali	Çarsamba	Persembe	Cuma
1. Dilim	EEM442-0 A306	EEM474-0 A306		EEM474-0 A306	EEM474-0 A306
2. Dilim	EEM442-0 A306	EEM474-0 A306		EEM442-0 A306	EEM442-0 A306
3. Dilim					
4. Dilim					
5. Dilim					
6. Dilim	EEM438-0 A306	EEM438-0 A306		EEM462-0 A306	EEM423-0 A306
7. Dilim	EEM438-0 A306	EEM462-0 A306		EEM462-0 A306	EEM423-0 A306
8. Dilim	EEM423-0 A306		ENG400- (002) C304		
9. Dilim					
10. Dilim					

## Endüstri Mühendisliği 1. Sınıf 001. Şube DERS PROGRAMI

	Pazartesi	Sali	Çarsamba	Persembe	Cuma
1. Dilim	FIZ110- (001) B304	MAT152- (001) C304	FIZ110- (001) B304	ENG100- (001) A304	EDE100- () A305
2. Dilim	FIZ110- (001) B304	MAT152- (001) C304	FIZ104- (001) B304	ENG100- (001) A304	EDE100- () A305
3. Dilim	FIZ104- (001) B304		MAT152- (001) C304	ENG100- (001) A304	
4. Dilim	FIZ104- (001) B304				
5. Dilim					
6. Dilim				BIL102- (001) C306	BIL102- (001) C306
7. Dilim		KIM110- (001) A309		BIL102- (001) C306	
8. Dilim	KIM110- (001) A309			KIM110- (001) A309	
9. Dilim	KIM110- (001) A309				
10. Dilim					

## Endüstri Mühendisliği 1. Sınıf 002. Şube DERS PROGRAMI

	Pazartesi	Sali	Çarsamba	Persembe	Cuma
1. Dilim	MAT152- (002) C304	FIZ104- (002) C407	MAT152- (002) C304	BIL106 (LAB) - (002) LAB2	EDE100- () A305
2. Dilim	MAT152- (002) C304	FIZ104- (002) C407	KIM110- (002) A309		EDE100- () A305
3. Dilim	KIM110 (LAB) - (002) LAB15		KIM110- (002) A309	FIZ110- (002) B311	KIM110- (002) A309
4. Dilim	FIZ104- (002) C407	FIZ110- (002) B311	BIL106- (002) C407	FIZ110- (002) B311	
5. Dilim					
6. Dilim				ENG100- (002) A304	
7. Dilim			BIL102- (002) C306	ENG100- (002) A304	BIL102- (002) C306
8. Dilim			BIL102- (002) C306	ENG100- (002) A304	
9. Dilim					
10. Dilim					

## Endüstri Mühendisliği 2. Sınıf . Şube DERS PROGRAMI

	Pazartesi	Salı	Çarsamba	Perembe	Cuma
1. Dilim	TAR200-0 C310	END202-0 B307	END202-0 B307	ENG200- (001) B407	SOS222- (001) A310
2. Dilim	TAR200-0 C310	END202-0 B307	END202-0 B307	ENG200- (001) B407	SOS222- (001) A310
3. Dilim	TAR200-0 C310			ENG200- (001) B407	
4. Dilim					
5. Dilim					
6. Dilim	END232 (LAB) - (001) LAB10			END232- (001) B303	END204- (001) B305
7. Dilim	END232 (LAB) - (001) LAB10			END232- (001) B303	END204- (001) B305
8. Dilim	END204- (001) B305			END204- (001) B305	
9. Dilim					
10. Dilim					

## Endüstri Mühendisliği 2. Sınıf . Şube DERS PROGRAMI

	Pazartesi	Salı	Çarsamba	Perembe	Cuma
1. Dilim	TAR200-0 C310	END202-0 B307	END202-0 B307	SOS222- (002) A310	
2. Dilim	TAR200-0 C310	END202-0 B307	END202-0 B307	SOS222- (002) A310	
3. Dilim	TAR200-0 C310				
4. Dilim					
5. Dilim					
6. Dilim	END204- (002) B303	END204- (002) B303	END204- (002) B303	ENG200- (002) B407	
7. Dilim	END204- (002) B303	END232- (002) B303	END232- (002) B303	ENG200- (002) B407	
8. Dilim		END232- (002) B303	END232- (002) B303	ENG200- (002) B407	
9. Dilim					
10. Dilim					

### Endüstri Mühendisliği 3. Sınıf . Şube DERS PROGRAMI

	Pazartesi	Sali	Çarsamba	Persembe	Cuma
1. Dilim		END304-0 B304	END308 (LAB) -0 LAB10	ENG300- (001) C309	END304- 0 B304
2. Dilim		END304-0 B304	END308 (LAB) -0 LAB10	ENG300- (001) C309	END304- 0 B304
3. Dilim		END308-0 B307		ENG300- (001) C309	END310- 0 B307
4. Dilim		END308-0 B307	END310- 0 B307		END310- 0 B307
5. Dilim					
6. Dilim				END306-0 B307	END306- 0 B307
7. Dilim		END306-0 B307		END306-0 B307	
8. Dilim	END312- (001) B304			END312- (001) B304	
9. Dilim	END312- (001) B304				
10. Dilim					

### Endüstri Mühendisliği 3. Sınıf . Şube DERS PROGRAMI

	Pazartesi	Salı	Çarşamba	Perşembe	Cuma
1. Dilim		END304- 0 B304	END308 (LAB) - 0 LAB10	END312- (002) B304	END304- 0 B304
2. Dilim		END304- 0 B304	END308 (LAB) - 0 LAB10	END312- (002) B304	END304- 0 B304
3. Dilim		END308- 0 B307	END312- (002) B304		END310- 0 B307
4. Dilim		END308- 0 B307	END310- 0 B307		END310- 0 B307
5. Dilim					
6. Dilim					END306- 0 B307
7. Dilim		END306- 0 B307			
8. Dilim				ENG300- (002) C309	
9. Dilim					
10. Dilim					



## Endüstri Mühendisliği 4. Sınıf 001. Şube DERS PROGRAMI

	Pazartesi	Sali	Çarsamba	Persembe	Cuma
1. Dilim	END450-0 B307			ENG400- (001) C304	END450-0 B307
2. Dilim	END415-0 B307			ENG400- (001) C304	END450-0 B307
3. Dilim	END415-0 B307	END409-0 B305	END415-0 B307	ENG400- (001) C304	
4. Dilim	END409-0 B305	END409-0 B305		END409-0 B305	
5. Dilim					
6. Dilim					
7. Dilim					
8. Dilim					
9. Dilim					
10. Dilim					

## Endüstri Mühendisliği 4. Sınıf 002. Şube DERS PROGRAMI

	Pazartesi	Sali	Çarsamba	Persembe	Cuma
1. Dilim	END450- () B307				END450- () B307
2. Dilim	END415- () B307				END450- () B307
3. Dilim	END415- () B307	END409- () B305	END415- () B307		
4. Dilim	END409- () B305	END409- () B305		END409- () B305	
5. Dilim					
6. Dilim				ENG400- (002) C304	
7. Dilim				ENG400- (002) C304	
8. Dilim				ENG400- (002) C304	
9. Dilim					
10. Dilim					

## Makina Mühendisliği 1. Sınıf 001. Şube DERS PROGRAMI

	Pazartesi	Salı	Çarsamba	Perembe	Cuma
1. Dilim	EDE100-0 A305	BIL106 (LAB) - (001) LAB?	BIL106- (001) C407	FIZ110- (001) B311	ENG100- (001) B311
2. Dilim	EDE100-0 A305		MAT152- (001) C304	FIZ110- (001) B311	ENG100- (001) B311
3. Dilim		MAT152- (001) C304		BIL102- (001) C306	ENG100- (001) B311
4. Dilim	BIL102- (001) C306	MAT152- (001) C304	FIZ110- (001) B311	BIL102- (001) C306	
5. Dilim					
6. Dilim	FIZ104 (LAB) - (001) LAB15	FIZ104- (001) B304		KIM110 (LAB) - (001) LAB19	MAK104- (001) B406
7. Dilim	MAT151- (001) A311	FIZ104- (001) B304	KIM110 (LAB) - (001) LAB19	KIM110 (LAB) - (001) LAB19	MAK104- (001) B406
8. Dilim	KIM110 (LAB) - (001) LAB19	MAK104 (LAB) - (001) LAB19	MAT151- (001) A311		
9. Dilim		MAK104 (LAB) - (001) LAB19	MAT151- (001) A311		
10. Dilim					

**Makina Mühendisliği 1. Sınıf 002. Şube DERS PROGRAMI**

	Pazartesi	Salı	Çarsamba	Perembe	Cuma
1. Dilim	EDE100-0 A305	MAK104 (LAB) - (002) LAB19	MAT151- (002) A311	MAT151- (002) A311	BIL102- (002) C307
2. Dilim	EDE100-0 A305	MAK104 (LAB) - (002) LAB19	BIL102- (002) C307	MAT151- (002) A311	BIL102- (002) C307
3. Dilim		FIZ110- (002) B304		FIZ110- (002) B304	
4. Dilim		FIZ110- (002) B304			
5. Dilim					
6. Dilim	FIZ104- (002) C310	FIZ104- (002) C310	BIL106 (LAB) - (002) LAB?	MAK104- (002) B406	ENG100- (002) B311
7. Dilim		FIZ104- (002) C310		MAK104- (002) B406	ENG100- (002) B311
8. Dilim		KIM110 (LAB) - (002) LAB10	KIM110 (LAB) - (002) LAB10	KIM110 (LAB) - (002) LAB10	ENG100- (002) B311
9. Dilim	BIL106- (002) C407	KIM110 (LAB) - (002) LAB10			
10. Dilim					

## Makina Mühendisligi 2. Sınıf 001. Şube DERS PROGRAMI

	Pazartesi	Salı	Çarsamba	Perembe	Cuma
1. Dilim	MAK208- (001) B303	TAR200- () C310	MAK202- (001) B303		ENG200- (001) B407
2. Dilim	MAK208- (001) B303	TAR200- () C310	MAK202- (001) B303		ENG200- (001) B407
3. Dilim	MAK208- (001) B303	TAR200- () C310	MAK208- (001) B303		ENG200- (001) B407
4. Dilim	MAK202- (001) B303		MAK202- (001) B303		
5. Dilim					
6. Dilim	MAK212- (001) B310		MAK212- (001) B310		MAK204- (001) B312
7. Dilim	MAK212- (001) B310		MAK206- (001) B307		MAK204- (001) B312
8. Dilim	MAK206- (001) B307	MAK204- (001) B312	MAK206- (001) B307	MAK206- (001) B307	
9. Dilim					
10. Dilim					

## Makina Mühendisliği 2. Sınıf 002. Şube DERS PROGRAMI

	Pazartesi	Salı	Çarşamba	Perşembe	Cuma
1. Dilim	MAK202- (002) B312	TAR200- 0 C310	MAK202- (002) B312		MAK208- (002) B305
2. Dilim	MAK202- (002) B312	TAR200- 0 C310			MAK208- (002) B305
3. Dilim		TAR200- 0 C310		MAK208- (002) B305	MAK208- (002) B305
4. Dilim		MAK202- (002) B312			
5. Dilim					
6. Dilim		MAK212- (002) B310	MAK206 (LAB) - (002) LAB19	MAK206- (002) B305	ENG200- (002) B407
7. Dilim	MAK206- (002) B305	MAK212- (002) B310	MAK212- (002) B310	MAK206- (002) B305	ENG200- (002) B407
8. Dilim	MAK204- (002) B312		MAK204- (002) B312		ENG200- (002) B407
9. Dilim	MAK204- (002) B312				
10. Dilim					

### Makina Mühendisliği 3. Sınıf 001. Şube DERS PROGRAMI

	Pazartesi	Sali	Çarsamba	Persembe	Cuma
1. Dilim			MAK306 (LAB) - 0 LAB19	MAK302 (LAB) - 0 LAB19	ENG300- (001) C309
2. Dilim					ENG300- (001) C309
3. Dilim			MAK303- 0 B305		ENG300- (001) C309
4. Dilim					
5. Dilim					
6. Dilim	MAK312- 0 B306	MAK308- 0 B306	MAK312- 0 B306	MAK308- 0 B306	MAK308- 0 B306
7. Dilim	MAK312- 0 B306	MAK308- 0 B306	MAK302- 0 B306	MAK306- 0 B306	MAK303- 0 B306
8. Dilim	MAK306- 0 B306	MAK302- 0 B306	MAK302- 0 B306	MAK306- 0 B306	MAK303- 0 B306
9. Dilim					
10. Dilim					

### Makina Mühendisliği 3. Sınıf 002. Şube DERS PROGRAMI

	Pazartesi	Salı	Çarşamba	Perşembe	Cuma
1. Dilim			MAK306 (LAB) - 0 LAB19	MAK302 (LAB) - 0 LAB19	
2. Dilim					
3. Dilim			MAK303- 0 B305		
4. Dilim					
5. Dilim					
6. Dilim	MAK312- 0 B306	MAK308- 0 B306	MAK312- 0 B306	MAK308- 0 B306	
7. Dilim	MAK312- 0 B306	MAK308- 0 B306	MAK302- 0 B306	MAK306- 0 B306	
8. Dilim	MAK306- 0 B306	MAK302- 0 B306	MAK302- 0 B306	MAK306- 0 B306	
9. Dilim					
10. Dilim					



## Makina Mühendisligi 4. Sınıf . Şube DERS PROGRAMI

	Pazartesi	Sali	Çarsamba	Perembe	Cuma
1. Dilim	MAK496-0 B305	MAK440-0 B305	MAK431-0 B305	MAK440-0 B305	ENG400- (001) C304
2. Dilim	MAK496-0 B305	MAK440-0 B305	MAK431-0 B305	MAK431-0 B305	ENG400- (001) C304
3. Dilim					ENG400- (001) C304
4. Dilim			MAK496-0 B305		
5. Dilim					
6. Dilim	MAK402-0 B305	MAK402-0 B305	MAK402-0 B305		
7. Dilim		MAK402-0 B305	MAK402-0 B305		
8. Dilim					
9. Dilim					
10. Dilim					

## Makina Mühendisligi 4. Sınıf . Şube DERS PROGRAMI

	Pazartesi	Sali	Çarsamba	Persembe	Cuma
1. Dilim	MAK496-0 B305	MAK440-0 B305	MAK431-0 B305	MAK440-0 B305	
2. Dilim	MAK496-0 B305	MAK440-0 B305	MAK431-0 B305	MAK431-0 B305	
3. Dilim					
4. Dilim			MAK496-0 B305		
5. Dilim					
6. Dilim	MAK402-0 B305	MAK402-0 B305	MAK402-0 B305		ENG400- (002) C304
7. Dilim		MAK402-0 B305	MAK402-0 B305		ENG400- (002) C304
8. Dilim					ENG400- (002) C304
9. Dilim					
10. Dilim					