

**PROGRAMLAMA TEMELLERİ DERSİ**

<b>MODÜL ADI</b>	<b>SÜRESİ</b>
• KODLAMAYA HAZIRLIK	40/32
• BASİT KODLAR	40/32
• KONTROL DEYİMLERİ	40/32
• METOTLAR	40/32

## DERS BİLGİ FORMU

<b>DERSİN ADI</b>	<b>PROGRAMLAMA TEMELLERİ</b>
<b>ALAN</b>	<b>BİLİŞİM TEKNOLOJİLERİ</b>
<b>MESLEK/DAL</b>	<b>ALAN ORTAK</b>
<b>DERSİN OKUTULACAĞI SINIF/YIL</b>	<b>10. SINIF</b>
<b>ÖNERİLEN SÜRE</b>	Haftalık 4 Ders Saati (Teknik Lise ve Anadolu Teknik Liselerde 3 Ders Saati)
<b>DERSİN AMACI</b>	Bu ders ile öğrenciye; programlama ile ilgili temel kavramları tanıyarak kodlama öncesi hazırlıkları yapabilme, basit kodlar yazabilme, kontrol deyimlerini kullanabilme ve alt programlarla çalışma yeterliklerinin kazandırılması amaçlanmaktadır.
<b>DERSİN TANIMI</b>	Programlamanın temeli olan kodlama öncesi hazırlık, basit kodlar, kontrol deyimleri ve alt programlarla ilgili konularda gerekli bilgilerin verildiği derstir.
<b>DERSİN ÖN KOŞULLARI</b>	Bu ders ön koşul yoktur.
<b>DERS İLE KAZANDIRILACAK YETERLİKLER</b>	Bu dersin sonunda; 1. Kodlama öncesi hazırlıkları yapmak 2. Basit kodlar yazmak 3. Kontrol deyimlerini kullanmak 4. Metotlarla çalışmak yeterlikleri kazandırılacaktır.
<b>DERSİN İÇERİĞİ</b>	1. Programlamanın yapı taşları 2. Algoritma ve akış diyagramı 3. Programlama yazılımı 4. Değişkenler ve sabitler 5. Operatörler 6. Karar kontrol deyimleri 7. Döngü deyimleri 8. Diziler 9. Metotlar 10. Hazır metotlar  İle ilgili konularını içermektedir
<b>YÖNTEM VE TEKNİKLER</b>	Bireysel öğretimi destekleyecek şekilde; Anlatım, göstererek yaptırma, problem çözme, soru-cevap, grup çalışması, beyin fırtınası, uygulama vb. yöntem ve teknikleri uygulanabilir.
<b>EĞİTİM ÖĞRETİM ORTAMI VE DONATIM</b>	<b>Ortam:</b> Bilişim teknolojileri laboratuvarı, işletme ortamı. <b>Donanım:</b> Projeksiyon, bilgisayar vb. sağlanmalıdır.
<b>ÖLÇME VE DEĞERLENDİRME</b>	Millî Eğitim Bakanlığı Orta Öğretim Kurumları Sınıf Geçme ve Sınav Yönetmeliği'ne uygun olarak modül ve ders sonunda ölçme araçları kullanılarak ölçme ve değerlendirme yapılacaktır.

<b>EĞİTİMCİLER</b>	<ol style="list-style-type: none"><li>1. Millî Eğitim Bakanlığına bağlı eğitim kurumlarına öğretmen olarak atanacaklardan atamalarına esas olan alanlar ile mezun oldukları yükseköğretim programları ve aylık karşılığı okutacakları derslere ilişkin çizelgeye uygun olanlar görev almalıdır.</li><li>2. Programın uygulanmasında gerektiğinde alanında sektör deneyimi olan usta öğretici, teknisyen ve meslek elemanlarından yararlanılabilir.</li></ol>
<b>İŞBİRLİĞİ YAPILACAK KURUM VE KURULUŞLAR</b>	Diğer alan öğretmenleri, üniversiteler, sosyal ortaklar, sivil toplum kuruluşları, çevrede bulunan işletmeler, özel, kamu kurum ve kuruluşlarıdır.

## MODÜL BİLGİ SAYFASI

<b>ALAN</b>	<b>: BİLİŞİM TEKNOLOJİLERİ</b>
<b>MESLEK/DAL</b>	<b>: ALAN ORTAK</b>
<b>DERS</b>	<b>: PROGRAMLAMA TEMELLERİ</b>
<b>MODÜL</b>	<b>: KODLAMAYA HAZIRLIK</b>
<b>KODU</b>	<b>: 482BK0123</b>
<b>SÜRE</b>	<b>: 40/32</b>
<b>ÖN KOŞUL</b>	: Bu modül için ön koşul yoktur.
<b>AÇIKLAMA</b>	: Uygulamalı olarak laboratuvar ortamında işlenmelidir.
<b>ÖĞRETİM YÖNTEM VE TEKNİKLERİ</b>	: Öğretim yöntem ve teknikleri olarak, anlatım, göstererek yaptırma, problem çözme, soru-cevap, grup çalışması, uygulamalı çalışma, araştırma ve bireysel öğretim yöntem ve teknikleri uygulanabilir.
<b>GENEL AMAÇ</b>	: Öğrenci bu modül ile gerekli ortam sağlandığında; programlamanın kodlama öncesi hazırlıklarını yapabilecektir.
<b>AMAÇLAR</b>	: <ol style="list-style-type: none"><li>1. Program yazmaya hazırlık yapabilecektir.</li><li>2. Algoritma ve akış diyagramı hazırlayabilecektir.</li><li>3. Programlama dili yazılımı ile çalışabilecektir.</li></ol>

## İÇERİK

### 1. PROGRAMLAMANIN YAPI TAŞLARI

- 1.1. Bilgisayarın Çalışma Mantığı
  - 1.1.1. Makine Dili
  - 1.1.2. İkili Sayı Sistemi
  - 1.1.3. Derleyici
- 1.2. Yazılım
  - 1.2.1. Bir Yazılımda Olması Gereken Temel Özellikler
  - 1.2.2. Yazılım Çeşitleri
- 1.3. Programlama Araçları
  - 1.3.1. Programlama Dili
  - 1.3.2. Programlama Dili Çeşitleri

### 2. ALGORİTMA VE AKIŞ DİYAGRAMI

- 2.1. Algoritma Yazım Aşamaları
  - 2.1.1. Problemi Tanımlama
  - 2.1.2. Problemi Geliştirme

- 2.1.3. Sisteme Uyumluluğunu Tespit Etme (Girdi-Çıktı Belirleme)
- 2.1.4. Çözümü Kâğıt Üzerinde Gösterme (Prototip Oluşturma)
- 2.1.5. Çözümü Deneme (Trace)
- 2.1.6. Çözümü Geliştirme
- 2.1.7. Oluşabilecek Hatalar
- 2.2. Akış Diyagramı
- 2.2.1. Akış Diyagramı Şekilleri

### 3. PROGRAMLAMA YAZILIMI

- 3.1. Programlama Yazılımının Arayüzü
- 3.1.1. Başlangıç Sayfası
- 3.1.2. Solution Explorer Paneli

### ÖLÇME VE DEĞERLENDİRME:

Öğrenci, aşağıdaki performans ölçütlerini yerine getirecektir.

#### A. PROGRAMLAMANIN YAPI TAŞLARI

1. Bilgisayarın çalışma mantığını yazılı /sözlü / görsel olarak açıklar.
2. Yazılımda olması gereken temel özelliklere göre yazılım çeşitlerini yazılı /sözlü / görsel olarak açıklar.

#### B. ALGORİTMA VE AKIŞ DİYAGRAMI

1. Algoritma yazım aşamalarını yazılı /sözlü / görsel olarak açıklar.
2. Bir problemin çözümünü algoritma kullanarak yapar.
3. Akış diyagramı şekillerini yazılı /sözlü / görsel olarak açıklar.
4. Bir problemin çözümünü akış şemaları kullanarak yapar.

#### C. PROGRAMLAMA YAZILIMI

1. Programlama yazılımını başlatır.
2. Programlama yazılımını kullanarak bir konsol uygulaması hazırlar.
3. Solution Explorer panelini kullanarak dosyaları düzenler.

Programda kazandırılması amaçlanan “Kodlama öncesi hazırlık yapmak” yeterliğine ait performans kriterlerinin değerlendirilmesinde aşağıdaki ölçme ve değerlendirme yöntemlerinden;

- Birey/öğrenciden beklenen davranışın özelliklerine ilişkin detaylı bilgileri ve öğrenci performansının eksikliklerinin belirlenmesi amacıyla uygulamaya yönelik kontrol listeleri tekniği,
  - Birey/Öğrencilere öğretilen bilgi ve becerileri kazanıp kazanmadığını, bu bilgi ve becerileri günlük yaşamla da ilişkilendirebildiklerini ortaya koyan proje ödevleri tekniği,
  - Birey/Öğrencilere çok sayıda davranış ve becerinin kısa sürede kazandırılması, daha çok bilgi, zihinsel beceri ve yeteneğin ölçülmesi için çoktan seçmeli, klasik sınav, vb.
- teknikler kullanılabilir.

## İŞLEM ANALİZ FORMU

<b>MESLEĞİN ADI</b>	ALAN ORTAK	<b>İŞ</b>	Temel Programlama işlemleri yapmak
<b>İŞLEM NUMARASI</b>	1	<b>İŞLEMİN ADI</b>	Program yazmaya hazırlık yapmak
<b>YETERLİK</b>	Kodlama öncesi hazırlık yapmak		
<b>ORTAM (Araç Gereç, Ekipman ve Koşullar)</b>	Bilgisayar, Programlama dili yazılımı		
<b>İŞLEMİN STANDARDI</b>	Yazılacak programa uygun programlama dilini seçmek		
<b>İŞLEM BASAMAKLARI</b>	<b>BİLGİ</b>	<b>BECERİ</b>	<b>TAVİR</b>
<ol style="list-style-type: none"> <li>1. Programlama araçlarını kullanmak</li> <li>2. Yazılımda olması gereken temel nitelikleri yazılım içerisinde kullanmak</li> <li>3. Yazılacak koda göre yazılım dilini seçmek</li> </ol>	<ol style="list-style-type: none"> <li>1. <b>PROGRAMLAMANIN YAPI TAŞLARI</b> <ol style="list-style-type: none"> <li>1.1. Bilgisayarın Çalışma Mantığı                             <ol style="list-style-type: none"> <li>1.1.1. Makine Dili</li> <li>1.1.2. İkili Sayı Sistemi</li> <li>1.1.3. Derleyici</li> </ol> </li> <li>1.2. Yazılım                             <ol style="list-style-type: none"> <li>1.2.1. Bir Yazılımda Olması Gereken Temel Özellikler</li> <li>1.2.2. Yazılım Çeşitleri</li> </ol> </li> <li>1.3. Programlama Araçları                             <ol style="list-style-type: none"> <li>1.3.1. Programlama Dili</li> <li>1.3.2. Programlama Dili Çeşitleri</li> </ol> </li> </ol> </li> </ol>	<ol style="list-style-type: none"> <li>1. Yazılımda olması gereken temel özellikleri kullanmak</li> <li>2. Programlama dili yazılımını seçmek</li> </ol>	<ol style="list-style-type: none"> <li>1. Lisanslı yazılım kullanmak</li> <li>2. Programlama dili çalışma mantığının farkında olmak</li> </ol>
<b>SÜRE:</b> İşlemin yapılma süresi: 30 dakika		<b>İşlemi öğrenme süresi:</b> 8 ders saati	
<b>NOT:</b> Bilgi sayfalarında verilen konuların öğrenci tarafından anlaşıldığını test eden ve işlem basamakları ile uyuşan bir uygulama faaliyeti oluşturulmasına dikkat ediniz.			

**İŞLEM ANALİZ FORMU**

<b>MESLEĞİN ADI</b>	ALAN ORTAK	<b>İŞ</b>	Temel Programlama işlemleri yapmak
<b>İŞLEM NUMARASI</b>	2	<b>İŞLEMİN ADI</b>	Algoritma ve akış diyagramı hazırlamak
<b>YETERLİK</b>	Kodlama öncesi hazırlık yapmak		
<b>ORTAM (Araç Gereç, Ekipman ve Koşullar)</b>	Bilgisayar, Programlama dili yazılımı		
<b>İŞLEMİN STANDARDI</b>	Özellikleri verilen işin algoritmasının ve akış diyagramının hazırlanması		
<b>İŞLEM BASAMAKLARI</b>	<b>BİLGİ</b>	<b>BECERİ</b>	<b>TAVİR</b>
<ol style="list-style-type: none"><li>1. Problem tanımını yapmak</li><li>2. Çözüm yolları üretmek</li><li>3. Oluşabilecek mantıksal hataları gidermek</li><li>4. Akış diyagramını şekillerini kullanmak</li><li>5. Akış diyagramını algoritma ile beraber kullanmak</li></ol>	<ol style="list-style-type: none"><li>2. <b>ALGORİTMA VE AKIŞ DİYAGRAMI</b><ol style="list-style-type: none"><li>2.1. Algoritma Yazım Aşamaları<ol style="list-style-type: none"><li>2.1.1. Problemi Tanımlama</li><li>2.1.2. Problemi Geliştirme</li><li>2.1.3. Sisteme Uyumluluğunu Tespit Etme (Girdi-Çıktı Belirleme)</li><li>2.1.4. Çözümü Kâğıt Üzerinde Gösterme (Prototip Oluşturma)</li><li>2.1.5. Çözümü Deneme (Trace)</li><li>2.1.6. Çözümü Geliştirme</li><li>2.1.7. Oluşabilecek Hatalar</li></ol></li><li>2.2. Akış Diyagramı</li><li>2.3. Akış Diyagramı Şekilleri</li></ol></li></ol>	<ol style="list-style-type: none"><li>1. Algoritma hazırlamak</li><li>2. Akış diyagramı hazırlamak</li><li>3. Problem çözümünde algoritma ve akış diyagramını beraber kullanmak</li></ol>	<ol style="list-style-type: none"><li>1. Oluşabilecek tüm olasılıkları değerlendirmek</li><li>2. En uygun yöntemle sonuca ulaşmak</li><li>3. Algoritmanın okunurluğuna dikkat etmek</li><li>4. Akış diyagramı şekillerinin ifadelerinin farkında olmak</li></ol>
<b>SÜRE:</b> İşlemin yapılma süresi: 30 dakika	İşlemi öğrenme süresi: 16 ders saati		
<b>NOT:</b> Bilgi sayfalarında verilen konuların öğrenci tarafından anlaşıldığını test eden ve işlem basamakları ile uyuşan bir uygulama faaliyeti oluşturulmasına dikkat ediniz.			

**İŞLEM ANALİZ FORMU**

<b>MESLEĞİN ADI</b>	ALAN ORTAK	<b>İŞ</b>	Temel Programlama işlemleri yapmak
<b>İŞLEM NUMARASI</b>	3	<b>İŞLEMİN ADI</b>	Programlama dili yazılımı ile çalışmak
<b>YETERLİK</b>	Kodlama öncesi hazırlık yapmak		
<b>ORTAM (Araç Gereç, Ekipman ve Koşullar)</b>	Bilgisayar, Programlama dili yazılımı		
<b>İŞLEMİN STANDARDI</b>	Programlama dili yazılımını etkili ve verimli şekilde kullanmak		
<b>İŞLEM BASAMAKLARI</b>	<b>BİLGİ</b>	<b>BECERİ</b>	<b>TAVİR</b>
<ol style="list-style-type: none"><li>1. Bilgisayara kurulmuş olan programlama dili yazılımını başlatmak</li><li>2. Başlangıç sayfasını kullanmak</li><li>3. Yeni bir konsol uygulaması başlatmak</li><li>4. Konsol uygulamasını kaydetmek</li><li>5. Kaydedilmiş konsol uygulamasını açmak</li><li>6. Konsol uygulamasını çalıştırmak</li><li>7. Solution Explorer panelini kullanmak</li></ol>	<ol style="list-style-type: none"><li>3. <b>PROGRAMLAMA YAZILIMI</b><ol style="list-style-type: none"><li>3.1. Programlama Yazılımının Arayüzü<ol style="list-style-type: none"><li>3.1.1. Başlangıç Sayfası</li><li>3.1.2. Solution Explorer Paneli</li></ol></li></ol></li></ol>	<ol style="list-style-type: none"><li>1. Programlama dili yazılımını işletim sisteminden başlatmak</li><li>2. Başlangıç sayfasını etkin şekilde kullanmak</li><li>3. Konsol uygulamaları için temel işlemleri yapmak</li><li>4. Solution Explorer panelini etkin bir şekilde kullanmak</li></ol>	<ol style="list-style-type: none"><li>1. Oluşabilecek tüm olasılıkları değerlendirmek</li><li>2. Konsol uygulaması işlemlerinde dikkat etmek</li></ol>
<b>SÜRE:</b> İşlemin yapılma süresi: 30 dakika	İşlemi öğrenme süresi: 8 ders saati		
<b>NOT:</b> Bilgi sayfalarında verilen konuların öğrenci tarafından anlaşıldığını test eden ve işlem basamakları ile uyuşan bir uygulama faaliyeti oluşturulmasına dikkat ediniz. Programlama dili yazılımında konsol uygulamasını kullanmaya özen gösteriniz.			



## MODÜL BİLGİ SAYFASI

<b>ALAN</b>	: BİLİŞİM TEKNOLOJİLERİ
<b>MESLEK/DAL</b>	: ALAN ORTAK
<b>DERS</b>	: PROGRAMLAMA TEMELLERİ
<b>MODÜL</b>	: BASİT KODLAR
<b>KODU</b>	: 482BK0122
<b>SÜRE</b>	: 40/32
<b>ÖN KOŞUL</b>	: “Kodlamaya Hazırlık” modülünü tamamlamış olmak.
<b>AÇIKLAMA</b>	: Uygulamalı olarak laboratuvar ortamında işlenmelidir.
<b>ÖĞRETİM YÖNTEM VE TEKNİKLERİ</b>	: Öğretim yöntem ve teknikleri olarak, anlatım, göstererek yaptırma, problem çözme, soru-cevap, grup çalışması, uygulamalı çalışma, araştırma ve bireysel öğretim yöntem ve teknikleri uygulanabilir.
<b>GENEL AMAÇ</b>	: Öğrenci bu modül ile gerekli ortam sağlandığında; temel programlamada basit kodlar yazabilecektir.
<b>AMAÇLAR</b>	: <ol style="list-style-type: none"><li>1. Değişken ve sabit kullanabilecektir.</li><li>2. Operatörleri kullanabilecektir.</li></ol>

## İÇERİK

### 1. DEĞİŞKENLER VE SABİTLER

- 1.1. Değişkenler
- 1.2. Değişkenleri İsimlendirme Kuralları
- 1.3. Veri tipleri
- 1.4. Sabitler
- 1.5. Atama İşlemi
- 1.6. Çıkış İşlemleri
  - 1.6.1. Bir metin ifadesini ekrana yazdırma
  - 1.6.2. İlk değer atanmış değişken değerini ekrana yazdırma
  - 1.6.3. Formatlı çıkış işlemleri
- 1.7. Giriş İşlemleri
  - 1.7.1. Klavyeden değişkene değer atama
- 1.8. Giriş-çıkış işlemleri hata mesajları
- 1.9. Açıklama Satırları

### 2. OPERATÖRLER

- 2.1. Aritmetiksel Operatörler
  - 2.1.1. Dört İşlem
  - 2.1.2. Mod Alma

- 2.2. İlişkisel Operatörler
- 2.3. Mantıksal Operatörler
- 2.4. İşlem Önceliği

## **ÖLÇME VE DEĞERLENDİRME:**

Öğrenci, aşağıdaki performans ölçütlerini yerine getirecektir.

### **A. DEĞİŞKENLER VE SABİTLER**

1. Değişkeni yazılı /sözlü / görsel olarak açıklar.
2. İsimlendirme kuralına uyarak ihtiyaca uygun değişkeni tanımlar.
3. Sabitleri yazılı /sözlü / görsel olarak açıklar.
4. Tanımlanan değişkene veri tipine uygun değeri atar.
5. Hata mesajları doğrultusunda gerekli düzenlemeleri yapar.

### **B. OPERATÖRLER**

1. Operatörleri yazılı /sözlü / görsel olarak açıklar.
2. Yapılacak işleme uygun operatörü doğru yerde doğru şekilde kullanır.

## İŞLEM ANALİZ FORMU

<b>MESLEĞİN ADI</b>	ALAN ORTAK	<b>İŞ</b>	Temel Programlama işlemleri yapmak
<b>İŞLEM NUMARASI</b>	1	<b>İŞLEMİN ADI</b>	Değişken ve sabit kullanmak
<b>YETERLİK</b>	Basit kodlar yazmak		
<b>ORTAM (Araç Gereç, Ekipman ve Koşullar)</b>		Bilgisayar, Programlama dili yazılımı	
<b>İŞLEMİN STANDARDI</b>		Program için uygun değişken tiplerini seçip, sabitleri kullanmak, giriş-çıkış işlemlerini hatasız yapmak	
<b>İŞLEM BASAMAKLARI</b>	<b>BİLGİ</b>	<b>BECERİ</b>	<b>TAVİR</b>
<ol style="list-style-type: none"> <li>1. Değişkenleri kullanmak</li> <li>2. Veri tiplerini uygulama çeşidine uygun olarak kullanmak</li> <li>3. Sabitleri kullanmak</li> <li>4. Değişken ve sabit kullanımı uygulamalarından önce algoritma ve akış diyagramını hazırlamak</li> <li>5. Çıkış işlemi yapmak</li> <li>6. Giriş işlemi yapmak</li> <li>7. Basit yapıda atama işlemi yapmak</li> <li>8. Kod parçaları içerisinde açıklama satırlarını kullanmak</li> <li>9. Hata kontrolü yapmak</li> </ol>	<ol style="list-style-type: none"> <li>1. <b>DEĞİŞKENLER VE SABİTLER</b> <ol style="list-style-type: none"> <li>1.1. Değişkenler</li> <li>1.2. Değişkenleri İsimlendirme Kuralları</li> <li>1.3. Veri tipleri</li> <li>1.4. Sabitler</li> <li>1.5. Atama İşlemi</li> <li>1.6. Çıkış İşlemleri                             <ol style="list-style-type: none"> <li>1.6.1. Bir metin ifadesini ekrana yazdırma</li> <li>1.6.2. İlk değer atanmış değişken değerini ekrana yazdırma</li> <li>1.6.3. Formatlı çıkış işlemleri</li> </ol> </li> <li>1.7. Giriş İşlemleri                             <ol style="list-style-type: none"> <li>1.7.1. Klavyeden değişkene değer atama</li> </ol> </li> <li>1.8. Giriş-çıkış işlemleri hata mesajları</li> <li>1.9. Açıklama Satırları</li> </ol> </li> </ol>	<ol style="list-style-type: none"> <li>1. Değişken tanımlamak</li> <li>2. Değişkene uygun veri tipini seçmek</li> <li>3. Sabit tanımlamak</li> <li>4. Atama işlemi mantığını çözmek</li> <li>5. Çıkış işlemlerini formatlı şekilde yapmak</li> <li>6. İşlem sonunda değişkene değer aktarmak</li> <li>7. Açıklama satırlarını kod içerisinde uygun yere yazmak</li> </ol>	<ol style="list-style-type: none"> <li>1. Oluşabilecek tüm olasılıkları değerlendirmek</li> <li>2. Konsol uygulaması işlemlerinde değişken kullanımına dikkat etmek</li> <li>3. Değişken isimlendirme kurallarına dikkat etmek</li> <li>4. Kod yazım kurallarına dikkat etmek</li> <li>5. Yazılan kod parçasının okunurluğuna dikkat etmek</li> </ol>
<b>SÜRE:</b> İşlemin yapılma süresi: 1 saat		<b>İşlemi öğrenme süresi:</b> 16 ders saati	
<b>NOT:</b> Bilgi sayfalarında verilen konuların öğrenci tarafından anlaşıldığını test eden ve işlem basamakları ile uyuşan bir uygulama faaliyeti oluşturulmasına dikkat ediniz. Programlama dili yazılımında konsol uygulamasını kullanmaya özen gösteriniz.			

İŞLEM ANALİZ FORMU			
MESLEĞİN ADI	ALAN ORTAK	İŞ	Temel Programlama işlemleri yapmak
İŞLEM NUMARASI	2	İŞLEMİN ADI	Operatörleri kullanmak
YETERLİK	Basit kodlar yazmak		
ORTAM (Araç Gereç, Ekipman ve Koşullar)		Bilgisayar, Programlama dili yazılımı	
İŞLEMİN STANDARDI		Matematiksel ifadeleri işlem önceliklerini kullanarak yapmak	
İŞLEM BASAMAKLARI	BİLGİ	BECERİ	TAVİR
<ol style="list-style-type: none"> <li>1. Değişkene ya da sabite uygulanacak işlemi belirlemek</li> <li>2. İşleme uygun operatörü belirlemek</li> <li>3. İşlem öncelik sırasını belirlemek</li> <li>4. Hata kontrolü yapmak</li> </ol>	<ol style="list-style-type: none"> <li>2. OPERATÖRLER               <ol style="list-style-type: none"> <li>2.1. Aritmetiksel Operatörler                   <ol style="list-style-type: none"> <li>2.1.1. Dört İşlem</li> <li>2.1.2. Mod Alma</li> </ol> </li> <li>2.2. İlişkisel Operatörler</li> <li>2.3. Mantıksal Operatörler</li> <li>2.4. İşlem Önceliği</li> </ol> </li> </ol>	<ol style="list-style-type: none"> <li>1. İşleme uygun operatörleri kullanmak</li> <li>2. İşlem önceliklerini belirlemek</li> <li>3. Hatayı fark edip düzeltmek</li> </ol>	<ol style="list-style-type: none"> <li>1. Performansı artıracak operatörü tercih etmek</li> <li>2. İşlem önceliklerini dikkate almak</li> <li>3. Gereksiz ayrıntıdan kaçınmak</li> <li>4. Kod yazım kurallarına dikkat etmek</li> <li>5. Kodların okunurluğuna dikkat etmek</li> </ol>
<b>SÜRE:</b> İşlemin yapılma süresi: 10 dakika		<b>SÜRE:</b> İşlemi öğrenme süresi: 16 ders saati	
<b>NOT:</b> Bilgi sayfalarında verilen konuların öğrenci tarafından anlaşıldığını test eden ve işlem basamakları ile uyuşan bir uygulama faaliyeti oluşturulmasına dikkat ediniz. Programlama dili yazılımında konsol uygulamasını kullanmaya özen gösteriniz.			

## MODÜL BİLGİ SAYFASI

<b>ALAN</b>	: BİLİŞİM TEKNOLOJİLERİ
<b>MESLEK/DAL</b>	: ALAN ORTAK
<b>DERS</b>	: PROGRAMLAMA TEMELLERİ
<b>MODÜL</b>	: KONTROL DEYİMERİ
<b>KODU</b>	: 482BK0124
<b>SÜRE</b>	: 40/32
<b>ÖN KOŞUL</b>	: "Basit Kodlar" modülünü tamamlamış olmak.
<b>AÇIKLAMA</b>	: Uygulamalı olarak laboratuvar ortamında işlenmelidir.
<b>ÖĞRETİM YÖNTEM VE TEKNİKLERİ</b>	: Öğretim yöntem ve teknikleri olarak, anlatım, göstererek yaptırma, problem çözme, soru-cevap, grup çalışması, uygulamalı çalışma, araştırma ve bireysel öğretim yöntem ve teknikleri uygulanabilir.
<b>GENEL AMAÇ</b>	: Öğrenci bu modül ile gerekli ortam sağlandığında; temel programlama işlemlerinde basit kodlar yazabilecektir.
<b>AMAÇLAR</b>	: <ol style="list-style-type: none"><li>1. Karar kontrol deyimlerini kullanabilecektir.</li><li>2. Döngü kontrollerini kullanabilecektir.</li><li>3. Dizilerle çalışabilecektir.</li></ol>

## İÇERİK

### 1. KARAR KONTROL DEYİMLERİ

- 1.1. If İfadesi
- 1.2. İç-İçe If İfadesi
- 1.3. Switch-Case Deyimi
  - 1.3.1. Break Komutu

### 2. DÖNGÜ DEYİMLERİ

- 2.1. Döngü Çeşitleri
  - 2.1.1. For Next Döngüsü
  - 2.1.2. While-Until Döngüsü
  - 2.1.3. Do Loop Döngüsü
  - 2.1.4. Foreach Döngüsü
- 2.2. İç-İçe Döngüler

### 3. DİZİLER

- 3.1. Dizi Oluşturma
- 3.2. Diziye Değer Girme
- 3.3. Diziye Yazdırma
- 3.4. Dizilerde Arama
- 3.5. Dizilerde Sıralama
- 3.6. Dinamik Diziler

## **ÖLÇME VE DEĞERLENDİRME:**

Öğrenci, aşağıdaki performans ölçütlerini yerine getirecektir.

### **A. KARAR KONTROL DEYİMLERİ**

1. Karar kontrol deyimlerini yazılı /sözlü / görsel olarak açıklar.
2. İşleme uygun karar kontrol yapısını doğru bir şekilde kullanır.

### **B. DÖNGÜ DEYİMLERİ**

1. Döngü deyimlerini yazılı /sözlü / görsel olarak açıklar.
2. İşleme uygun döngü yapısını doğru bir şekilde kullanır.

### **C. DİZİLER**

1. Biz dizi tanımlayarak doğru bir şekilde değer ataması yapar.
2. Diziye atanmış değerleri isteğe uygun bir şekilde ekrana yazdırır.
3. İstenilen ifadeyi dizi değerleri arasında arama işlemini gerçekleştirir.
4. Dizi değerlerini ihtiyaca uygun şekilde sıralamasını yapar.

İŞLEM ANALİZ FORMU			
MESLEĞİN ADI	ALAN ORTAK	İŞ	Temel Programlama işlemleri yapmak
İŞLEM NUMARASI	1	İŞLEMİN ADI	Karar kontrol deyimlerini kullanmak
YETERLİK	Kontrol deyimlerini kullanmak		
ORTAM (Araç Gereç, Ekipman ve Koşullar)		Bilgisayar, Programlama dili yazılımı	
İŞLEMİN STANDARTI		İşlem için en uygun kontrol deyimlerini kullanmak.	
İŞLEM BASAMAKLARI	BİLGİ	BECERİ	TAVİR
<ol style="list-style-type: none"> <li>1. Problemin algoritmasını oluşturmak</li> <li>2. Akış diyagramını çizmek</li> <li>3. Karar kontrol deyimini belirlemek</li> <li>4. Karar kontrol deyiminin şart kısmını belirlemek</li> <li>5. Karar kontrol deyim bloğunu oluşturmak</li> <li>6. Hata kontrolünü yapmak</li> </ol>	<ol style="list-style-type: none"> <li>1. <b>KARAR KONTROL DEYİMLERİ</b> <ol style="list-style-type: none"> <li>1.1. If İfadesi</li> <li>1.2. İç-İçe If İfadesi</li> <li>1.3. Switch-Case Deyimi</li> <li>1.4. Break Komutu</li> </ol> </li> </ol>	<ol style="list-style-type: none"> <li>1. Yapılacak işe uygun karar kontrol deyimlerini kullanmak</li> <li>2. Kodların kontrolünü yapıp oluşabilecek hataları gidermek</li> </ol>	<ol style="list-style-type: none"> <li>1. Okunabilir kodlar kullanmak</li> <li>2. Performansı düşürmeyecek kod kullanmaya özen göstermek</li> <li>3. Detayları dikkate almak</li> <li>4. Kodların okunurluğuna dikkat etmek</li> <li>5. Problem çözümünde algoritma ve akış diyagramı kullanmak</li> </ol>
<b>SÜRE:</b> İşlemin yapılma süresi: 1 saat		İşlemi öğrenme süresi: 6 ders saati	
<b>NOT:</b> Bilgi sayfalarında verilen konuların öğrenci tarafından anlaşıldığını test eden ve işlem basamakları ile uyuşan bir uygulama faaliyeti oluşturulmasına dikkat ediniz. Programlama dili yazılımında konsol uygulamasını kullanmaya özen gösteriniz.			

İŞLEM ANALİZ FORMU			
MESLEĞİN ADI	ALAN ORTAK	İŞ	Temel Programlama işlemleri yapmak
İŞLEM NUMARASI	2	İŞLEMİN ADI	Döngü kontrollerini kullanmak
YETERLİK	Kontrol deyimlerini kullanmak		
ORTAM (Araç Gereç, Ekipman ve Koşullar)		Bilgisayar, Programlama dili yazılımı	
İŞLEMİN STANDARTI		İşlem için en uygun döngü kontrollerini kullanmak	
İŞLEM BASAMAKLARI	BİLGİ	BECERİ	TAVİR
<ol style="list-style-type: none"> <li>1. Problemin algoritmasını oluşturmak</li> <li>2. Akış diyagramını çizmek</li> <li>3. Döngü Kontrol Deyimini belirlemek</li> <li>4. Döngü Kontrol Deyiminin şart kısmını belirlemek</li> <li>5. Döngü Kontrol Deyim bloğunu belirlemek</li> <li>6. İç-içe döngüleri kullanmak</li> <li>7. Döngü Kontrol Deyimlerinde hata kontrolünü yapmak</li> </ol>	<ol style="list-style-type: none"> <li>2. <b>DÖNGÜ DEYİMLERİ</b> <ol style="list-style-type: none"> <li>2.1. Döngü Çeşitleri           <ol style="list-style-type: none"> <li>2.1.1. For Next Döngüsü</li> <li>2.1.2. While-Until Döngüsü</li> <li>2.1.3. Do Loop Döngüsü</li> <li>2.1.4. Foreach Döngüsü</li> </ol> </li> <li>2.2. İç-içe Döngüler</li> </ol> </li> </ol>	<ol style="list-style-type: none"> <li>1. Uygun döngü kontrol deyimini kullanmak</li> <li>2. Döngü bloğunu oluşturmak</li> <li>3. Döngü işlemi sonucu oluşan hataları gidermek</li> </ol>	<ol style="list-style-type: none"> <li>1. Okunabilir kodlar kullanmak</li> <li>2. Performansı düşürmeyecek kod kullanmaya özen göstermek</li> <li>3. Detayları dikkate almak</li> <li>4. Döngü oluştururken kurallara uymak</li> <li>5. Kodların okunurluğuna dikkat etmek</li> <li>6. Problem çözümünde algoritma ve akış diyagramı kullanmak</li> </ol>
<b>SÜRE:</b> İşlemin yapılma süresi: 1 saat		İşlemi öğrenme süresi: 16 ders saati	
<b>NOT:</b> Bilgi sayfalarında verilen konuların öğrenci tarafından anlaşıldığını test eden ve işlem basamakları ile uyuşan bir uygulama faaliyeti oluşturulmasına dikkat ediniz. Programlama dili yazılımında konsol uygulamasını kullanmaya özen gösteriniz.			



İŞLEM ANALİZ FORMU			
MESLEĞİN ADI	ALAN ORTAK	İŞ	Temel Programlama işlemleri yapmak
İŞLEM NUMARASI	3	İŞLEMİN ADI	Dizilerle çalışmak
YETERLİK	Kontrol deyimlerini kullanmak		
ORTAM (Araç Gereç, Ekipman ve Koşullar)		Bilgisayar, Programlama dili yazılımı	
İŞLEMİN STANDARDI		Dizi işlemlerinin düzenli ve verimli şekilde kullanmak	
İŞLEM BASAMAKLARI	BİLGİ	BECERİ	TAVİR
<ol style="list-style-type: none"> <li>1. Problemin algoritmasını oluşturmak</li> <li>2. Akış diyagramını çizmek</li> <li>3. Bir dizi oluşturmak</li> <li>4. Diziye değer girmek</li> <li>5. Girilen değerleri ekrana yazdırmak</li> <li>6. Girilen değerleri istenilen şarta göre sıralamak</li> <li>7. Dizi içerisinde istenilen değeri aramak</li> <li>8. Oluşabilecek hataları gidermek</li> <li>9. Program çıktısını ekranda göstermek</li> </ol>	<ol style="list-style-type: none"> <li>3. DİZİLER</li> <li>3.1. Dizi Oluşturma</li> <li>3.2. Diziye Değer Girme</li> <li>3.3. Diziyi Yazdırma</li> <li>3.4. Dizilerde Arama</li> <li>3.5. Dizilerde Sıralama</li> <li>3.6. Dinamik Diziler</li> </ol>	<ol style="list-style-type: none"> <li>1. Dizi oluşturarak diziye değer girmek</li> <li>2. Diziye girilen değerleri istenilen sıralama ölçütüne göre sıralayarak ekrana yazdırmak</li> <li>3. Dizide istenilen bir değeri aramak</li> <li>4. Dizi kullanımında oluşabilecek hataya karşı önlem almak</li> <li>5. Oluşan hatayı düzeltmek</li> </ol>	<ol style="list-style-type: none"> <li>1. Detaylara özen göstermek</li> <li>2. Dizileri kullanırken indislere dikkat etmek</li> <li>3. Kod yazım kurallarına dikkat etmek</li> <li>4. Kodların okunurluğuna dikkat etmek</li> <li>5. Problem çözümünde algoritma ve akış diyagramı kullanmak</li> </ol>
<b>SÜRE:</b> İşlemin yapılma süresi: 1 saat		İşlemi öğrenme süresi: 10 ders saati	
<b>NOT:</b> Bilgi sayfalarında verilen konuların öğrenci tarafından anlaşıldığını test eden ve işlem basamakları ile uyuşan bir uygulama faaliyeti oluşturulmasına dikkat ediniz. Programlama dili yazılımında konsol uygulamasını kullanmaya özen gösteriniz.			

## MODÜL BİLGİ SAYFASI

<b>ALAN</b>	<b>: BİLİŞİM TEKNOLOJİLERİ</b>
<b>MESLEK/DAL</b>	<b>: ALAN ORTAK</b>
<b>DERS</b>	<b>: PROGRAMLAMA TEMELLERİ</b>
<b>MODÜL</b>	<b>: METOTLAR</b>
<b>KODU</b>	<b>: 482BK0121</b>
<b>SÜRE</b>	<b>: 40/32</b>
<b>ÖN KOŞUL</b>	: “Kontrol Deyimleri” modülünü tamamlamış olmak.
<b>ÖĞRETİM YÖNTEM VE TEKNİKLERİ</b>	: Öğretim yöntem ve teknikleri olarak, anlatım, göstererek yaptırma, problem çözme, soru-cevap, grup çalışması, uygulamalı çalışma, araştırma ve bireysel öğretim yöntem ve teknikleri uygulanabilir.
<b>AÇIKLAMA</b>	: Uygulamalı olarak laboratuvar ortamında işlenmelidir.
<b>GENEL AMAÇ</b>	: Öğrenci bu modül ile gerekli ortam sağlandığında; temel programlama işlemlerinde metotlar hazırlayarak, mevcut metotları kullanabilecektir.
<b>AMAÇLAR</b>	: <ol style="list-style-type: none"><li>1. Metotları kullanabilecektir.</li><li>2. Hazır metotları kullanabilecektir.</li></ol>

## İÇERİK

### 1. METOTLAR

- 1.1. Metot Kavramı
- 1.2. Metot Tanımlama
- 1.3. Metotlarda Parametre Kullanımı
- 1.4. Metotlar İle İlgili Önemli Özellikler
- 1.5. Özyineli (Rekürsif) Metotlar
- 1.6. Main Metodu

### 2. HAZIR METOTLAR

- 2.1. Metinsel (String) Fonksiyonlar
- 2.2. Matematiksel Fonksiyonlar
- 2.3. Tarih ve Zaman Fonksiyonları

## ÖLÇME VE DEĞERLENDİRME:

Öğrenci, aşağıdaki performans ölçütlerini yerine getirecektir.

## **A. METOTLAR**

1. Tekrar eden işlemler için ihtiyaç duyulan metotları hazırlar.
2. İhtiyaç duyulan metodu program içerisinde uygun yerde çağırır.
3. Hata mesajlarına göre gerekli düzenlemeleri yapar.

## **B. HAZIR METOTLAR**

1. İşleme uygun hazır metodu seçer.
2. Program içinde hazır metotları parametrelerle birlikte kullanır.

### İŞLEM ANALİZ FORMU

<b>MESLEĞİN ADI</b>	ALAN ORTAK	<b>İŞ</b>	Temel Programlama işlemleri yapmak
<b>İŞLEM NUMARASI</b>	1	<b>İŞLEMİN ADI</b>	Metotları kullanmak
<b>YETERLİK</b>	Metotlarla çalışmak		
<b>ORTAM (Araç Gereç, Ekipman ve Koşullar)</b>	Bilgisayar, Programlama dili yazılımı		
<b>İŞLEMİN STANDARDI</b>	Metotları düzenli ve verimli kullanmak		
<b>İŞLEM BASAMAKLARI</b>	<b>BİLGİ</b>	<b>BECERİ</b>	<b>TAVİR</b>
<ol style="list-style-type: none"><li>1. Problemin algoritmasını oluşturmak</li><li>2. Problemin akış diyagramını oluşturmak</li><li>3. Akış diyagramını çizmek</li><li>4. Metot tanımlamak</li><li>5. Metot oluşturmak</li><li>6. Metot çağırarak</li><li>7. Kendi kendini çağıran rekürsif fonksiyonlar hazırlamak</li><li>8. Metot kullanımında oluşabilecek hataları gidermek</li></ol>	<ol style="list-style-type: none"><li>1. <b>METOTLAR</b><ol style="list-style-type: none"><li>1.1. Metot Kavramı</li><li>1.2. Metot Tanımlama</li><li>1.3. Metotlarda Parametre Kullanımı</li><li>1.4. Metotlar İle İlgili Önemli Özellikler</li><li>1.5. Özyineli (Rekürsif) Metotlar</li><li>1.6. Main Metodu</li></ol></li></ol>	<ol style="list-style-type: none"><li>1. Kullanıcı tanımlı metot hazırlamak</li><li>2. Program içerisinde kullanıcı tanımlı metotları kullanmak</li><li>3. Metot kullanımında oluşabilecek hataya karşı önlem almak</li></ol>	<ol style="list-style-type: none"><li>1. Kullanıcı tanımlı metot kullanmayı alışkanlık haline getirmek</li><li>2. Metot yazarken veri tipini ve kapsama alanını dikkate almak</li><li>3. Kod yazım kurallarına dikkat etmek</li><li>4. Kodların okunurluğuna dikkat etmek</li><li>5. Problem çözümünde algoritma ve akış diyagramı kullanmak</li></ol>
<b>SÜRE:</b> İşlemin yapılma süresi: 1 saat	İşlemi öğrenme süresi: 12 ders saati		
<b>NOT:</b> Bilgi sayfalarında verilen konuların öğrenci tarafından anlaşıldığını test eden ve işlem basamakları ile uyuşan bir uygulama faaliyeti oluşturulmasına dikkat ediniz. Programlama dili yazılımında konsol uygulamasını kullanmaya özen gösteriniz.			

**İŞLEM ANALİZ FORMU**

<b>MESLEĞİN ADI</b>	ALAN ORTAK	<b>İŞ</b>	Temel Programlama işlemleri yapmak
<b>İŞLEM NUMARASI</b>	2	<b>İŞLEMİN ADI</b>	Hazır metotları kullanmak
<b>YETERLİK</b>	Metotlarla çalışmak		
<b>ORTAM (Araç Gereç, Ekipman ve Koşullar)</b>	Bilgisayar, Programlama dili yazılımı		
<b>İŞLEMİN STANDARDI</b>	Program için gerekli hazır metotları düzenli ve verimli şekilde kullanmak		
<b>İŞLEM BASAMAKLARI</b>	<b>BİLGİ</b>	<b>BECERİ</b>	<b>TAVİR</b>
<ol style="list-style-type: none"><li>1. Problemin algoritmasını oluşturmak</li><li>2. Akış diyagramını çizmek</li><li>3. Hazır metot tanımlamak</li><li>4. Hazır metot oluşturmak</li><li>5. Hazır metot çağırmak</li><li>6. Metot kullanımında oluşabilecek hataları gidermek</li></ol>	<ol style="list-style-type: none"><li>2. <b>HAZIR METOTLAR</b><ol style="list-style-type: none"><li>2.1. Metinsel (String) Fonksiyonlar</li><li>2.2. Matematiksel Fonksiyonlar</li><li>2.3. Tarih ve Zaman Fonksiyonları</li></ol></li></ol>	<ol style="list-style-type: none"><li>1. Kullanıcı tanımlı metotlar hazırlamak</li><li>2. Program içerisinde kullanıcı tanımlı metotları kullanmak</li><li>3. Metotların kullanımında oluşabilecek hataya karşı önlem almak</li></ol>	<ol style="list-style-type: none"><li>1. Kullanıcı tanımlı metot kullanmayı alışkanlık haline getirmek</li><li>2. Metotları oluştururken veri tipini ve kapsama alanını dikkate almak</li><li>3. Kod yazım kurallarına dikkat etmek</li><li>4. Kodların okunurluğuna dikkat etmek</li><li>5. Problem çözümünde algoritma ve akış diyagramı kullanmak</li></ol>
<b>SÜRE:</b> İşlemin yapılma süresi: 1 saat	İşlemi öğrenme süresi: 20 ders saati		
<b>NOT:</b> Bilgi sayfalarında verilen konuların öğrenci tarafından anlaşıldığını test eden ve işlem basamakları ile uyuşan bir uygulama faaliyeti oluşturulmasına dikkat ediniz. Programlama dili yazılımında konsol uygulamasını kullanmaya özen gösteriniz.			