



2013  
2014

## Programlama Dilleri II

Ders Sorumlusu: Yrd. Doç. Dr. İsmail KURNAZ

Yrd. Doç. Dr. Ümit ATİLA

# İÇİNDEKİLER

<b>İÇİNDEKİLER .....</b>	<b>2</b>
<b>Deney 1: Pointerlar (İşaretçiler).....</b>	<b>3</b>
1.0. Amaç ve Kapsam .....	3
1.1. Uygulamadan Önce Yapılması Gerekenler .....	3
1.2. Sorular.....	3
1.3. Değerlendirme .....	3
<b>Deney 2: Pointerlar (İşaretçiler) II.....</b>	<b>4</b>
2.0. Amaç ve Kapsam .....	4
2.1. Uygulamadan Önce Yapılması Gerekenler .....	4
2.2. Sorular.....	4
2.3. Değerlendirme .....	4
<b>Deney 3: Bit Operatörleri .....</b>	<b>5</b>
3.0. Amaç ve Kapsam .....	5
3.1. Uygulamadan Önce Yapılması Gerekenler .....	5
3.2. Sorular.....	5
3.3. Değerlendirme .....	5
<b>Deney 4: Bağlı Listeler .....</b>	<b>6</b>
4.0. Amaç ve Kapsam .....	6
4.1. Uygulamadan Önce Yapılması Gerekenler .....	6
4.2. Sorular.....	6
4.3. Değerlendirme .....	6
<b>Deney 5: Dosya İşlemleri .....</b>	<b>7</b>
5.0. Amaç ve Kapsam .....	7
5.1. Uygulamadan Önce Yapılması Gerekenler .....	7
5.2. Sorular.....	7
5.3. Değerlendirme .....	7
<b>Deney 6: Visual C++ Uygulamaları I .....</b>	<b>8</b>
6.0. Amaç ve Kapsam .....	8
6.1. Uygulamadan Önce Yapılması Gerekenler .....	8
6.2. Sorular.....	8
6.3. Değerlendirme .....	8
<b>Deney 7: Visual C++ Uygulamaları II .....</b>	<b>9</b>
7.0. Amaç ve Kapsam .....	9
7.1. Uygulamadan Önce Yapılması Gerekenler .....	9
7.2. Sorular.....	9
7.3. Değerlendirme .....	10
<b>Deney 8: Visual C++ Uygulamaları III .....</b>	<b>11</b>
8.0. Amaç ve Kapsam .....	11
8.1. Uygulamadan Önce Yapılması Gerekenler .....	11
8.2. Sorular.....	11
8.3. Değerlendirme .....	11

## Deney 1: Pointerlar (İşaretçiler)

### 1.0. Amaç ve Kapsam

Bu deneyde pointer işlemleri öğrenilecektir. Bunlar;

- Pointer tanımlama
- Pointerdan veri ve adres okuma
- Pointerlar ile aritmetik işlemler
- Pointer dizileri oluşturma ve kullanmadır.

### 1.1. Uygulamadan Önce Yapılması Gerekenler

Uygulamaya başlamadan önce uygulanacak konular hakkında teorik bilgilerin araştırılarak okunması ve hazır kodlamaların incelenmesi gerekmektedir.

### 1.2. Sorular

1. Klavyeden 0 (sıfır) girilene kadar girilen integer sayıları okuyan ve bu sayıları ilk sayıdan başlayarak ekrana yazdıran programı yazınız.
2. Klavyeden girilen text bir ifadeyi okuyan ve bu text in içindeki harflerin sayısını veren programı yazınız.

Örnek : "Karabük Üniversitesi" için;

2 adet "a", 1 adet "b"....

### 1.3. Değerlendirme

Değerlendirmede dikkat edilecek noktalar;

- Programın çalışması
- Pointer tanımlama
- Pointerdan veri okuma
- Pointer dizisi oluşturma ve kullanımı
- Döngü kullanımı
- Ekran çıktılarının alınması

## Deney 2: Pointerlar (İşaretçiler) II

### 2.0. Amaç ve Kapsam

C++ 'da, dinamik diziler işaretçi kullanılarak ve standart kütüphanedeki malloc(), calloc(), realloc() ve free() fonksiyonlarının yardımıyla oluşturulur veya boşaltılır.

Bu deneyde malloc, calloc, realloc ve free fonksiyonlarının kullanımı, struct yapıları, birden fazla değişken türünün aynı yapı içerisinde nasıl kullanıldığı öğrenilecektir.

### 2.1. Uygulamadan Önce Yapılması Gerekenler

Uygulamaya başlamadan önce uygulanacak konular hakkında teorik bilgilerin araştırılarak okunması ve hazır kodlamaların incelenmesi gerekmektedir.

### 2.2. Sorular

1. Hafızadan n adet integer sayı için yer ayıran ve bunlara rastgele sayılar atayan programı yazınız. (n sayısı klavyeden girilecektir).
2. Geri dönüş değeri olmayan kare() isimli bir fonksiyon tanımlayın. Bu program;
  - Kendisine gönderilen sayının karesini hesaplar
  - Elde edilen sonucu aynı adresine yazar.
3. İsim, adres, telefon ve borç verilerinden oluşan bir struct yapısı oluşturun. Sonra bu yapı için n elemanlı bir dizi oluşturun (n klavyeden girilecek). Oluşturduğunuz dizinin içine klavyeden verileri girin. Girilen verileri ekrana yazdırmak için son bir fonksiyon yazın (void yazdir(struct \*z)).

### 2.3. Değerlendirme

Değerlendirmede dikkat edilecek noktalar aşağıdaki gibidir.

- Programın çalışması
- Dinamik dizi oluşturma
- Dizi için hafızadan yer ayırma ve silme
- Struct yapısı oluşturma ve kullanımı
- Ekran çıktılarının alınması

## Deney 3: Bit Operatörleri

### 3.0. Amaç ve Kapsam

Bitsel operatörler (Bitwise operators) bir tamsayının bitleri üzerinde işlem yapan operatörlerdir, daha çok sistem programlarında kullanılırlar. Bitsel operatörlerin ortak özellikleri operandları olan tamsayıları bir bütün olarak değil, bit bit ele alarak işleme sokmalarıdır. Bitsel operatörlerin operandları tamsayı türlerinden olmak zorundadır.

Bu deneyde bitsel operatörleri ve bunlarla gerçekleştirilen işlemler incelenecektir.

### 3.1. Uygulamadan Önce Yapılması Gerekenler

Uygulamaya başlamadan önce uygulanacak konular hakkında teorik bilgilerin araştırılarak okunması ve hazır kodlamaların incelenmesi gerekmektedir.

### 3.2. Sorular

1. Aşağıdaki fonksiyonları yazınız.
  - x değişkeninin ikili formunu geri döndüren  
void binary\_yaz(unsigned x);
  - x sayısının sağdan b. bitinden itibaren n bitini veren  
unsigned copybits(x, b, n);
  - x sayısının sağdan b. bitinden itibaren n bitinin tersini alan  
unsigned ters(x, b, n);
  - x' in n adet bitini sağa kaydıran  
unsigned rdon(x, n);

### 3.3. Değerlendirme

- Programın çalışması
- İstenilen fonksiyonları oluşturabilme
- Program içerisinde fonksiyonların kullanımı
- Ekran çıktılarının alınması

## Deney 4: Baęlı Listeler

### 4.0. Amaç ve Kapsam

Baęlı liste, her elemanın bir deęerinin yanında bir de referans içerdii veri yapısıdır. Bölünme nedeni ile ardışık bellek sıkıntısının çekildiđi durumlarda sabit eleman sayısı olmayan ve dinamik olarak büyüyen diziler için baęlı listeler kullanılır. Ayrıca baęlı listelerde ekleme silme vb. işlemler dizilere göre daha iş gerektirir.

Bu deneyde baęlı liste türlerini, aralarındaki yapısal farkları ve bunlar üzerinde node ekleme, silme gibi işlemlerin nasıl gerçekleştirildiğini inceleyeceğiz.

### 4.1. Uygulamadan Önce Yapılması Gerekenler

Uygulamaya başlamadan önce uygulanacak konular hakkında teorik bilgilerin araştırılarak okunması ve hazır kodlamaların incelenmesi gerekmektedir.

### 4.2. Sorular

1. Aşağıdaki adımları içeren bir program yazınız.
  - İsimler içeren doğrusal bir liste oluşturun (isimler klavyeden girilecek)
  - Bu listeye yeni bir node ekleyin
  - Listenin herhangi bir node' unu silin
  - Listedeki en ismin olduđu node' u bulun.
2. İsim ve final notu deęişkenlerini içeren bir struct yapısı tanımlayıp aşağıdaki adımları da içeren bir program yazınız.
  - Veriler dairesel bir liste içerisinde saklanmalıdır
  - N adet veri listeye girilecektir
  - Listedeki dersi geçenlerin isimleri ekrana yazdırılacaktır.

### 4.3. Deęerlendirme

- Programın çalışması
- Liste yapısının oluşturulması
- Listeye verilerin girilmesi ve okunması
- Liste üzerinde sorgular yapılabilmesi
- Ekran çıktılarının alınması

## Deney 5: Dosya İşlemleri

### 5.0. Amaç ve Kapsam

Fiziksel depolama ortamlarında verilerin saklandığı mantıksal yapılara dosya denir. Bu deneyde sıralı erişim dosyaların yapısını ve bunlar üzerinde yapılan işlemleri (kayıt ekleme, kayıt silme, kayıt düzenleme...) inceleyeceğiz.

### 5.1. Uygulamadan Önce Yapılması Gerekenler

Uygulamaya başlamadan önce uygulanacak konular hakkında teorik bilgilerin araştırılarak okunması ve hazır kodlamaların incelenmesi gerekmektedir.

### 5.2. Sorular

1. **Sizeof** operatörüyle değişik veri türlerinin boyutunu belirleyen ve bu değerleri "datasize.dat" isimli bir dosyaya kaydeden programı yazınız. Daha sonra dosyadaki verileri tablo şeklinde ekranda gösteriniz.
2. Aşağıdaki adımları takip eden programı yazınız.
  - Rastgele bir n sayısı oluşturun
  - n sayısının sınırlarını belirlemek için klavyeden iki adet sayı girin
  - Bu sayıları bir dosyaya yazın ve dosyayı kapatın
  - Sonra tekrar dosyayı açarak sayıların ortalamasını dosyanın sonuna ekleyin

### 5.3. Değerlendirme

- Programın çalışması
- Dosyaya veri yazma ve okuma
- Ekran çıktılarının belirtilen şekilde alınması

## Deney 6: Visual C++ Uygulamaları I

### 6.0. Amaç ve Kapsam

Bu deneyde Visual Studio ortamını tanıyıp görsel programlama hakkında giriş düzeyinde bilgiler edinilecektir. Deney sonunda Visual Studio ortamında kullanılan nesnelerin ve özelliklerinin kavranması hedeflenmektedir.

### 6.1. Uygulamadan Önce Yapılması Gerekenler

Uygulamaya başlamadan önce Visual Studio ortamı incelenmelidir. Visual Studio içerisinde öncelikle "Toolbox" ve "Properties" pencerelerinin işlevi araştırılıp buradaki nesnelere ve temel özellikleri öğrenilmelidir. Son olarak kullanılacak nesnelerin kodlanması hakkında bilgi edinilmelidir.

### 6.2. Sorular

1. Text kutusuna girilen bir sayıyı kontrol ederek (max 4 basamaklı olabilir), bu sayının basamak değerlerini bulup liste kutusuna bu basamakların okunuşları ekleyen programı yazınız. Örneğin sayı 1234 ise; liste kutusunda alt alta bir, iki, üç ve dört yazacak. Sayıların okunuşları tanımlanan diziden alınmalıdır.
2. ComboBox açılır kutusuna form açıldığında Türkiye'deki plaka kodlarını (01 - 81'e kadar) ekleyen programı yazınız.

### 6.3. Değerlendirme

- Kullanılacak nesnelerin uygun şekilde form üzerine yerleştirilmesi
- Nesnelerin özelliklerinin doğru kullanımı
- Programın çalışması



## Deney 7: Visual C++ Uygulamaları II

### 7.0. Amaç ve Kapsam

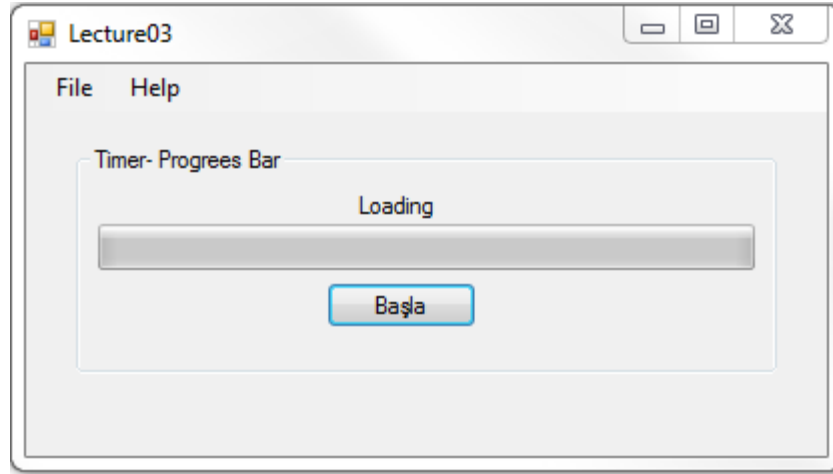
Bu deneyde Visual Studio ortamında kullanılan menuStrip, timer, progressBar, pictureBox, numericUpDown, panel ve groupBox nesnelerinin özellikleri ve ilgili olayları incelenecektir. Deney sonunda belirtilen nesnelere ilgili kodlamaların gerçekleştirilmesi hedeflenmektedir.

### 7.1. Uygulamadan Önce Yapılması Gerekenler

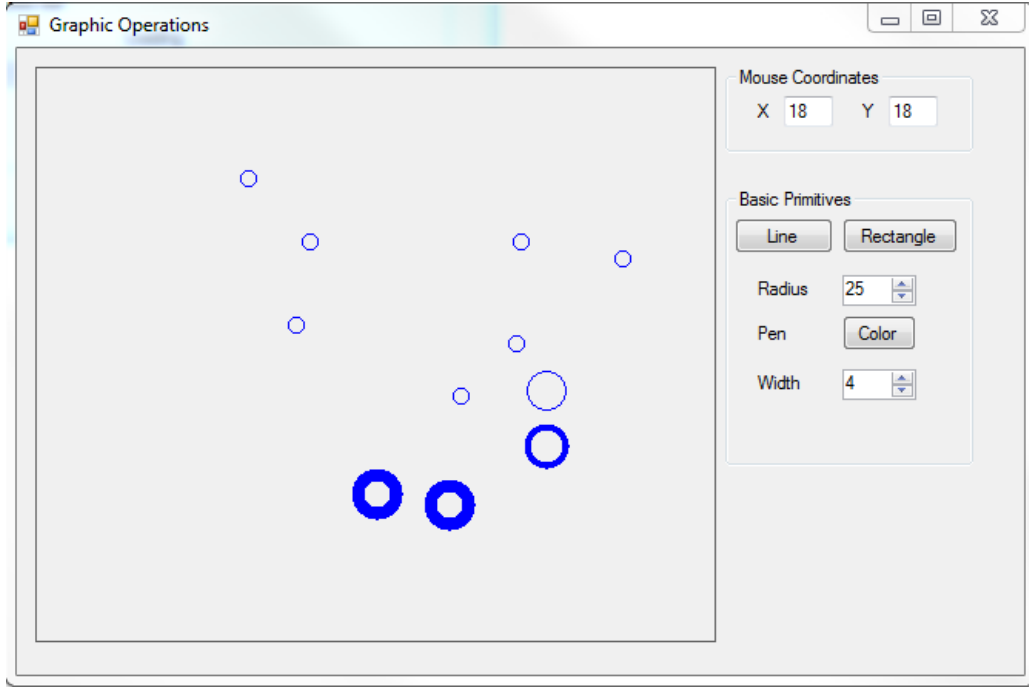
Uygulamaya başlamadan önce uygulanacak konular hakkında teorik bilgilerin araştırılarak okunması ve hazır kodlamaların incelenmesi gerekmektedir.

### 7.2. Sorular

1. Oluşturulan proje için 4 adet form tanımlayıp aşağıdaki adımları takip ediniz.
  - Başlangıç formunu aşağıdaki gibi tasarlayınız. Menü kısmında, File sekmesi altında (new, graphics, close), help sekmesi altında ise (about) seçenekleri bulunacaktır. Bu seçeneklere gerekli geçişler eklenecektir. Başla butonu tıkladığında ise progressBar saniyede 5 birim artacak şekilde ilerletilecektir.



- 'File->new' seçilerek açtığımız yeni form için üzerinde bir butonun bulunduğu boş bir form oluşturulacak. Buton tıkladığında yeniden ana forma dönüş sağlanacaktır.
- 'File->graphics' seçildiğinde aşağıdaki form açılacaktır. Mouse Coordinates kısmında mouse' un formun sol tarafına yerleştirilmiş olan pictureBox nesnesi üzerindeki koordinatları verecektir. Line butonu ekranın belli bir yerine çizgi, Rectangle butonu ise belirtilen en boy oranında bir dikdörtgen çizecektir. Yine Mouse ile ekranın tıklanan noktasına daire çizilecektir. Bu dairelerin çapı, rengi ve kalınlığı ise Basic Primitives bölümünden kullanıcı tarafından ayarlanacaktır.



- 'File->close' seçildiğinde program sonlandırılacaktır.
- 'Help->about' seçildiğinde ise açılan form üzerinde bir panel içerisinde programla ilgili kısa bir açıklama metni bulunacak yine ana forma dönmek için bir buton olacaktır.

### 7.3. Değerlendirme

- Kullanılacak nesnelerin uygun şekilde form üzerine yerleştirilmesi
- Nesnelerin özelliklerinin doğru kullanımı
- Programın çalışması

## Deney 8: Visual C++ Uygulamaları III

### 8.0. Amaç ve Kapsam

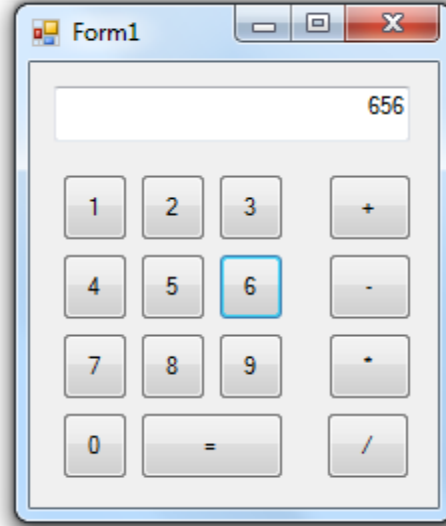
Bu deneyde Visual Studio ortamında gerekli nesnelere kullanılarak dört işlem yapabilen bir hesap makinesi uygulaması gerçekleştirilecektir.

### 8.1. Uygulamadan Önce Yapılması Gerekenler

Uygulamaya başlamadan önce uygulanacak konular hakkında teorik bilgilerin araştırılarak okunması ve hazır kodlamaların incelenmesi gerekmektedir.

### 8.2. Sorular

1. Aşağıdaki arayüzü oluşturarak dört işlem yapabilen bir hesap makinesi gerçekleştiriniz.



### 8.3. Değerlendirme

- Kullanılacak nesnelere uygun şekilde form üzerine yerleştirilmesi
- Nesnelere özelliklerinin doğru kullanımı
- Programın çalışması