

**Erzurum Atatürk Üniversitesi**

**Mühendislik Fakültesi**

**Bilgisayar Mühendisliği Bölümü**

**BM 104: Nesneye Yönelik Programlama - Bahar 2020**

**PROGRAMLAMA PROJESİ 2**

**(Son Teslim Tarihi: 30 Nisan 2020 - 23:59)**

**SORU 1:**  şeklinde ifade edilen 2. dereceden bir polinomu bilgisayar ortamında temsil edebilmek amacıyla **Polinom** adında bir **C++ sınıfı (class)** tanımlayınız.

* ve katsayıları ile sabiti sınıfınızın **int** veri tipinde **private** üye değişkenleri olmalıdır.
* Sınıfınızın **public** üye fonksiyonları ise aşağıda listelenmiştir:
  + - **KatsayilaraDegerAta** adındaki **void** üye fonksiyon sınıfınızın herhangi bir nesnesinin , ve adlı üye değişkenlerine değer aktarmak amacıyla kullanılmalıdır. Bu fonksiyon kendisine **int** veri tipinde 3 parametre almalıdır.
    - **DegerHesapla** adında, geriye bir **int** değer döndüren üye fonksiyon kendisine parametre olarak aktarılan parametresini (**int** veri tipinde) ve ilgili sınıf nesnesinin üye değişkenlerini kullanarak ’yi hesaplamalıdır. Bu fonksiyonun **KatsayilaraDegerAta** fonksiyonundan sonra kullanılması gerektiğini unutmayınız.
    - Kendisine herhangi bir parametre almayan **KokleriHesapla** adındaki **void** üye fonksiyon kendisini çağıran nesnenin üye değişkenlerini kullanarak bu polinomun kökü/kökleri olup olmadığını belirlemelidir ve ekrana yazdırmalıdır. Kökler sanal mıdır, birbirine eşit tek bir kök mü ya da iki farklı kök mü vardır sorularının tamamına bir cevap verilmelidir; eğer varsa kök/kökler ekrana yazdırılmalıdır. Bu fonksiyonun **KatsayilaraDegerAta** fonksiyonundan sonra kullanılması gerektiğini unutmayınız.
    - Sınıfınızın bir **public** varsayılan yapıcı fonksiyona (default constructor) ihtiyacı olup olmadığını düşününüz ve eğer gerekli ise bir adet tanımlayınız.

Geliştireceğiz program bir konsol uygulaması olacaktır. Kullanıcının gireceği parametreler ve kullanıcıya sunulacak çıktılar sizin tarafınızdan belirlenecektir. Kullanıcıyla etkileşim kurarken, yani konsol üzerinden gerçekleştirilecek tüm girdi ve çıktılarda, olabildiğince açıklayıcı ve yönlendirici olmalısınız (örneğin; “Gir:” demek yerine “x değişkeninin değeri olarak bir tamsayı giriniz:” demelisiniz).

Son olarak fonksiyonlara ait puanlandırmaların sınıfın doğru tanımlanması ile ilişkili olacağını ve ayrıca bu projenizi 3 farklı dosya kullanarak çözmeniz gerektiğini unutmayınız.

* **Polinom.h** (Sınıfın başlık dosyası)
* **Polinom.cpp** (Sınıfın tanım dosyası)
* **YurutucuProje2Soru1.cpp**

**YurutucuProje2Soru1.cpp** adlı dosyanın içerisinde **main** fonksiyonunu barındırdığını ve sınıfınızın doğru çalışıp çalışmadığını sınıfın bir nesnesini/nesnelerini kullanarak test etmek amacıyla yazıldığını hatırlayınız.

**SORU 2:**  ile temsil edilen ve bir elektrik devre elemanı olan direnci bilgisayar ortamında temsil edebilmek amacıyla **Direnc** adında bir **C++ sınıfı (class)** tanımlayınız.

* direnci (**Ohm** cinsinden **(Ω)**)sınıfınızın **int** veri tipindeki **private** üye değişkeni içerisinde saklanmalıdır.
* Sınıfınızın **public** üye fonksiyonları ise aşağıda listelenmiştir:
  + - **DegerAta** adındaki **void** üye fonksiyon sınıfınızın herhangi bir nesnesinin adlı üye değişkenine belirli bir direnç değerini aktarmak amacıyla kullanılmalıdır. Bu fonksiyon kendisine **int** veri tipinde tek bir parametre almalıdır.
    - **AkimHesapla** adında, geriye **double** değer döndüren üye fonksiyon kendisine parametre olarak aktarılan gerilim parametresini (**int** veri tipinde) ve ilgili sınıf nesnesinin üye değişkenini **()** kullanarak direncin içinden geçen akımı hesaplamalıdır. Bu fonksiyonun **DegerAta** fonksiyonundan sonra kullanılması gerektiğini unutmayınız.
    - **GerilimHesapla** adında, geriye **int** değer döndüren üye fonksiyon kendisine parametre olarak aktarılan akım parametresini (**int** veri tipinde) ve ilgili sınıf nesnesinin üye değişkenini **()** kullanarak direnç uçlarında oluşacak gerilimi hesaplamalıdır. Bu fonksiyonun **DegerAta** fonksiyonundan sonra kullanılması gerektiğini unutmayınız.
    - Sınıfınızın bir **public** varsayılan yapıcı fonksiyona (default constructor) ihtiyacı olup olmadığını düşününüz ve eğer gerekli ise bir adet tanımlayınız.

Geliştireceğiz program bir konsol uygulaması olacaktır. Kullanıcının gireceği parametreler ve kullanıcıya sunulacak çıktılar sizin tarafınızdan belirlenecektir. Kullanıcıyla etkileşim kurarken, yani konsol üzerinden gerçekleştirilecek tüm girdi ve çıktılarda, olabildiğince açıklayıcı ve yönlendirici olmalısınız (örneğin; “Gir:” demek yerine “V gerilim parametrisinin değeri olarak bir tamsayı giriniz:” demelisiniz).

Son olarak fonksiyonlara ait puanlandırmaların sınıfın doğru tanımlanması ile ilişkili olacağını ve ayrıca bu projenizi 3 farklı dosya kullanarak çözmeniz gerektiğini unutmayınız.

* **Direnc.h** (Sınıfın başlık dosyası)
* **Direnc.cpp** (Sınıfın tanım dosyası)
* **YurutucuProje2Soru2.cpp**

**YurutucuProje2Soru2.cpp** adlı dosyanın içerisinde **main** fonksiyonunu barındırdığını ve sınıfınızın doğru çalışıp çalışmadığını sınıfın bir nesnesini/nesnelerini kullanarak test etmek amacıyla yazıldığını hatırlayınız.

**PROJENİZDE DİKKAT ETMENİZ GEREKEN GENEL HUSUSLAR**

**1)** Uygulamanızı **DevC++** programını kullanılarak yazmanız beklenmektedir**.**

**2)** Oluşturduğunuz bütün dosyaların (**.h** veya **.cpp**) en üst bölümünde adınız, soyadınız, öğrenci numaranız, bölümünüz, dersin adı ve kodu, ayrıca dersin verildiği akademik dönem, projenin numarası ve dersi veren öğretim üyesinin adı **açıklama satırı şeklinde yer almalıdır.**

**3)** C sitil kütüphane deklarasyonuna izin **yoktur.** Bütün kütüphane deklarasyonları C++ sitilinde olmalıdır. Örneğin **<time.h>** yerine **<ctime>** kullanılmalıdır.

**4)** Uygulamanızda ilgili kütüphaneler muhakkak **include** edilmelidir. Örneğin **rand** ve **srand** fonksiyonlarını kullanabilmek için programınıza **<cstdlib>** kütüphanesi eklenmelidir.

**5)** Değişken isimlendirmelerinde deve notasyonu ve fonksiyon isimlendirmelerinde paskal notasyonu kullanılmalıdır.

**6)** Yazdığınız programın içerisine **MUHAKKAK** açıklayıcı satırlar eklenmelidir.

**7)** Visual Studio programının yazım sitiline müdahale edilmemelidir. Girintilere özellikle dikkat edilmelidir.

**8)** Teslim edeceğiniz proje **SADECE** ve **SADECE** kendi emeğinizin ürünü olmalıdır. Aksine hareket edenleri çok ciddi yaptırımların beklediğini unutmayınız.

**9) A**, **B**, **C** ve **D** grubundaki öğrencilerin teslim ettikleri kodların tamamının özel bir yazılım ile benzerlik testine tabi tutulacağını unutmayınız.

**10)** Hazırladığınız projeyi **OgrenciNumarasi-Ad-Soyad-BM104-Proje2.zip** şeklinde isimlendirilmiş sıkıştırılmış bir klasör formatında son teslim saatine kadar **DBS’ye (Ders Bilgi Sistemi)** yüklemeniz gerekmektedir. Sıkıştırılmış klasör içerisinde **sadece** aşağıdaki dosyalar olmalıdır:

* **Polinom.h**
* **Polinom.cpp**
* **YurutucuProje2Soru1.cpp**
* **Direnc.h**
* **Direnc.cpp**
* **YurutucuProje2Soru2.cpp**

**11) Proje teslim kurallarına uymayan ve/veya intihal içeren projeler değerlendirmeye alınmayacaktır.**