

**Erzurum Atatürk Üniversitesi**

**Mühendislik Fakültesi**

**Bilgisayar Mühendisliği Bölümü**

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

**(Dr. Öğr. Üyesi Deniz DAL)**

**PROGRAMLAMA PROJESİ 4**

**(Son Teslim Tarihi: 25 Mayıs 2020 - 23:59)**

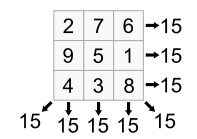
***İki boyutlu dinamik integer diziler*** üzerinde işlem yapabilmeyi kolaylaştıran **DinamikBellekYonetimi** adında bir **C++ sınıfı (class)** tanımlayınız.Sınıfınız detayları aşağıda verilen **2** **public** üye fonksiyona sahip olmalıdır. Ayrıca sınıfınız içerisinde **public** üye fonksiyonların işlevlerini kolaylaştıracak ve ilgili problemleri küçük parçalar şeklinde çözmenize yardımcı olacak **en az 3 adet** **private** üye fonksiyon kullanılmalıdır.

1. Prototibi **bool GizemliMatrisMi(int\*\*,int,int);** şeklinde tanımlı ve **GizemliMatrisMi** isimli birinci **public** üye fonksiyon anlaşıldığı üzere kendisine 3 parametre almaktadır. 1. parametre iki boyutlu dinamik bir integer diziyi işaret eden pointer’ı, 2. parametre iki boyutlu dinamik integer dizinin satır sayısını ve 3. parametre de iki boyutlu dinamik integer dizinin sütun sayısını ifade etmektedir. Bu üye fonksiyon kendisine aktarılan parametrelerle temsil edilen iki boyutlu dinamik integer dizi bir **Gizemli Matris** ise geriye **true**, aksi takdirde geriye **false** değerini döndürmelidir. Bir gizemli matrisin özellikleri aşağıda sıralanmıştır:

* Gizemli matris bir kare matristir. (Satır sayısı sütun sayısına eşittir.)
* **nxn** boyutundaki bir gizemli matris sadece **1** ile **n2** arasındaki tam sayıları içerir.
* Bir gizemli matris benzersiz/eşsiz elemanlardan oluşur.
* Bir gizemli matrisin her bir satırının, her bir sütununun, ana köşegen üzerindeki elemanlarının ve ikincil köşegen üzerindeki elemanlarının toplamı gizemli matris sabiti olan **M**’ye eşittir.
* **nxn** boyutundaki bir gizemli matrisin **M** sabiti aşağıdaki formül ile hesaplanır.



* Aşağıda verilen matris **M** sabiti **15** olan **3x3** boyutunda bir gizemli matristir.



1. Prototibi **int SansliMatrisElemanlarininSayisi(int\*\*,int,int);** şeklinde tanımlı ve **SansliMatrisElemanlarininSayisi** isimli ikinci **public** üye fonksiyon anlaşıldığı üzere kendisine 3 parametre almaktadır. 1. parametre iki boyutlu dinamik bir integer diziyi işaret eden pointer’ı, 2. parametre iki boyutlu dinamik integer dizinin satır sayısını ve 3. parametre de iki boyutlu dinamik integer dizinin sütun sayısını ifade etmektedir. Bu üye fonksiyon kendisine aktarılan parametrelerle temsil edilen iki boyutlu dinamik integer dizi içerisindeki **Şanslı Matris Elemanlarının** sayısını bularak geriye döndürmelidir. Bir şanslı matris elemanının özellikleri aşağıda sıralanmıştır:

* Bir şanslı matris elemanı tüm elemanları benzersiz/eşsiz elemanlardan oluşan bir matris içerisinde yer alır. Bir başka deyişle tekrarlı eleman barındıran bir matris içerisindeki şanslı matris elemanlarının sayısı sıfırdır.
* Bir şanslı matris elemanı bulunduğu satırdaki minimum ve bulunduğu sütundaki maksimum elemandır.
* Aşağıda verilen **M** matrisinin içerisinde sadece 1 adet şanslı matris elemanı vardır. Bu eleman 3. satırda ve 1. sütunda yer alan 15’tir.

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

* **DinamikBellekYonetimi.h** (Sınıfın başlık dosyası)
* **DinamikBellekYonetimi.cpp** (Sınıfın tanım dosyası)
* **Yurutucu.cpp**

**Yurutucu.cpp** adlı dosyanın içerisinde **main** fonksiyonunu barındırdığını ve sınıfınızın doğru çalışıp çalışmadığını sınıfın bir nesnesini/nesnelerini kullanarak test etmek amacıyla oluşturulduğunu hatırlayınız. **main** fonksiyonu içerisinde aşağıdaki senaryoları gerçekleyiniz:

1. Kullanıcıdan klavye aracılığıyla iki integer sayı temin ediniz ve bu sayılardan birincisini yine **main** içerisinde oluşturacağınız *iki boyutlu dinamik bir integer dizinin* satır sayısı ve ikincisini de sütun sayısı olarak kullanınız.
2. Kullanıcıdan klavye aracılığıyla iki integer sayı daha temin ediniz ve bu sayılardan birincisini yine **main** içerisinde oluşturacağınız *iki boyutlu bir başka dinamik integer dizinin* satır sayısı ve ikincisini de sütun sayısı olarak kullanınız.
3. **DinamikBellekYonetimi** isimli sınıfın bir nesnesini **main** içerisinde deklare ediniz.
4. İlk tanımladığınız iki boyutlu dinamik dizinin içerisini doldurunuz.
5. İkinci tanımladığınız iki boyutlu dinamik dizinin içerisini doldurunuz.
6. 3. adımda deklare ettiğiniz nesnenin **GizemliMatrisMi** isimli public üye fonksiyonunu ilk tanımladığınız iki boyutlu dinamik diziyi parametre olarak kullanarak çağırınız.
7. 3. adımda deklare ettiğiniz nesnenin **SansliMatrisElemanlarininSayisi** isimli public üye fonksiyonunu ikinci tanımladığınız iki boyutlu dinamik diziyi parametre olarak kullanarak çağırınız.
8. Her iki public üye fonksiyonun da doğru çalışıp çalışmadığını test etmek için problem tanımında örnek olarak verilen matrislerden faydalanabilirsiniz.
9. Yazdığınız fonksiyonların puanlandırma aşamasında daha büyük matrislerle ve senaryolarla çalıştırılıp test edileceğini ve alacağınız puanın bu testlerin sonucuna bağlı olduğunu hatırlayınız.
10. **main** fonksiyonunu sonlandırmadan önce dinamik olarak oluşturduğunuz iki boyutlu dizileri yok etmeyi unutmayınız.

**PROJENİZDE DİKKAT ETMENİZ GEREKEN GENEL HUSUSLAR**

**1)** Uygulamanızı **Dev C++**  programını kullanarak yazmanız beklenmektedir**.**

**2)** Oluşturduğunuz bütün dosyaların (**.h, .dev,** 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)** Hazırladığınız projeyi **OgrenciNumarasi-Ad-Soyad-BM104-Proje4.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:

* **DinamikBellekYonetimi.h**
* **DinamikBellekYonetimi.cpp**
* **Yurutucu.cpp**

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

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

**11)** Proje teslim kurallarına uymayan ve/veya intihal içeren projelerin değerlendirmeye alınmayacağını unutmayınız.