

**Middterm Assignment: A "Matrix, Image and Table Operations" Program - Dönem Ödevi: "Matris, Görüntü ve Tablo İşlemleri" Yazılımı**

**Purpose - Amaç**

The purpose of this assignment is to help you learn or review (1) the fundamentals of the C++ programming language and coding some mathematical operations in C++, (2) the inheritance and template concepts, (3) designing template classes, (4) how to read from and write to files on disk, (5) some morphological operations in image processing, and (6) function and operator overloading concepts. - *Bu ödevin amacı, (1) C++ programlama dilinin temellerini ve veri bilimcilerinin kullandığı bazı matematiksel işlemleri, (2) kalıtım ve şablon kavramlarını, (3) şablon sınıflar tasarlamayı, (4) diskteki dosyaları okumayı ve bunlara yazmayı, (5) görüntü işlemede bazı morfolojik işlemler yapabilmeyi, (6) fonksiyon ve operatör aşırı yüklemeyi öğrenmenizi ve gözden geçirmenizi sağlamaktır.*

**Rules - Kurallar**

Make sure you do coding of your program on your own. You can ask for help from your friends to understand the problem, figure out how to tackle the problem, but you may never copy someones code and present it as your own. If this is detected, you will receive a caution, your code will not be marked. If you copied your work from your classmate, you will both receive a caution and your work will not be marked. - *Programınızın kaynak kodunu kendi başınıza yazınız. Problemi anlamak için arkadaşlarınızdan yardım isteyebilir, problemin nasıl çözüleceğini bulabilir, ancak asla başka birisinin kodunu kopyalayıp kendinizinmiş gibi sunamazsınız. Bu tespit edilirse, bir uyarı alırsınız ve kodunuz sıfır (0) puan olarak değerlendirilir. Çalışmanızı sınıf arkadaşınızdan kopyaladıysanız, hem bir uyarı alacaksınız hem de kodunuz sıfır (0) puan olarak değerlendirilir.*

**Background - Arkaplan**

It is necessary for the data scientist to perform matrix operations, image processing applications and to study some tabular data. Therefore, it is necessary to create an application that the data scientist can use. To do this, you need to encode a program for the data scientist to perform the necessary operations. - *Veri bilimcisinin matris işlemleri, görüntü işleme uygulamaları yapması ve bazı tablo verilerini incelemesi gerekmektedir. Bu nedenle, veri bilimcisinin kullanabileceği bir uygulama oluşturmak oldukça önemlidir. Veri bilimcisinin gerekli işlemleri yapması için bir program kodlamanız gerekir.*

**The Task - Yapılacak İş**

You are required to code a matrix, image and table classes. - *Bir matris, görüntü ve tablo sınıfları kodlamanız istenmektedir.*

- A Matrix template class . We must be able to do the following operations in main function with matrix class. - *Tamsayı değerleri olan bir Matrix sınıfı olmalıdır. Ana fonksiyonda aşağıdaki işlemleri matris sınıfıyla yapabilmeliyiz.*
  - `Matrix<int> *m1 = new Matrix<int>();` → 10x10 matrix with zeros. - *0 değerli 10x10 matris.*
  - `Matrix<int> *m2 = new Matrix<int>(3,2,1);` → 3x2 matrix with ones. - *1 değerli 3x2 matris.*
  - `Matrix<int> *m2 = new Matrix<int>(3,4,9);` → 3x4 matrix with 9's. - *9 değerli 3x4 matris.*
  - `Matrix<int> *m3 = new Matrix<int>(5,5,'e');` → 5x5 identification matrix. - *5x5 birim matris*
  - `Matrix<int> *m4 = new Matrix<int>(15,15,'r');` → 15x15 matrix with random integers between [0-255] - *değerleri 0-255 arasında rastgele değerler olan 15x15 matris.*
  - `m2->resize(5,5);` → matrix resize operations after creation - *matris oluşturulduktan sonra boyutu değiştirilmelidir.*
  - `m3->print();` → print matrix on the screen - *matrisin ekrana yazdırılması.*
  - `m3->print("file.txt");` → write to file - *matrisin dosyaya yazdırılması.*
  - matrix - matrix operations → *matris -matris işlemleri*
    - `(*m1) = (*m1) + (*m2);` // m1, m2, pointer ise
    - `m1 = m1 - m2;` // m1, m2, nesne ise
    - `m1 = m1 * m2;`

- matrix - scalar operations → *matris - skalar işlemleri*
  - $m1 = m1 + 12;$
  - $m1 = m1 - 20;$
  - $m1 = m1 * 10;$
  - $m1 = m1 / 10;$
  - $m1 = m1 \% 10;$
  - $m1 = m1^2;$  // power, üs alma
- matrix operations - *matris işlemleri*, sonuç yeni bir matris nesnesi olarak geri döndürülecek
  - $m1 \rightarrow T();$  → transpoze
  - $m1 \rightarrow \text{emul}(m2);$  → elementwise multiply - *eleman düzeyinde çarpma*,
  - $m1 \rightarrow \text{inv}()$  → matrix inversion if available - *matrisin tersi*
  - $m1 \rightarrow \text{det}()$  → determinant of matrix - *matrisin determinanti*
- An Image class derived from Matrix class, Image type must be a structure rgb which has three values (red,green,blue). - *Matrix sınıfından türetilmiş Image sınıfı, Image türü (kırmızı, yeşil, mavi) değerler içeren bir struct yapısıdır.*
  - The image could be gray level or color with the range of 0-255 values. If the image is gray, the red and blue values are zero - *Görüntü 0-255 değer aralığında gri veya renkli olabilir. Görüntü gri ise kırmızı ve mavi bileşenler 0 olarak alınır.*
  - Gray level image has one integer value as illuminance per pixel. - *Gri seviye görüntünün, piksel başına 1 adet 0-255 arasında parlaklık değeri vardır.*
  - Color image has three values as (red,green,blue) per pixel. - *Renkli görüntünün piksel başına (kırmızı, yeşil, mavi) üç değeri vardır.*
  - This class must contain the following methods to perform desired image processing operations: - *Bu sınıf, istenen görüntü işleme işlemlerini gerçekleştirmek için aşağıdaki methodları içermelidir.*
    - $\text{Image}(\text{rgb})()$  → default constructor, with 255x255 image size, default pixel values are 0. - *varsayılan yapıcı, 255x255 görüntü boyutuna sahip, varsayılan piksel değerleri 0'dır.*
    - $\text{Image}(\text{rgb})(\text{int width}, \text{int height})$  → constructor with (widthxheight) image size, default pixel values are 0. - *(width x height) görüntü boyutuna sahip yapıcı, varsayılan piksel değerleri 0*
    - $\text{Image}(\text{rgb})(\text{string filename}, \text{string format})$  → read image from file. if the format is 'bmp' then the image must read from bmp file (you should browse bmp file format on the internet). Additionally, binary file format could be read from file. In this file format the file contains an integer array as [height,width,m\_00,m\_01,...] etc. - *görüntüyü dosyadan oku. dosya formatı "bmp" ya da "bin" olabilir. bmp dosya formatına internetten araştırarak öğrenebilirsiniz. bin dosya biçiminde dosyanın ilk iki bytelık kısmı yükseklik ve genişlik geri kalan değerler sırasıyla piksel değerleridir.*
    - $\text{imread}(\text{string filename}, \text{string format})$  → read image from file (gray or color) - *dosyadan görüntüyü oku (renkli ya da gri)*
    - $\text{imwrite}(\text{string filename}, \text{string format})$  → write image to a file specified with filename. Format could be "bmp" or "bin". - *filename ile belirtilen dosyaya görüntüyü yaz. format bmp ya da bin olabilir.*
    - $\text{color2gray}()$  → convert color image to gray level. *renkli görüntüyü gri seviyeye çevir*
    - $\text{gray2binary}(\text{int thr})$  → convert gray image to binary with given threshold. - *gri görüntüyü verilen eşik değeri kullanarak ikili görüntüye çevir.*
    - $\text{erosion}()$  → if the image is binary, the morphological erosion must be done, if not, no need any operation. - *Eğer görüntü ikili ise, morfolojik erozyon yapılmalıdır, eğer değilse, herhangi bir işleme gerek yoktur.*
    - $\text{dilation}()$  → if the image is binary, the morphological dilation must be done, if not, no need any operation. - *Eğer görüntü ikili ise, morfolojik genişleme yapılmalıdır, eğer değilse, herhangi bir işleme gerek yoktur*

- opening() → if the image is binary, the morphological opening must be done, if not, no need any operation. - *Eğer görüntü ikili ise, morfolojik açma yapılmalıdır, eğer değilse, herhangi bir işleme gerek yoktur*
- closing() → if the image is binary, the morphological closing must be done, if not, no need any operation. - *Eğer görüntü ikili ise, morfolojik kapama yapılmalıdır, eğer değilse, herhangi bir işleme gerek yoktur*
- A Table class derived from Matrix class. - *Matrix sınıfından türetilmiş bir Table sınıfı.*
  - this class must contain special row names and col names. - *bu sınıf özel satır isimleri ve sütun isimleri içermelidir.*
  - default row names must be as '0', '1', '2', ... etc. - *varsayılan satır isimleri '0', '1', '2', ... şeklinde olmalıdır.*
  - default col names must be as 'A', 'B', 'C', ... etc. - *varsayılan sütun isimleri 'A', 'B', 'C', ... (en fazla 'Z') şeklinde olmalıdır.*
  - This class must contain the following methods to perform desired operations: - *Bu sınıf, istenen işlemleri gerçekleştirmek için aşağıdaki methodları içermelidir.*
    - Table<int>() → default constructor, a 10x10 table with zeros. - *varsayılan yapıcı, sıfırlara sahip 10x10'luk bir tablo*
    - Table<int>(int r,int c,int d) → rxc table with the value of d. - *d değerine sahip rxc boyutlu bir tablo.*
    - Table<int>(int r,int c,char ch) → rxc table. if the ch is 'r', then the value of the table must be random between 0 and 255. - *ch değeri 'r' ise 0-255 arasında rastgele değerlere sahip tablo*
    - itemAt(int r,int c) → return the value at [r,c] index - *[r,c] indisindeki değeri geri döndür*
    - itemAt(string s) → return the value at the location specified by s. if s is "A2", then the returning value must be the value at [2,0] index. sample code: `itemAt("A2")` - *belirtilen hücredeki değeri geri döndür. eğer s=="A2" ise [2,0] indisindeki değeri geri döndür demektir. örnek kod: itemAt("A2")*
    - itemAt(string rs,string cs) → sample code - *örnek kod: itemAt("1","A")*
    - setRowNames(string s[],int n) → set row names with given the string s. n is the length of s. sample code: *verilen s stringi ve n sayısına göre satır isimlerini günceller.*
      - `string s[] = {"0. satır","1. satır","2. satır"};`
      - `tb->setRowNames(s,3);`
      - `cout << tb->itemAt("1. satır","B");`
    - setColNames(string s[],int n) → set col names with given the string s. n is the length of s. - *satır isimlerinin güncellenmesine benzer şekilde sütun isimlerini günceller.*
    - print() → print the table values with row and col names, row names could be 8 characters at most and col names could be 4 characters at most. - *Tablo değerlerini satır ve sütun adlarıyla yazdırır; satır adları en fazla 8 karakter, sütun adları en fazla 4 karakter olabilir.*

### **Sending the Assignment - Ödevin Gönderilmesi**

Your submission should contain only one source file and the file name must be your student ID. Example filename could be like this: 123456.cpp. Your name, student ID, a brief description of the assignment, your e-mail address and your mobile phone number must be included as comment at the top of the file. Be sure that your file could be compiled using gcc/clang compiler on unix/linux (like ubuntu) systems. Compiling and running the source code is your responsibility. Then, you must submit your file at the latest by the **12th week** of the semester using the moodle / classroom platform used by your instructor.. - Gönderiniz sadece bir kaynak dosya içermeli ve dosya adı öğrenci numaranız olmalıdır. Örnek dosya adı şöyle olabilir: 123456.cpp. Cpp dosyasının en başına Adınızı, öğrenci numaranızı, ödevin kısa bir tanımını, e-posta adresinizi ve cep telefonu numaranızı yorum satırı olarak ekleyiniz. Dosyanızın unix / linux (ubuntu gibi) sistemlerinde gcc / clang compiler kullanılarak derlendiğinden emin olun. Kaynak kodu derlemek ve çalıştırmak sizin

sorumluluğunuzdadır. Daha sonra dosyanızı, en geç dönemin **12. haftası** içerisinde öğretim elemanınızın kullandığı moodle / classroom platformunu kullanarak gönderebilirsiniz.

### **Grading - Değerlendirme**

We will grade your work on two kinds of quality: quality from *the user's* point of view, and quality from *the programmer's* point of view. From the user's point of view, a program has quality if it behaves as it should. From the programmer's point of view, a program has quality if it is well styled and thereby easy to maintain. For this assignment these rules will be applied: Naming convention (variable, function, class, etc...), Comments (in whole source code or in class), Function/Class modularity (Don't Repeat Yourself-DRY principle), Line lengths (Limit line lengths in your source code to 72 characters) : - *Çalışmanızı iki tür kalite üzerinde değerlendireceğiz: kullanıcı bakış açısından ve programcı bakış açısından. Kullanıcı bakış açısından, bir program olması gerektiği gibi davranırsa yeterli kaliteye sahiptir. Programcı bakış açısına göre, bir program, iyi tasarlanmışsa ve dolayısıyla bakımı kolaysa, yeterli kaliteye sahiptir. Bu ödev için şu kurallar uygulanacaktır: Adlandırma kuralı (değişken, işlev, sınıf, vb ...), Yorumlar (tüm kaynak kodunda veya sınıfta), Fonksiyon / Sınıf modülerliği (Kendinizi tekrar etmeyin prensibi), Satır uzunlukları (Kaynak kodunuzdaki satır uzunluklarını 72 karakterle sınırlayın):*

### **Special Notes - Özel Notlar**

-