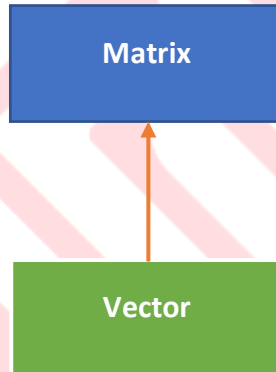


Fatih Sultan Mehmet Vakıf Üniversitesi, Mühendislik Fakültesi,
Bilgisayar Mühendisliği Bölümü, 2020 – 2021 GÜZ Dönemi,
BLM204, BLM217 ve BLM19204 dersleri Final Projesi

Bu projede Lineer Cebir'in temel öğeleri olan matris ve vektörler üzerinde tanımlı temel cebirsel işlemleri içinde barındıran Linear Algebra Library (LAL) adlı bir C++ kütüphanesi oluşturulacaktır. Bu kütüphanenin özellikleri ve yapılması istenenler ile ilgili talimatlar aşağıda verilmiştir.

- ✓ 1. Projede üç adet dosya bulunacaktır. Bu dosyaların isim ve uzantıları aşağıdaki gibidir:
 - a. *lal.h*
 - b. *lal.c*
 - c. *test.c*
- ✓ 2. Projede iki adet sınıf bulunmaktadır. Bu sınıfların adları aşağıdaki gibidir:
 - a. **Matrix**
 - b. **Vector**
- ✓ 3. Lineer Cebirde vektörler aslında sütun adedi 1 olan matrislerdir. Dolayısıyla aralarında "is a" ilişkisi bulunmaktadır (Vector is a Matrix). Bu nedenle aralarında aşağıdaki gibi bir kalıtım ilişkisi bulunmaktadır:



- ✓ 4. **Matrix** ve **Vector** sınıflarına ait UML tabloları aşağıdaki gibidir:

| Matrix |
|----------------------------|
| -entries: double * |
| -rows: int |
| -cols: int |
| +Matrix(double *,int, int) |
| +~Matrix() |
| +print(void):void |

| Vector |
|------------------------|
| -l2norm:double |
| +Vector(double *, int) |
| +~Vector() |
| +print(void):void |

- ✓ 5. **Matrix** sınıfının yapıcı metodu ilk argüman olarak matrisin **entries** değişkenini bir `double *` ile dolduracaktır. Aynı zamanda **rows** ve **cols** değişkenlerinin değerleri de yine yapıcı fonksiyon yardımıyla oluşturulacaktır.
- ✓ 6. **Vector** sınıfının yapıcı metodu ilk argüman olarak ilgili vektörün elemanlarını, ikinci argümanı ise ilgili vektörün eleman sayısını (satır sayısı) belirtmektedir. (Not: Vektörler

sütun adedi 1 olan matrislerdir) Bunların yanı sıra kurucu fonksiyon ilgili vektörün L_2 normunu hesaplayarak L_2 norm değişkenine atayacaktır.

- ✓ 7. Hem Matrix hem de Vector sınıfını yıkıcı fonksiyonları implemente edilecektir.
- ✓ 8. Hem Matrix hem de Vector sınıfında bulunan `print()` metodları ilgili matris ya da vektörü formatlı biçimde ekrana bastırarak şekilde implemente edileceklerdir.
- 9. Aşağıda belirtilen operatörler aşırı yüklenerek istenilen işlemi gerçekleştirecek biçimde tasarlanacaklardır. İstenilen işlemler Lineer Cebir'deki boyut uyumları sağlandığı müddetçe gerçekleştirilecek aksi takdirde *"İşlem geçersiz: boyut uyumsuzluğu!"* mesajı döndürülecektir.
 - a. + operatörü aşırı yüklenerek aşağıdaki işlemlerin gerçekleşmesi sağlanacaktır.
 - ✓ i. İki matrisin toplamı ($M_1 + M_2$)
 - ✓ ii. İki vektörün toplamı ($v_1 + v_2$)
 - b. - operatörü aşırı yüklenerek aşağıdaki işlemlerin gerçekleşmesi sağlanacaktır.
 - ✓ i. İki matrisin farkı ($M_1 - M_2$)
 - ✓ ii. İki vektörün farkı ($v_1 - v_2$)
 - c. * operatörü aşırı yüklenerek aşağıdaki işlemlerin gerçekleşmesi sağlanacaktır.
 - ✓ i. İki matrisin çarpımı ($M_1 \times M_2$)
 - ✓ ii. İki vektörün iç çarpımı ($v_1 \cdot v_2$)
 - ✓ iii. Matris vektör çarpımı ($M \times v$)
 - ✓ d. == operatörü aşırı yüklenerek kullanıldığında iki matrisin (iki vektörün) birbirine eşit olduğu durumlarda `true`, aksi takdirde `false` döndürülmesi sağlanacaktır.
 - ✓ e. `v` Vector sınıfından bir nesne olmak üzere, `++v` ve `v++` şeklinde yazıldığında `v` vektörünün tüm elemanlarını pre-increment ve post-increment kurallarına göre bir arttıran `++` operatörü aşırı yüklenecektir.
- 10. Ne Matrix, ne de Vector sınıfına ait olan tek bir `transpose()` metodu ile ilgili matris ya da vektörün transpozesi alınıp geri döndürülecektir. (Yardım: `typeid` operatörünün özelliklerinden yararlanabilirsiniz.)
- 11. Ne Matrix, ne de Vector sınıfına ait olan tek bir `norm()` metodu ile ilgili matris ya da vektörün L_2 normu bulunup geri döndürülecektir. (Yardım: `typeid` operatörünün özelliklerinden yararlanabilirsiniz.)
- ✓ 12. `test.c` içerisinde `p` sayıda `m` adet vektörü bir arada tutan Vector * tipindeki `vectorArray` dizisi oluşturulup içi doldurulacaktır. `vectorArray` dizisinin elemanları olan vektörlerin elemanları ise rasgele sayılarla doldurulacaktır.
- 13. 12'de sözü edilen `vectorArray` dizisi ile `m` ve `p` değerlerini argüman olarak alıp `vectorArray` dizisindeki vektörleri yan yana dizerek Matrix sınıfında `p x m`'lik bir matris geri döndüren `vectorArray2Matrix` metodunu implemente ediniz.
- 14. Hem Matrix, hem de Vector sınıfından nesneleri bir arada tutabilen bir dizi oluşturulup içi en az ikişer matris ve vektörle doldurulacaktır.
- ✓ 15. Projenizi sorunsuz biçimde derleyen `makeFile` dosyasını oluşturunuz.

Başarılar!

Dr. Öğr. Üyesi Süha Tuna

Araş. Gör. Kadir Aram

Araş. Gör. Samet Kaya