

# KOCAELİ ÜNİVERSİTESİ

## BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

### PROGRAMLAMA LABORATUVARI-I

#### PROJE 3

#### MAKALE GRAF ANALİZ UYGULAMASI

Proje Teslim Tarihi: 29.12.2025

## 1. Giriş ve Proje Tanımı

### 1.1. Proje Tanımı

Bu projede, bilimsel makaleler arasındaki atıf ilişkilerini yönlü bir graf yapısında modelleyen, oluşturulan bu model üzerinde çeşitli graf analiz metriklerini hesaplayan ve yapılan sorguların sonucunu görsel olarak kullanıcıya döndüren bir uygulama geliştirilmesi beklenmektedir.

### 1.2. Temel Amaçlar

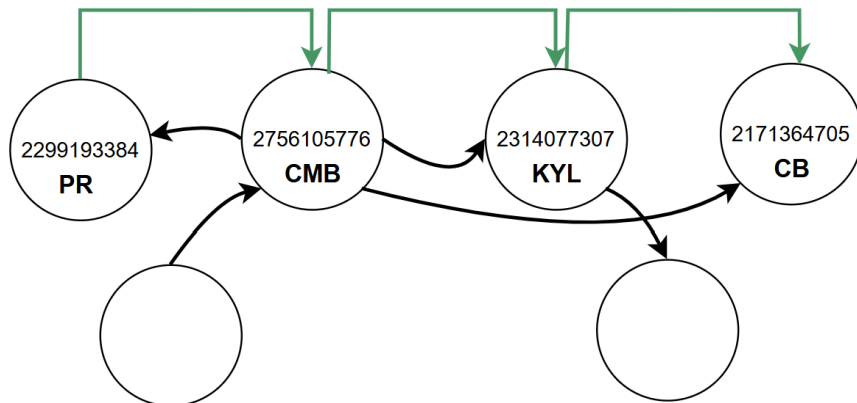
Bu projenin tamamlanmasıyla aşağıdaki yetkinliklerin kazanılması ve hedeflerin gerçekleştirilmesi amaçlanmaktadır:

- Graph yapılarını modelleme ve uygulama pratiğinin geliştirilmesi
- Temel graph analiz ölçütlerinin öğrenilmesi
- Graflarda görselleştirme işlemlerini gerçekleştirme

## 2. Proje Adımları

### 2.1. Graf Yapısının Oluşturulması

Verilen .json dosyasındaki her makale için id ve referenced\_works alanları kullanılarak graf yapısı oluşturulacaktır. Şekil 1’de graftaki her düğüm, id ile tanımlanan bir makaleye karşılık gelirken yönlü her siyah kenar ise referans veren makaleden (düğümden) referans verilen makaleye (düğüme) yönelik ilişkiyi temsil etmektedir. Yeşil kenarlar ise makaleleri artan id sırasına göre birbirlerine bağlamaktadır.



Şekil 1. Örnek graph, düğüm ve kenar gösterimi

```

{
  "id": "https://openalex.org/W2756105776",
  "doi": "https://doi.org/10.22336/rjo.2017.21",
  "title": "Eyesight quality and Computer Vision Syndrome",
  "year": 2017,
  "authors": [
    "Camelia Margareta Bogdănici",
    "Diana Elena Săndulache",
    "Corina Andreea Nechita"
  ],
  "venue": "Romanian Journal of Ophthalmology",
  "keywords": [
    "Blurred vision",
    "Medicine",
    "Gadget"
  ],
  "referenced_works": [
    "https://openalex.org/W2299193384",
    "https://openalex.org/W2314077307",
    "https://openalex.org/W2171364705"
  ],
  "in_json_reference_count": 3
}

```

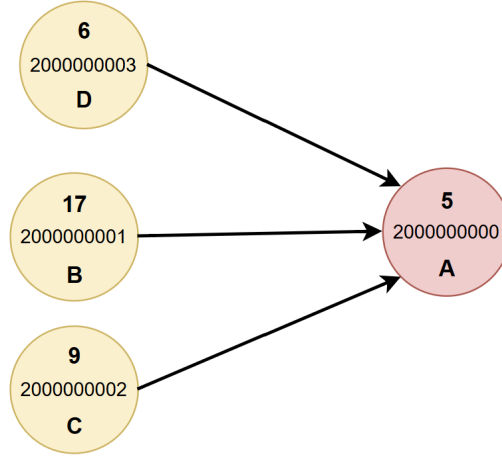
Şekil 2. Örnek makale dosya verisi (.json)

Şekil 1’deki grafta Şekil 2’de gösterilen dosya verisindeki bir “id” numaralı makalenin “referenced\_works” alanından yine “id” bilgileri verilen makaleleri ilişkilendirmesi örneklendirilmiştir. Projenin 1. adımında verilen json dosyasındaki tüm makalelerin id, authors, title, year ve referans bilgileri kullanılarak graf modeli oluşturulacaktır. Oluşan grafta bulunan toplam makale (düğüm) sayısı, toplam referans (siyah kenar) sayısı, toplam verilen referans sayısı, toplam alınan referans sayısı, en çok referans alan makale id ve aldığı referans sayısı, en çok referans veren makale id ve verdiği referans sayısı bilgileri arayüzde dinamik olarak gösterilecektir.

## 2.2. Makale H-Index Skorunun Hesaplanması

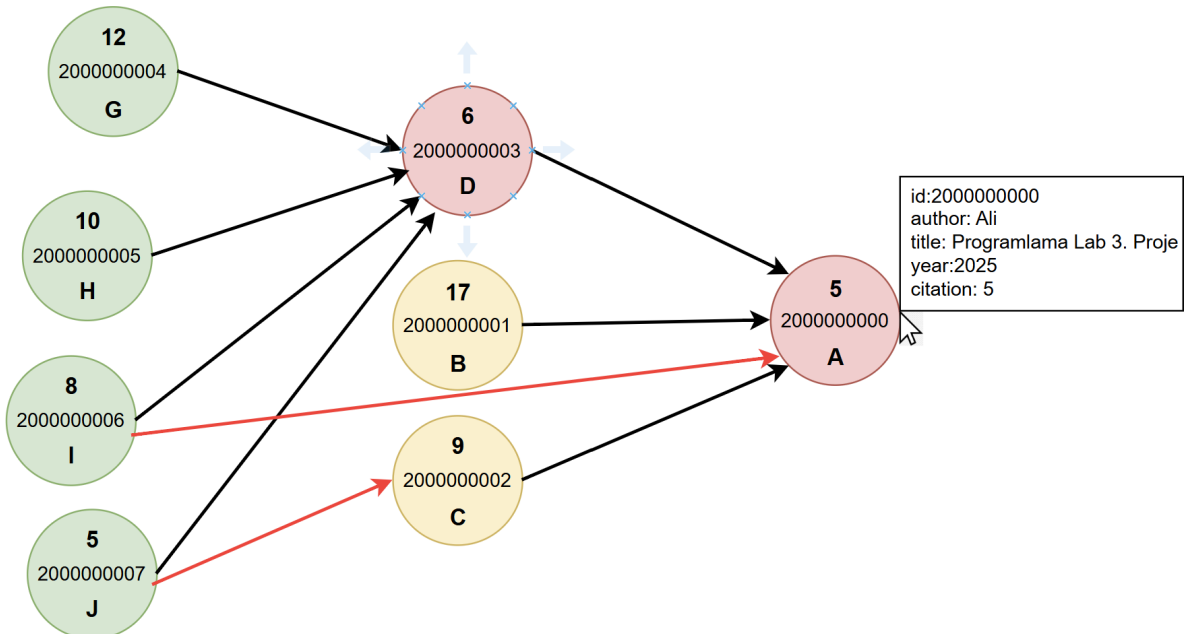
- İlk aşamada oluşturulmuş graf kullanılarak kullanıcı tarafından id bilgisi girilen bir makalenin h-index, h-core ve h-median sonuçları aşağıda verilen tanımlara uygun olarak bulunmalıdır:
  - **h-index:** Bir makalenin h-indeksi, bu makaleye atıf yapan (referans veren) makaleler içerisinde en az h atıfa sahip minimum h makalenin bulunma koşulunu sağlayan en büyük h sayısını ifade eder. Örneğin, bir A makalesine atıf yapan B, C, D, E, F makaleleri olsun. Bu 5 makalenin de kendilerine yapılan atıf sayılarının büyükten küçüğe sırasıyla 17, 9, 6, 3 ve 2 olduğunu varsayalım. Bu durumda A makalesinin h-indeksi 3 olacaktır.
  - **h-core :** Bir makaleye atıf yapan makaleler içerisinde kendisine en çok atıf alan h sayıdaki makalelerin kümesidir. Yukarıdaki örnekte A makalesinin h indeks şartını sağlayan {B, C, D} makale kümesi h-core a karşılık gelir.
  - **h-median:** h-core u oluşturan makalelerin atıf sayılarının median (ortanca) değeridir. Yukarıdaki örnekte B, C, D makalelerinin kendilerine yapılan atıf sayıları 17, 9 ve 6 dır. Bunlardan ortancası C makalesi olduğu için bu örnekte median değeri 9 olur.

- Hesaplanan h-index ve h-median değerleri arayüzde gösterilmeli ayrıca h-core u içeren ilgili makalelerin graf görseli Şekil 3 deki gibi arayüzde verilmelidir.



Şekil 3. h-core graf gösterimi

- Gösterilen grafa herhangi bir düğüm üzerine imleç getirildiğinde o düğümün temsil ettiği makalenin id, authors, title, year ve atıf sayısı (makaleye referans veren makale sayısı hesaplanarak bulunmalı) bilgileri düğümün üzerinde bilgi kartı şeklinde anlık gösterilmelidir.
- Çizilen graf üzerindeki herhangi bir düğüme tıklıldığında bu sefer tıklanan düğümün temsil ettiği makalenin h-index, h- median sonuçları arayüzde gösterilmeli ve yeni makalenin h-core düğümleri mevcut graf yapısına entegre edilerek graf genişletilmelidir. Bu işlem kullanıcı başka bir düğüme her tıkladığında yeniden tekrarlanarak yeni düğüm ve kenarlar ilave edilmeli, graf güncellenmelidir. (Yeni bulunan h-core makalelerin grafa entegre edilmesi sırasında önceki ve yeni eklenen düğümler arasındaki referans ilişkileri de genel graftan kontrol edilerek gerekli ilişki kenarları eklenmelidir.)

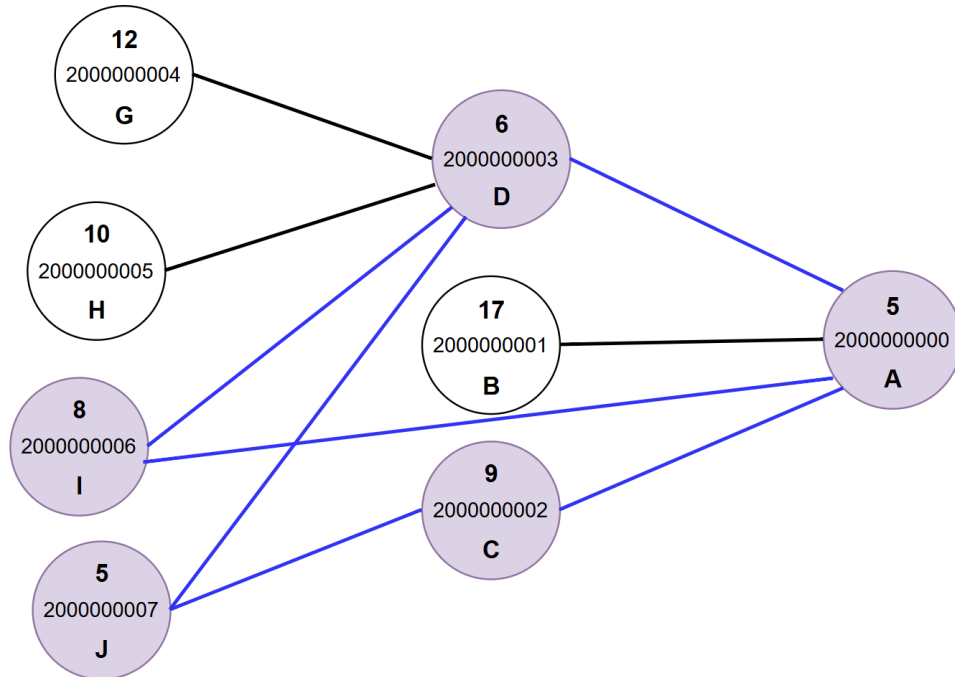


Şekil 4. 2. kademe h-core graf gösterimi

- Oluşan grafa bulunan toplam makale (düğüm) sayısı, toplam referans (siyah kenar) sayısı, toplam verilen referans sayısı, toplam alınan referans sayısı, en çok referans alan makale id ve aldığı referans sayısı, en çok referans veren makale id ve verdiği referans sayısı bilgileri arayüzde dinamik olarak gösterilecektir.

### 2.3. Graf Analiz Metriklerinin Uygulanması

- 2. adım sonunda elde edilen nihai graf yapısının tüm yönlü kenarları öncelikle yönsüz kenarlarla değiştirilerek yeniden yapılandırılmalı sonrasında ise elde edilen yönsüz kenarlardan oluşan bu graf üzerinde aşağıdaki yöntemleri uygulanmalıdır:
  - **Betweenness Centrality** : Bir düğümün graftaki diğer düğümlerin olası tüm ikili düğümleri için bulunan en kısa yollardan kaç tanesinin üzerinde bulunduğunu hesaplar. Örneğin A, B, C, D düğümlerinden A için betweenness centrality hesaplanması için B, C, D düğümlerinin tüm ikilileri (B-C, B-D, C-D) her biri için en kısa yollar bulunur bu yolların kaç tanesinde A düğümünün var olduğu bulunur.
  - **K-core Decomposition**: k-core terimi, ana grafiğin her bir düğümünün en az k derecesine sahip olduğu maksimum bir alt grafiğini ifade eder. Yani uygulandığı graftan elde edilecek alt graf öyle olmalı ki her bir düğümünün derecesi (komşu sayısı) en az k olsun.
- Grafın tüm düğümleri için betweenness centrality hesaplanarak sonuçlar arayüzde gösterilmelidir.
- Kullanıcıdan alınan k değeri için graf üzerinde k-core decomposition işlemi yapılmalı, sonrasında oluşan yeni grafın düğüm ve kenarları önceki graf üzerinde farklı renge boyanarak gösterilmelidir.



Şekil 5. k=2 için k-core decomposition sonrası graf gösterimi

## NOTLAR:

- Proje C++, C# ve Java dilleri kullanılarak gerçekleştirilecektir. Projede görselleştirme dışında hazır kütüphane kullanılmayacaktır.

## Proje Teslimi

- Rapor IEEE formatında (önceki yıllarda verilen formatta) 4 sayfa, akış diyagramı veya yalancı kod içeren, özet, giriş, yöntem, deneysel sonuçlar, sonuç ve kaynakça bölümünden oluşmalıdır. Rapor LATEX te yazılmalıdır.
- Dersin takibi projenin teslimi dahil edestek2.kocaeli.edu.tr sistemi üzerinden yapılacaktır. edestek2.kocaeli.edu.tr sitesinde belirtilen tarihten sonra getirilen projeler kabul edilmeyecektir.
- Proje ile ilgili sorular edestek2.kocaeli.edu.tr sitesindeki forum üzerinden Arş. Gör. Abdurrahman GÜN veya Arş. Gör. Muhammed Ali Irmak'a sorulabilir.
- Demo sırasında algoritma, geliştirdiğiniz kodun çeşitli kısımlarının ne amaçla yazıldığı ve geliştirme ortamı hakkında sorular sorulabilir.
- Kullandığınız herhangi bir satır kodu açıklamanız istenebilir.
- Proje grupları her öğretim kendi içinde olmak üzere en fazla iki kişiden oluşturulmalıdır.