

Yazılım Geliştirme Laboratuvarı Proje-2

Graf Renklendirme Algoritması ile

Ders Programı Oluşturma

Emirhan Kayhan

211307023
211307023@kocaeli.edu.tr

Bariş Fahri Kahrıman

211307024
211307024@kocaeli.edu.tr

Esmagül Kılınçat

211307025
211307025@kocaeli.edu.tr

I. ÖZET

Python'da yazılmış olan bu program, MySQL'deki veri tabanından derslerin hoca ve kaçınıcı sınıf olduğu verilerini işleyerek bu verilere göre bir çizge oluşturur ve dersleri ilişkilendirir. Bu ilişkilendirilen dersleri "**Greedy Coloring**" algoritması ile bu çizgedeki ders düğümlerini renklendirir. Bunun sonucunda derslerin çakışmadığı bir ders programı oluşturur.

Programda Python'un "**NetworkxX**" kütüphanesi kullanılarak dersler çizge renklendirme teorisine göre düğümler halinde tanımlanır. Sonrasında ilişkili düğümler renklendirilir. Sonrasında "**matplotlib**" çizim kütüphanesi ile bu oluşturulan çizmeyi görsel hale getirir.

Bu derslerin renklendirmelerine göre derslerin çakışma durumlarına göre gün, saat ve derslik numarası atama işlemlerini yapar. Sonrasında kodda oluşan ders_gunsaat ve derslik_no bilgilerini MySQL'deki veri tabanına kaydeder.

Anahtar kelimeler: Python, NetworkX, graph theory, greedy renklendirme algoritması, ders programı

Abstract

This program, written in Python, creates a drawing by processing the data yield in MySQL, the instructor of the courses and the grades of the courses, and separate the courses. These associated courses color the nodes of the courses in this diagram with the "Greedy Coloring" programs. As a result, a curriculum is created in which processes do not overlap.

In the program, Python's "**NetworkX**" library is used to connect courses as nodes according to the drawing coloring theory. Then the Associated nodes are colored. Then, this program visualizes the plot with the "**matplotlib**" drawing file.

According to the coloring of these courses, it assigns days, hours and classroom numbers according to the conflict situations of the courses. Thus, lesson_daytime and lesson_number created in the code are connected to the database in MySQL.

Keywords: Python, NetworkX, graph theory, greedy coloring applications, syllabus

II. GİRİŞ

A. NetworkX

NetworkX, Python programlama dilinde geliştirilen bir açık kaynaklı bir ağ analizi kütüphanesidir. Bu kütüphane, graf teorisi ve ağ analizi ile ilgili birçok işlemi gerçekleştirmek için kullanılır. Graf teorisi, düğüm (node) ve kenar (edge) adı verilen elemanlar arasındaki ilişkileri inceleyen bir matematik dalıdır. NetworkX, bu tür graf teorisine dayalı yapıları oluşturmanıza, analiz etmenize ve görselleştirmenize olanak tanır.

NetworkX'in temel özellikleri şunlardır;

- **Graf Türleri:** NetworkX, çeşitli çizge türlerini destekler, örneğin yönlendirilmiş ve yönlendirilmemiş grafikler, çoklu grafikler, ağırlıklı grafikler ve diğerleri.
- **Düğüm ve Kenar Özellikleri:** Çizgenin düğüm ve kenarlarına özel özellikler ekleyebilirsiniz. Özellikler, düğümleri veya kenarları temsil eden ana veri yapısına ek bilgiler eklemenizi sağlar.
- **Çizge Analizi:** NetworkX, çizgenin yapısını inceleme, düğümlerin derecesini hesaplama, alt graf arama, kenar kesme gibi graf analizi işlemlerini gerçekleştirmenize olanak tanır.
- **Algoritmalar:** Kütüphane, çizge teorisine dayalı bir dizi algoritma içerir. Örneğin, en kısa yol bulma, Merkezilik hesaplama, minimum kesilmiş set bulma gibi çeşitli algoritmalar mevcuttur.
- **Görselleştirme:** NetworkX, Matplotlib gibi diğer kütüphanelerle entegre çalışarak grafikleri görselleştirmenize olanak tanır. Bu sayede ağ yapınızı anlamak ve başkalarına göstermek daha kolay hale gelir.

B. Matplotlib

- Matplotlib, Python programlama dilinde geliştirilen ve bilimsel veri görselleştirmesi için kullanılan bir kütüphanedir. Matplotlib, çeşitli grafik türleri oluşturmanızı ve bu grafikleri özelleştirmenizi sağlar. Bilimsel araştırmalardan veri analizine, öğrenme uygulamalarına kadar birçok alanda

kullanılabilir. Özellikle NumPy kütüphanesi ile birlikte sıkça kullanılır.

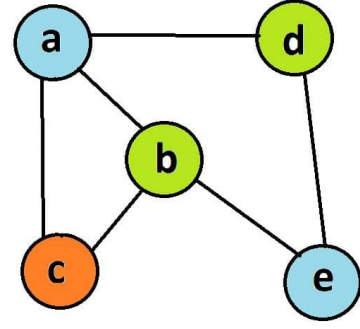
Matplotlib'in temel özellikleri şunlardır;

- **Çeşitli Grafik Türleri:** Matplotlib, çizgi grafikleri, bar grafikleri, scatter plot'lar, histogramlar, 3D grafikler, polar plot'lar gibi birçok farklı grafik türünü destekler.
- **Özelleştirme:** Grafikleri kişiselleştirmek için geniş bir özelleştirme seçeneği sunar. Renk, çizgi stilleri, etiketler, eksenlerin ölçekleri gibi birçok özelliği kontrol edebilirsiniz.
- **İnteraktif Grafikler:** Matplotlib, grafiklerinizi etkileşimli hale getirebileceğiniz bazı araçlar sunar. Örneğin, zoom yapma, grafiklerde gezinme gibi özellikleri kullanabilirsiniz.
- **TeX Desteği:** LaTeX benzeri bir dil olan TeX formatında metin girişi yapabilirsiniz. Bu özellik, özellikle akademik yazı ve raporlar için matematiksel sembollerin ve denklemlerin eklenmesini kolaylaştırır.
- **Çeşitli Çıktı Formatları:** Matplotlib ile oluşturulan grafikleri çeşitli formatlarda kaydedebilirsiniz, bu da sunumlar, makaleler veya web sayfaları gibi farklı ortamlarda kullanılabilirliği artırır.

C. Greedy Renklendirme Algoritması

Greedy renklendirme algoritması, graf teorisindeki renklendirme problemini çözmek için kullanılan bir algoritma türüdür. Renklendirme problemi, bir grafın düğümlerine, her iki komşu düğümün aynı renkte olmadığı şekilde renk atama işlemidir. Temel amacı, grafın düğümlerini minimum sayıda renk kullanarak renklendirmektir. Bu tür bir renklendirme problemi, graf teorisi, programlama yarışmaları ve çeşitli uygulamalarda karşılaşılan bir optimizasyon sorunudur.

- *Başlangıçta, hiçbir düğüm renklenmemiş olarak kabul edilir.*
- *Düğüm, bir sıra belirlenerek bu sıraya göre işlenir. Sıranın belirlenmesi genellikle düğümlerin derecelerine (kaç tane kenarı olduğuna) veya başka bir kriterle yapılabilir.*
- *Her düğüm, önceki renklenmiş düğümlerin renginden farklı bir renkle renklendirilir. Eğer tüm komşularıyla aynı renkte renklendirilebiliyorsa, o düğüme komşu renk atanır.*
- *Düğüm, sırayla işlendiğinde, grafın tüm düğümleri renklendirilmiş olur. Bkz: Şekil-1*



Şekil -1

III. GRAFIN UYGULANIŞI

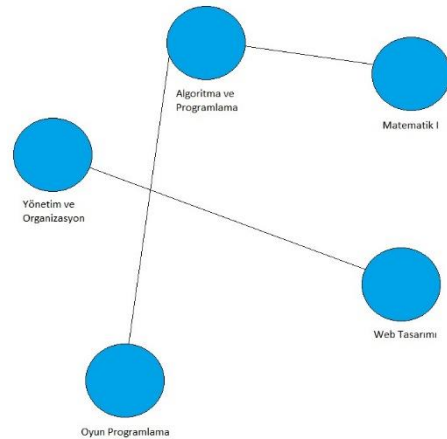
İlk olarak planlanması gereken 5 ders örneği verelim.

Algoritma ve Programlama: Hoca: Yavuz Sınıf: 1. Sınıf
Oyun Programlama: Hoca: Yavuz Sınıf: 4. Sınıf
Web Tasarımı: Hoca: Önder Sınıf: 3. Sınıf
Yönetim ve Organizasyon: Hoca: Gazi Sınıf: 3. Sınıf
Matematik I: Hoca: Çiğdem Sınıf: 1. Sınıf

Bu bilgilere göre hoca veya sınıf bilgisi çatışan dersler için çatışma tablosu oluşturuyoruz. bkz: Şekil-2

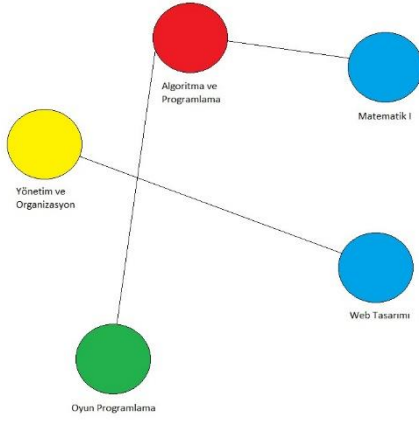
	Algoritma ve Programlama	Oyun Programlama	Web Tasarımı	Yönetim ve Organizasyon	Matematik I
Algoritma ve Programlama		X			X
Oyun Programlama	X				
Web Tasarımı				X	
Yönetim ve Organizasyon			X		
Matematik I	X				

Bu tablodaki dersleri düğümler halinde gösterip çatışan ders düğümlerini birbirine bağlıyoruz. bkz: Şekil-3



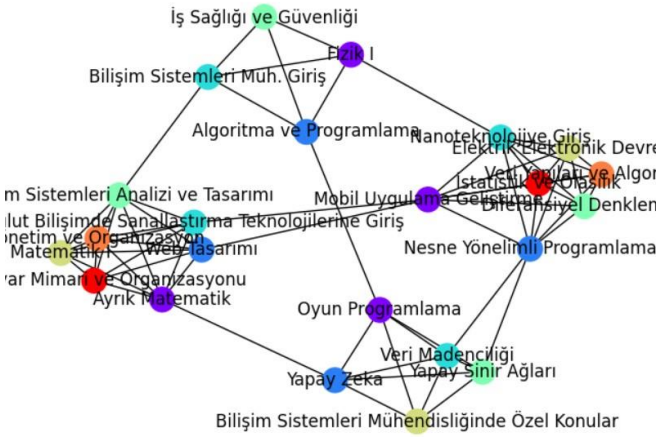
Şekil-3

Şimdi bu düğümleri bağlara göre renklendireceğiz. Önce en çok bağı olan düğüm renklendirilecek ve bağı 2 düğüm birbirinden farklı renkte olacak. bkz: Şekil-4



Şekil-4

Koddaki esas çizge çıktısı: bkz: Şekil-5



Şekil-5

IV. DERSLERE GÜN, SAAT, DERSLİK ATANMASI

Çizgede düğüm olarak tanımlanan dersler, çatışma durumuna göre renklendirildikten sonra derslerin gün saat ve derslik bilgisinin çatışan dersleri aynı gün saate koymayacak şekilde gün saat derslik bilgilerini atamamız gerekiyor. Farklı renkteki derslere farklı gün saat bilgisi atayacak şekilde Python kodu ve sorguları hazırlandı. Sonrasında bu bilgiler, MySQL'deki dersprogrami veri tabanına girildi ve ders programı oluşturuldu. bkz: Şekil-6

ders_id	ders_adi	ders_gunsaat	derslik_no
101	Algoritma ve Programlama	Çarşamba 12:00	1040
102	Matematik I	Perşembe 12:00	1044
103	Ayrık Matematik	Cuma 13:00	1050
104	Fizik I	Perşembe 13:00	1040
105	Web Tasarımı	Pazartesi 13:00	1041
106	Mobil Uygulama Geliştirme	Salı 14:00	1036
107	Yönetim ve Organizasyon	Salı 13:00	1041
108	İş Sağlığı ve Güvenliği	Salı 15:00	1041
109	Bilgisayar Mimari ve Organizasyonu	Cuma 10:00	2023
110	Diferansiyel Denklemler	Cuma 12:00	1044
111	Bulut Bilişimde Sanallaştırma Teknolojilerine Giriş	Cuma 11:00	1036
112	Oyun Programlama	Pazartesi 13:00	1050
113	Bilişim Sistemleri Mühendisliğinde Özel Konular	Cuma 15:00	1040

Şekil-6

REFERANSLAR

- [1] <https://networkx.org/>
- [2] https://www.academia.edu/2790141/STUDENT_TIME_TABLE_BY_USING_GRAPH_COLORING_ALGORITHM
- [3] <https://towardsdatascience.com/graph-coloring-with-networkx-88c45f09b8f4>
- [4] <https://consultanubhav-1596.medium.com/time-table-generation-via-graph-colouring-algorithm-b4f16bff7ca7>
- [5] <https://www.youtube.com/watch?v=VetBkjc9Go&t=28s>
- [6] <https://www.youtube.com/watch?v=rldK11CNx-A&list=PLGZqdNxqKzfYXTwYAZIlmjnQmrvtCSR1J>