

Yazılım Geliştirme Labratuvarı-1

Ders Programı

Emirhan Kayhan¹

Bilişim Sistemleri Mühendisliği

¹211307023

211307023@kocaeli.edu.tr

Barış Fahri Kahrıman²

Bilişim Sistemleri Mühendisliği

²211307024

211307024@kocaeli.edu.tr

Esmagül Kılınçat³

Bilişim Sistemleri Mühendisliği

³211307025

211307025@kocaeli.edu.tr

I. ÖZET

Yazılım Geliştirme Laboratuvarı-I kapsamında gerçekleştirilen 2. proje, ders programı oluşturmayı hedefleyen kapsamlı bir geliştirme sürecini içermektedir. Bu projede, öğrencilerin programlarını daha etkili bir şekilde düzenlemelerine yardımcı olacak bir çizge renklendirme algoritması olarak "Greedy graph coloring" algoritması tercih edilmiştir.

Projenin temel amacı, ders programlarını optimize etmek ve çakışmaları en aza indirmek için bir yazılım aracı sağlamaktır. Bu bağlamda, "Greedy graph coloring" algoritması, graf teorisinde sıkça kullanılan bir renklendirme algoritmasıdır. Bu algoritma, çizgedeki komşu düğümlere farklı renkler atanarak çakışmaları önlemeye çalışır. Bu sayede, ders programının düzenlenmesindeki karmaşıklık azaltarak daha düzenli ve kullanıcı dostu bir program elde etmeyi hedefler.

Projede kullanılan "Greedy graph coloring" algoritması, öğrencilere graf teorisinin temel kavramlarına ve algoritmalarına dair derinlemesine bir anlayış kazandırmaktadır. Aynı zamanda, algoritmanın gerçek dünya problemlerine nasıl uygulanabileceği konusunda da pratik bir deneyim sunmaktadır.

Proje sürecinde öğrenciler, yazılım geliştirme pratiğiyle birlikte algoritma tasarımı ve analizi konularında da becerilerini geliştireceklerdir. Bu proje, öğrencilere hem yazılım geliştirme hem de algoritmik düşünme konularında kapsamlı bir deneyim sunarak gelecekteki projelerine daha güvenli bir şekilde yaklaşmalarına yardımcı olacaktır.

Anahtar kelimeler: Greedy ,graph coloring,graph theory

Abstract

The second project carried out within the scope of Software Development Laboratory-I includes a comprehensive development process aimed at creating a course curriculum. In this project, the "Greedy graph coloring" algorithm was preferred as a graph coloring algorithm that will help students organize their programs more effectively.

The main aim of the project is to provide a software tool to optimize course schedules and minimize conflicts. In this context, the "Greedy graph coloring" algorithm is a coloring algorithm frequently used in graph theory. This algorithm tries to avoid conflicts by assigning different colors to neighboring nodes in the graph. In this way, it aims to obtain

a more organized and user-friendly program by reducing the complexity of organizing the course program.

The "Greedy graph coloring" algorithm used in the project gives students an in-depth understanding of the basic concepts and algorithms of graph theory. It also provides practical experience on how the algorithm can be applied to real-world problems.

During the project, students will improve their skills in algorithm design and analysis along with software development practice. This project will provide students with comprehensive experience in both software development and algorithmic thinking, helping them approach their future projects with more confidence.

Keywords: Greedy ,graph coloring,graph theory

II. GİRİŞ

A. XAMPP

- XAMPP, Apache, MySQL, PHP ve Perl gibi popüler açık kaynaklı web geliştirme araçlarını içeren bir pakettir.
- İsmi, X (cross-platform), Apache, MySQL, PHP, Perl'in baş harflerinden gelir.
- XAMPP, web uygulamaları geliştirmek ve test etmek için kullanılan bir hızlı geliştirme ortamı sağlar.

B. Apache HTTP Server

- Apache, açık kaynaklı bir web sunucu yazılımıdır.
- Web sunucusu olarak kullanılır ve istemcilere HTML dosyaları gibi statik içerikleri sunmanın yanı sıra, dinamik içerikleri işleyerek PHP, Python, Perl gibi programlama dilleri ile uyumlu çalışabilir.
- Modüler tasarımı sayesinde genişletilebilir ve farklı modüllerle özelleştirilebilir.

C. HTML (Hypertext Markup Language)

- HTML, web sayfalarını oluşturmak için kullanılan standart işaretleme dilidir.
- Metin, bağlantılar, görseller ve diğer medya öğelerini bir araya getirerek web sayfalarının yapısını tanımlar.
- Tarayıcılar, HTML kodunu okuyarak web sayfalarını kullanıcıya gösterir.

D. PHP (Hypertext Preprocessor):

- PHP, sunucu taraflı bir programlama dilidir ve özellikle web geliştirmek için kullanılır.
- Dinamik içerik oluşturmak, veri tabanı işlemleri yapmak, form verilerini işlemek gibi görevlerde kullanılır.
- PHP kodları genellikle HTML içerisine gömülerek kullanılır.

E. MySQL Workbench:

- MySQL Workbench, MySQL veri tabanı yönetim sistemleri için bir grafik kullanıcı arayüzü (GUI) sağlayan bir araçtır.
- Veri tabanı tasarımı, sorgu oluşturma, yönetim ve izleme işlemlerini kolaylaştırır.
- Görsel arayüzü, veri tabanı geliştirme ve yönetim süreçlerini daha kullanıcı dostu hale getirir.

Bu araçlar, web geliştirme süreçlerinde sıkça kullanılan temel bileşenlerdir. Apache web sunucusu, XAMPP geliştirme ortamı, HTML ve PHP front-end ve back-end tarafında kullanılırken, MySQL Workbench ise veri tabanı yönetimini kolaylaştırır.

III. UYGULAMA ÖZELLİKLERİ

A. Greedy Graph Coloring

"Greedy Graph Coloring" algoritması, bir grafın düğümlerini belirli bir kurala göre renklendirmek üzere tasarlanmıştır. Algoritma, her adımda henüz renklendirilmemiş bir düğümü seçip, çakışmaları önleyerek bir renk atar. Bu adım adım ilerleyen strateji, her düğümü en iyi şekilde renklendirmeyi amaçlar. Greedy graph coloring, pratik uygulamalarda kullanım kolaylığı ve etkililiği nedeniyle tercih edilen bir algoritma olup, genellikle karmaşıklığı azaltmak ve hızlı çözümler elde etmek için kullanılır. Ancak, her durumda en optimal çözümü garanti etmese de birçok durumda başarılı sonuçlar elde edebilir.

IV. GELİŞTİRME SÜRECİ

Projemiz kapsamında MySQL veri tabanında "Dersler", "Hocalar", ve "Hoca_ders" adlı üç tabloyu başarıyla oluşturduk. Bu tabloları temel verilerle doldurarak gerekli kısıtlamaları ekledik. Veri tabanı oluşturma sürecinden sonra proje gelişimine geçtik.

XAMPP kullanarak Apache üzerinden yerel bir sunucu oluşturduk. Ardından, bir PHP dosyası hazırlayarak bu dosyayı yerel sunucu üzerinde başarıyla çalıştırdık. Bu adım, projemizin web tabanlı olacağına dair ilk adımdı.

PHP dosyamızı çalıştırmayı başardıktan sonra, veri tabanından veri çekme işlemi üzerinde çalışmaya başladık. Veri çekme işlemi başarılı olduğunda, HTML kullanarak bir index sayfası oluşturduk. Bu sayfa, gün ve saat bilgilerini içeren bir ders programı tablosunu içeriyordu.

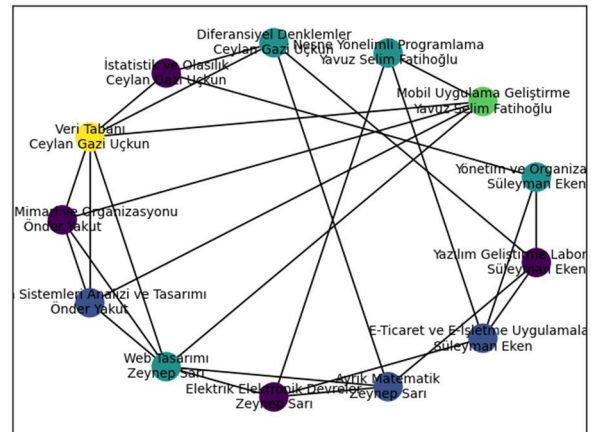
Ders programının ana yapısını oluşturduktan sonra, temel algoritmayı kurmaya geçtik. Graf teorisi ve renklendirme yöntemlerinden biri olan "Greedy" algoritmasını kullanarak PHP ile özgün bir algoritma geliştirdik. Bu algoritma, belirlediğimiz kısıtlamalar doğrultusunda derslerin çakışmalarını önlemek amacıyla renklendirme yöntemini uyguluyor. Temel olarak "hoca adı" ve "sınıf" verilerini baz alan bu algoritma, çakışmaları bu iki değerin karşılaştırılması üzerinden tespit edip ilgili şekilde renklendirme yöntemini uyguluyor. Algoritma, daha sonrasında renklere bağlı olarak ders programındaki hücrelere dersleri atıyor.

Ek olarak bir veri düzenleme sayfası oluşturduk. Ana sayfadan buton aracılığıyla ulaşabildiğimiz bu sayfa, ders düzenleme uygulamasını oluşturan temel bir yapı sunar. Veritabanı bağlantısı, yeni ders ekleme, ders güncelleme, ders silme işlemlerini gerçekleştirmeyi sağlar. Veritabanından alınan dersler ve hoca adları, HTML tablosu içinde görüntülenir. JavaScript ve modaller kullanılarak kullanıcı dostu bir arayüz sağlanır.

Ana sayfa kısmına eklediğimiz bir diğer buton olan "ders programını onayla" butonu ise oluşturduğumuz ders programını pdf dosyasına çevirerek kullanıcıya bu programı kaydetme imkanını sunuyor.

Bu adımların sonucunda projemizde, veri tabanından veri çekme, HTML arayüz tasarlama ve özgün bir algoritma geliştirme başarıyla tamamlandı. Geliştirilen algoritma, ders programının daha düzenli ve çakışmasız olmasını sağlamak için etkili bir çözüm sunmaktadır.

Bunlara ek olarak bir de Python dili ve "networkx", "matplotlib" gibi yardımcı kütüphaneler kullanarak bir dosya oluşturduk. Bu Python dosyası temel olarak index.php dosyamıza benzer yapıda. Index dosyamızda kurduğumuz algoritma ve veri işlerini gerçekleştirip, yardımcı kütüphaneler ile birlikte bize oluşturduğumuz algoritmayı görsel graf figür aracılığıyla görüntülememizi sağlıyor.



Örnek figür

REFERENCES

- [1] Anubhav Sharma, "Time Table Generation via Graph Colouring Algorithm," Medium, [Online]. Available: <https://consultanubhav-1596.medium.com/time-table-generation-via-graph-colouring-algorithm-b4f16bff7ca7>
- [2] Muthu Krishnan, "Graph Coloring with NetworkX," Towards Data Science, [Online]. Available: <https://towardsdatascience.com/graph-coloring-with-networkx-88c45f09b8f4>
- [3] "NetworkX Documentation," [Online]. Available: <https://networkx.org/>
- [4] "D3 Graph Theory," [Online]. Available: <https://d3gt.com/index.html>
- [5] A. S. Kumar and M. Srinivas, "Student Time Table by Using Graph Coloring Algorithm," Academia.edu, [Online]. Available: https://www.academia.edu/2790141/STUDENT_TIME_TABLE_BY_USING_GRAPH_COLORING_ALGORITHM
- [6] <https://www.apachefriends.org/tr/index.html>