

Manisa Celal Bayar Üniversitesi Yazılım Mühendisliği Bölümü
YZM 2116- Veri Yapıları Dersi

Proje#2

“İkili Arama Ağacı, Heap, Hash Tabloları ve Çizgeler”

Bahar 2018

Veriliş Tarihi: 24.04.2018

Son Teslim Tarihi: 25.05.2018

Proje Teslimi: cbuyzm2116@gmail.com

Projeyle İlgili Genel Kurallar:

1. Projeden tam puan alınabilmesi için, proje metninde yer alan, **Ödev#1** ve **Ödev#2**'nin yapılması gerekmektedir.
2. Proje, **iki** veya **üç kişilik** gruplar halinde yapılabilir. Zorunlu haller dışında, birinci projedeki ekip üyeleri ile devam edilmesi beklenmektedir.
3. Proje, dersi veren öğretim üyesine e-posta yolu ile (cbuyzm2116@gmail.com adresine) teslim edilecektir.
4. **Son teslim tarihinden sonra** teslim edilen projeler **değerlendirmeye alınmayacaktır.**
5. Farklı grupların teslim ettikleri projeler arasında kopya kontrolü yapılacak olup kopya tespit edilen tüm grupların, **dönem sonu proje notları** **“sıfır”** olarak değerlendirilecektir.
6. **Rapor içermeyen ödev teslimleri değerlendirmeye alınmayacaktır.**
7. Proje metninde yer alan, her iki ödev için **ayrı birer rapor** hazırlanmalıdır. Proje raporları, her iki programın, **açıklama satırları içeren kaynak kodları** ile birlikte, belirtilen son teslim tarihinde gönderilmelidir. Proje için, yazıcı çıktısı istenmemektedir.

Rapor aşağıda belirtilen ilkeler doğrultusunda hazırlanmalıdır:

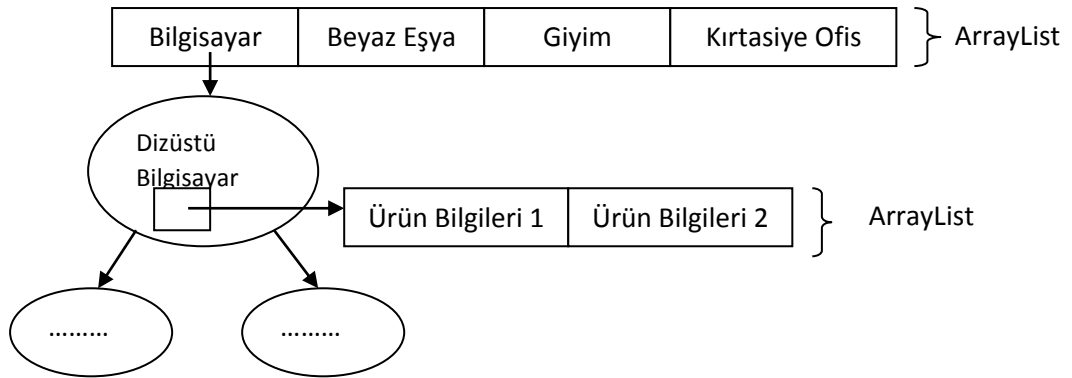
- **Kapak:** Ödev numarası ve adı, Proje grubundaki öğrenci numaraları ve ad-soyadları ile proje teslim tarihi bilgilerini içermelidir.
- **Rapor İçeriği:**
 - i. **Bölüm#1:** Gerçekleştirilen Platform ve Dil (Java, C#, Eclipse, vs.) ve Sürüm Adı
 - ii. **Bölüm#2:** Gerçekleştirimi Yapılan Problemin Kısa Tanımı (Ödev metninden özetlenebilir.)
 - iii. **Bölüm#3:** Veri Yapısı Kataloğu (Kullanılan veri yapılarının, sınıfların ve metotların kısa açıklamaları)
 - iv. **Bölüm#4:** Yazılım Geliştirme İçin Harcanan Süreler (Grup üyeleri için kişi ve saat bazında)
 - v. **Bölüm#5:** Elde Edilen Örnek Sonuçlar (Bu bölüm ekran görüntüleri ve farklı durumlarda kodun çalıştığına ilişkin destekleyici kanıtlar içermelidir.)
 - vi. **Bölüm#6:** Proje Gerçekleştirmede Yararlanılan Kaynaklar (Kod parçalarının alındığı ya da gerçekleştirmede yararlanılan başlıca kaynakların listelenmesi beklenmektedir.)

Ödev#1

“Sanal Market Benzetimi”

Ödev Metni:

Geliştirilecek bir sanal market yazılımı kapsamında, ürünlere ilişkin bilgiler kategori ağacında tutulacaktır. Her bir elemanı, Bilgisayar, Beyaz Eşya gibi kategorilerden biri olan ArrayList içerisinde ilgili kategorideki ürünleri tutan ağaçlar yer alacaktır. Ürünler, ürün adına göre, ilgili kategorideki **ikili arama ağacı (binary search tree)** veri yapısı üzerinde tutulacaktır:



- Ağacın her düğümünde, Ürünün adı (örnek olarak Dizüstü Bilgisayar) ve ilgili ürün adındaki tüm marka ve modellerin listesi (ArrayList veya daha ileri bir veri yapısında) tutulmalıdır. Listenin her bir elemanında ürüne ilişkin bilgileri (Marka, Model, Miktarı, Maliyeti, Satış Fiyatı, Ürün Açıklaması) içerecektir. **Bu gerçekleştirim, ikili arama ağacı tabanlı yapılacaktır.**
- Ürün açıklamalarında geçen kelimelere ilişkin listenin** (sözcüğün) de, **Hash Tablosu üzerinde** oluşturulması gerekmektedir. Hash tablosunda her bir kelime için, bir dinamik dizi (ArrayList) tutulmalıdır. Örnek olarak, Ürün Açıklaması sahasında “USB” kelimesi geçen Ürünlerin referans (adres) listesi, Hash Tablosu’nun ilgili sahasında ArrayList olarak tutulmalıdır. Bu şekilde bir kullanıcı (müşteri), herhangi bir kelimenin geçtiği ürünlere daha hızlı erişebilecektir. İkili arama ağacını dolaşmadan doğrudan kelime ile ilgili ürünlerin bilgilerine ulaşabilecektir.
- Verilen bir kategorideki, örnek olarak Bilgisayar kategorisindeki ürünler Satış Fiyatına göre ayrı bir Heap’te (Yığın) yani ağaç şeklindeki **öncelik kuyruğunda küçükten büyüğe** sıralı tutulacaktır.

Yapılacak İşlemler:

1) Verilerin hazırlanması:

Market.dat adlı dosyayı kendiniz hazırlayabilirsiniz. İlk bilgiler dosya yerine bellekten veya klavyeden de alınabilir (dosya kullanımı seçimlik). Kategoriler, Bilgisayar, Beyaz Eşya, Giyim, Kırtasiye_Ofis, YapıMarket, Bahçe, Tekstil, Yiyecek olarak düzenlenebilir. Diğer bilgileri de İnternet üzerinden Sanal marketlerden toplayabilirsiniz.

2) Verileri kullanarak Veri Yapılarının (Kategori tabanlı Binary Search Tree listesi, Hash Table ve Heap) belirtildiği şekilde oluşturulması.

Bu bilgilere göre 3. ve 4. adımlardaki işlemleri yapan programın yazılması

3) Market Personelinin Kullanacağı Modülün Geliştirilmesi:

- Markete yeni isimde ve/veya kategoride ürün girişi
- Markete, yeni bir marka/modelde ürün girişi
- Adından Ürün arama ve silme
- Ürün bilgilerinde değişiklik (Kategori, Marka, Model, Miktar, Fiyat)
- Ayrıca şirketin toplam gelir, gider ve kârının hesaplanması istenmektedir (Kar = o ana kadar satışı yapılan ürünlerin tümü için [Satış Fiyatı – Maliyet] toplamı)

4) Müşterilerin Kullanacağı Modülün Geliştirilmesi:

- Adından Ürün bilgisi arama (tüm markalardaki modellerinin fiyatlarını listeleme)
- Belirli fiyatlar arasındaki tüm ürünlerin bulunup listelenmesi
- **Belirtilen bir kategorideki tüm Ürünleri, ikili arama ağacındaki** düzeyleri ile birlikte listeleme (Inorder, preorder, postorder). Ağacın derinliğini ve eleman sayısını yazdırma.
- Ürün Siparişi ve Alımı
- Verilen bir kelimenin geçtiği ürünlere Hash Tablosu üzerinden erişilerek listelenmesi
- Heap oluşturularak, kullanıcının vereceği kategorideki en ucuz N adet ürünün satın alınması. İkili arama ağacı ve Hash Tablosu üzerinde de gerekli güncellemelerin yapılması.
- Kategori ağaçlarını dengeleme.
- Müşterilerin üye olabilmesi (Sistem, birden çok müşteri için düzgün biçimde çalışabilmeli).
- Alışveriş listesi eklenmesi (Belirli bir müşterinin almış olduğu ürünlere ilişkin listenin tutulması).
- Müşterilerin ürünler hakkında yorum yazabilmesi.

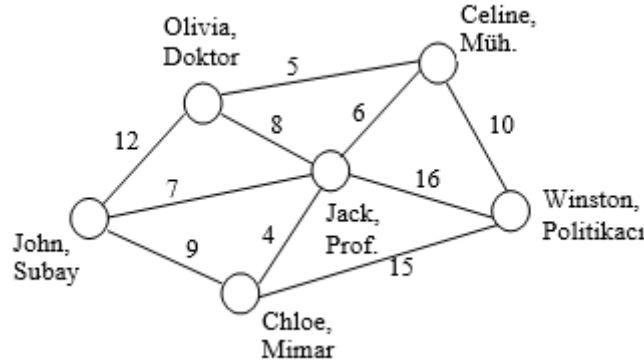
Ödev#2

“Çizge Algoritmaları ve Sosyal Ağlar”

Ödev Metni:

Aşağıdaki **yöntemlerin kodlarını ağırlıklı (weighted) yönsüz çizgeler** için yazınız ve bir çizge (graph) aduzayı paketi oluşturunuz, arayüzü form kullanarak sağlayınız:

1. **Dijkstra’nın En kısa yol problemi (SP):** Kullanıcıdan alınan bir köşe çifti arasındaki **en kısa yolun bulunması**.
2. **Genişlik Öncelikli Dolaşma problemi (BFS):** Kullanıcıdan alınan bir köşe numarasından başlanılarak, çizgenin genişlik öncelikli olarak dolaşılması.
3. **Prim Algoritması Tabanlı En Küçük Kapsayan Ağaç problemi (Prim-MST):** Çizgenin en küçük kapsayan ağacının (MST) ve maliyetinin Prim algoritmasına dayalı olarak bulunması.
4. **Kruskal Algoritması Tabanlı En Küçük Kapsayan Ağaç problemi (Prim-MST):** Çizgenin en küçük kapsayan ağacının (MST) ve maliyetinin Kruskal algoritmasına dayalı olarak bulunması.
5. **Edmonds Karp Algoritması Tabanlı Maksimum Akış problemi (EK-MAX):** Belirli bir çizge üzerinde, maksimum akış problemin, Edmonds-Karp algoritmasına dayalı olarak çözülmesi.
6. **Ford Fulkerson Algoritması Tabanlı Maksimum Akış problemi (EK-MAX):** Belirli bir çizge üzerinde, maksimum akış problemin, Ford Fulkerson algoritmasına dayalı olarak çözülmesi.
7. **Sosyal Ağ:** Aşağıdaki şekilde bir sosyal ağ verilmiştir. Belirtilen sosyal ağ, çizge yapısı şeklinde oluşturunuz ve uygun görselleştirme araçlarını kullanarak çizdiriniz:



Çizgenin kenarlarındaki ağırlıklar, kişilerin birbirlerinin çağrılarına ne kadar süre sonra yanıt verdiklerini temsil etmektedir. Bağlantı olmayan kişilerin, birbirlerini doğrudan tanımadıklarını varsayabilirsiniz. **Seçeneklerdeki işlemler belirtilen sosyal ağ için yapılmalıdır:**

- i) Yukarıdaki sosyal ağ üzerinde, en kısa yol problemi, genişlik öncelikli arama ve en küçük kapsayan ağaç problemlerinin hangi amaçlar için kullanılabileceğini raporunuzda kısaca yanıtlayınız.

- ii) Verilen çizgede, verilen bir kişinin en kısa sürede doktor olan kişiye ulaşması için etkin bir algoritma gerçekleştirimi yapınız. Yapılan gerçekleştirimde kaç arama gerektiğini belirleyiniz. Raporunuzda algoritmanızın etkinlik sınıfını belirtiniz.