

Dersin Adı : Algoritma ve Programlama - II
Ödev - 4 : Öğrenci-Ders Takip Sistemi (ver. 2)
Verilme Tarihi : 29.04.2009 Çarşamba
Teslim Tarihi : 06.05.2009 Çarşamba, Saat 23:59 (kaynak kod)

Bir lisede okuyan öğrencilerin ve aldıkları derslerin takip edilmesini sağlayacak bir uygulama geliştirilmesi istenmektedir. Bu amaçla öğrencilere ilişkin aşağıda verilen bilgiler tutulacaktır:

Numarası : 4 basamaklı tamsayı
Adı : 10 karakter (sadece İngilizce'de yer alan harfleri içerecektir.)
Soyadı : 15 karakter
Sınıf adı : 2 karakter (1A, 2D, 3E, 4C v.b.)

Tek düğüm (node) yapısında tutulacak bu bilgilere, iki farklı hashing yöntemi ile erişilecektir. Bu yöntemlerden ilkinde, öğrenci numarasına bir hash fonksiyonu¹ uygulanarak düğümlere erişilecektir. Bu hash fonksiyonu ile aynı hash değerini üreten öğrencilere ait düğümler, tek-bağlı listeler (çakışma listeleri) kullanılarak öğrenci numarasına göre artan (ascending) sırada tutulacak ve bu listelerin ilk düğümlerinin adresleri ise 500 elemanlı bir işaretçi dizisinde tutulacaktır. İkinci yöntemde ise, öğrenci adına bir hash fonksiyonu² uygulanarak düğümlere erişilecektir. Bu hash fonksiyonu ile aynı hash değerini üreten öğrencilere ait düğümler, liste_baş (head) düğümüne sahip dairesel (circular) tek-bağlı listeler (çakışma listeleri) kullanılarak öğrenci adına göre alfabetik sırada tutulacak ve bu listelerin liste_baş düğümleri ise 26 elemanlı bir dizide tutulacaktır.

Ayrıca öğrencilerin o zamana kadar almış oldukları derslere ilişkin aşağıda verilen bilgiler de, aynı öğrenciye ait bilgiler nota göre azalan (descending) sırada olacak şekilde tek-bağlı listelerde tutulacaktır. Bu listelerin ilk düğümlerinin adreslerine ise, öğrenci bilgilerinin tutulduğu düğümler aracılığıyla erişilecektir.

Ders kodu : Tamsayı
Notu : Tamsayı

Buna göre, aşağıda listelenen isteklerin bir menü aracılığıyla gerçekleştirilmesi istenmektedir:

1) Yeni bir öğrencinin eklenmesi

Önce öğrencinin numarası kullanıcıdan alınmalı ve bu numaralı bir öğrenci varsa ekleme yapılmamalıdır, yoksa öğrencinin diğer bilgileri kullanıcıdan alınarak her iki çakışma listesine de (numaraya göre ve ada göre) ekleme işlemi gerçekleştirilmelidir. Aynı ada sahip öğrenciler olabilecektir, bunların kendi içinde sıralanmasına gerek yoktur.

2) Bir öğrencinin almış olduğu derslerin eklenmesi

Önce öğrencinin numarası kullanıcıdan alınmalı ve öğrenci varsa almış olduğu bir derse ilişkin ders kodu ve notu verileri kullanıcıdan alınarak ekleme işlemi gerçekleştirilmelidir. Kullanıcı isterse o öğrencinin almış olduğu diğer dersleri eklemeye devam edebilmelidir.

3) Bir öğrencinin sınıfının güncellenmesi

Önce öğrencinin numarası kullanıcıdan alınmalı ve öğrenci varsa sınıf adı güncellenmelidir.

¹ hash_değeri = öğrenci_no / 20

² hash_değeri = ingiliz_alfabesindeki_sırası (ilk_harf (öğrenci_adi))

4) Bir öğrencinin almış olduğu bir dersin notunun güncellenmesi

Önce öğrencinin numarası kullanıcıdan alınmalı, öğrenci varsa ders kodu kullanıcıdan alınmalı ve öğrenci bu dersi daha önce almışsa bu derse ait notu güncellenmelidir.

5) Bir öğrencinin silinmesi

Önce öğrencinin numarası kullanıcıdan alınmalı, öğrenci varsa o öğrenciye ilişkin bilgiler ve o öğrencinin almış olduğu tüm derslere ilişkin bilgiler (varsa) silinmelidir.

6) Bir öğrencinin bilgilerinin ve almış olduğu derslerin listelenmesi

Önce öğrencinin numarası kullanıcıdan alınmalı, öğrenci varsa öğrenciye ilişkin bilgiler, nota göre azalan sırada olacak şekilde o öğrencinin almış olduğu tüm derslere ilişkin bilgiler (varsa) ve öğrencinin aritmetik not ortalaması aşağıdaki şekilde listelenmelidir:

Öğr No	Ad	Soyad	Sınıf
-----	-----	-----	-----

Almış Olduğu Dersler:

Ders Kodu	Notu
-----	----

...	...
...	...

Öğrencinin aritmetik not ortalaması: ##.##

7) Tüm öğrencilerin, numara sırasına göre listelenmesi

Tüm öğrencilerin bilgileri, öğrenci numarasına göre artan sırada aşağıdaki şekilde listelenmelidir:

Öğr No	Ad	Soyad	Sınıf
-----	-----	-----	-----

8) Tüm öğrencilerin, ad sırasına göre listelenmesi

Tüm öğrencilerin bilgileri, öğrenci adına göre alfabetik sırada aşağıdaki şekilde listelenmelidir:

Ad	Soyad	Öğr No	Sınıf
-----	-----	-----	-----

9) Adı belirli bir harf ile başlayan öğrencilerin listelenmesi

Kullanıcıdan bir harf alınarak, adı bu harf ile başlayan tüm öğrencilerin bilgileri aşağıdaki şekilde listelenmelidir:

Ad	Soyad	Öğr No	Sınıf
-----	-----	-----	-----

10) Bir sınıfta okuyan öğrencilerin listelenmesi

Kullanıcıdan öğrencilerini görmek istediği sınıfın adı alınarak, bu sınıfta okuyan tüm öğrencilerin bilgileri, almış oldukları ders sayıları ve aritmetik not ortalamaları; öğrenci numarasına göre artan sırada aşağıdaki şekilde listelenmelidir:

Öğr No	Ad	Soyad	Sınıf	Ders Sayısı	Not Ort
-----	-----	-----	-----	-----	-----

11) Bir dersi almış olan öğrencilerin listelenmesi

Kullanıcıdan dersin kodu alınarak, bu dersi almış olan tüm öğrencilerin bilgileri ve bu dersten aldıkları notları öğrenci numarasına göre artan sırada olacak şekilde ve dersin genel not ortalaması aşağıdaki şekilde listelenmelidir:

Öğr No	Ad	Soyad	Notu
-----	-----	-----	----

Dersin genel not ortalaması: ##.##

12) Hiç ders almamış olan öğrencilerin listelenmesi

O zamana kadar hiç ders almamış olan öğrencilerin bilgileri, öğrenci adına göre alfabetik sırada aşağıdaki şekilde listelenmelidir:

Ad	Soyad	Öğr No
-----	-----	-----

Notlar:

1. Veri girişlerinin doğru bir şekilde yapılacağını varsayınız, hata kontrolü yapmayınız.
2. Tekrar eden işlemler varsa, fonksiyon kullanımına dikkat ediniz.
3. Programda kullanılacak veri yapılarının herhangi bir zamandaki örnek bir görünümü ekte verilmiştir.

DİKKAT EDİLMESİ GEREKEN NOKTALAR:

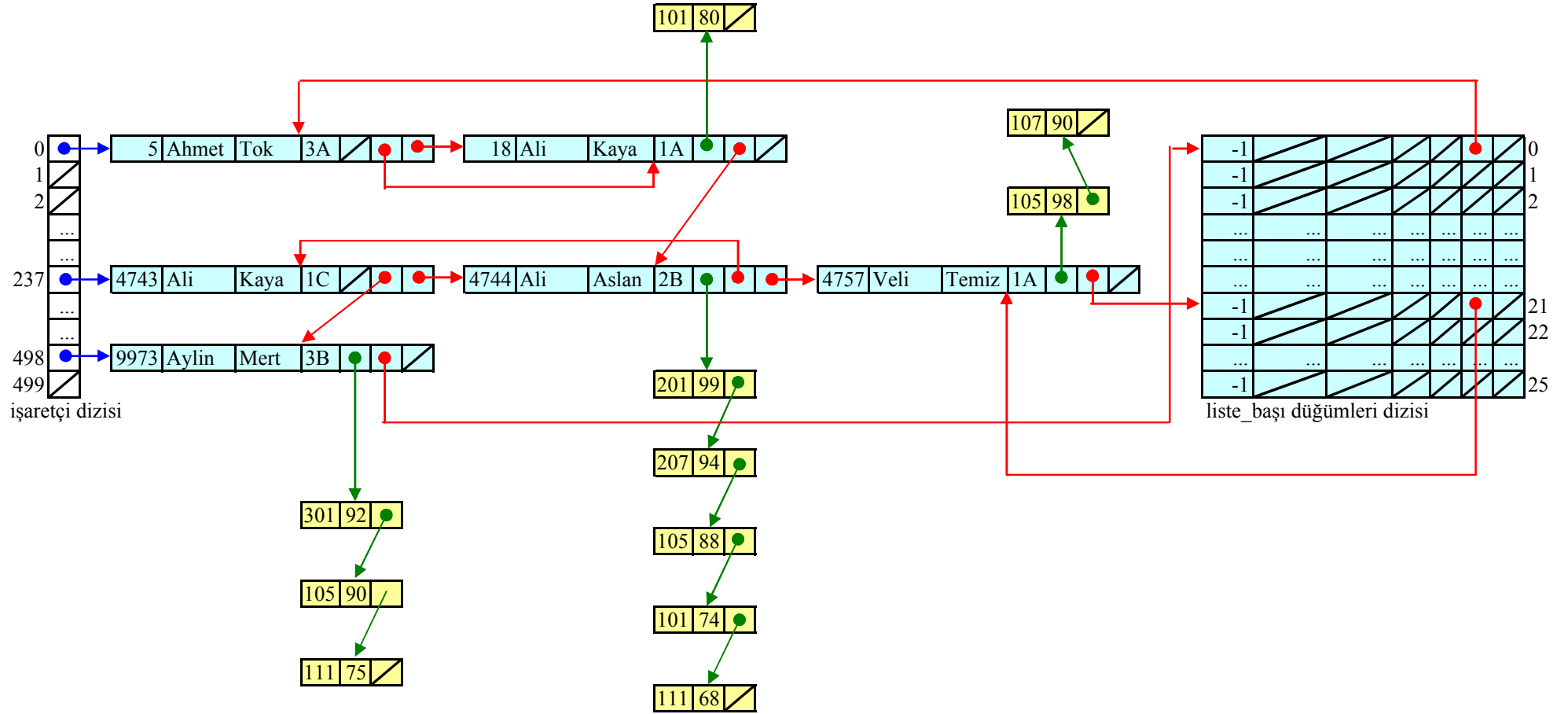
Ödevin Teslim Edilmesi:

1. Ödevin sadece kaynak kod dosyası (.c uzantılı), dosya adı öğrenci numarasının son 5 rakamı olacak şekilde (örneğin 04219.c), elektronik posta ile ozgur.gumush@gmail.com adresine dosya eki (attachment) olarak gönderilmelidir.
2. Mesajın konusu BIM112-ODEV4 olmalıdır.
3. Son teslim tarihi geçmediği sürece ödevin tekrar gönderilmesinde bir sakınca yoktur, en son gönderilen ödev değerlendirilecektir.
4. Ödev tesliminde en fazla 2 günlük gecikmeler kabul edilecek, ancak son teslim tarihinden sonraki her gün için ödev notunda %20 kesinti yapılacaktır.

Ödevin Değerlendirilmesi:

1. Ödevin değerlendirmesinde; programın doğru ve eksiksiz çalışmasının yanında etkinlik, yapısal ve modüler programlama ilkelerine uygunluk ta dikkate alınacaktır. Buna göre puanlama şu şekildedir:
 - Etkinlik (gereksiz işlemlerden kaçınma): 10 puan
 - Fonksiyon kullanımı: 15 puan
 - Doğru çalışma: 75 puan
2. Kopya çekildiği tespit edildiğinde, çeken ve çektiren kişiler ödevden sıfır alacaktır.

Veri Yapılarının Herhangi Bir Zamanaki Örnek Bir Görünümü



- : Öğrenci yapısındaki (öğr_no, ad, soyad, sınıf, ilk_ders_notu, ada_göre_sonraki, noya_göre_sonraki) düğümleri göstermektedir.
- : Ders_notu yapısındaki (ders_kodu, not, sonraki) düğümleri göstermektedir.
- : Öğrenci numarasına göre artan sırada olan tek-bağlı listelerin bağlarını göstermektedir.
- : Öğrenci adına göre alfabetik sırada olan dairesel tek-bağlı listelerin bağlarını göstermektedir.
- : Nota göre azalan sırada olan tek-bağlı listelerin bağlarını göstermektedir.
- : Değeri NULL olan string ve işaretçileri göstermektedir.