

BLM 2002 ÖDEV3

Bağlantılı listeler ile ilgili aşağıda verilen metotlar yazılmalıdır. Tüm durumlarda başlangıç durum analizi yapılmalıdır. Yani listelerin boş vb durumları kontrol edilmelidir. Ayrıca tüm metotların O notasyonuna göre çalışma zamanı analizi yapılmalı ve rapor olarak sunulmalıdır.

- İki bağlantılı listeyi birleştirmek için bir metot yazın. Verilen listeler $l1 = (2, 3, 1)$ ve $l2 = (4, 5)$, concatenate($l1, l2$)'den döndükten sonra $l1$ listesi $l1 = (2, 3, 1, 4, 5)$ olarak değiştirilmelidir.
- Sıralanmış bir bağlantılı listeye bir sayı eklemek için bir metot yazın. Listenin en küçükten en büyüğe doğru sıralandığını varsayalım. Ekledikten sonra, liste hala sıralanmış olmalıdır. $L1 = (3, 17, 18, 27)$ listesi ve 20 değeri verildiğinde, $L1$ 'in dönüşünde liste $(3, 17, 18, 20, 27)$ olur.
- Sıralanmış bir bağlantılı listede medyan değeri döndüren bir metot yazın. Listenin i uzunluğu tek ise, medyan $\text{ceil}(i/2)$ üyesidir. Örneğin, girdi olarak $(1, 2, 2, 5, 7, 9, 11)$ listesi verildiğinde, metot 5 değerini döndürmelidir. Listenin uzunluğu çift ise, medyan i 'nin ortalamasıdır. $i/2$ ve $(i/2)+1$ üye. Böylece sıralı listenin $(2, 4, 8, 9)$ medyanı $(4+8)/2$ 'dir. Son olarak, boş bir listenin medyanını 0 olarak tanımlayın.
- Bağlantılı bir listedeki düğümleri tersine çeviren bir fonksiyon yazın. Metodun zaman karmaşıklığı $O(n)$ olmalıdır, burada n , listenin uzunluğudur. Yeni düğüm oluşturmamalısınız.

KISITLAR:

- Projede sadece bir ana sınıf olmalıdır.
- Ek kütüphane kullanılamaz.
- Ödevlerde Geç gönderim **KESİNLİKLE** kabul edilmeyecektir.
- Gönderim dosya ismi= ogrencino.zip şeklinde olmalıdır. Bu dosya içinde ödev raporu da olmalıdır.
- Ödev 3 kısımdan oluşmaktadır. Rapor, Uygulama(Kod) ve javadoc
- Rapor kısmında proje raporu olmalıdır. Kod, UML diyagramı ve genel açıklamalar. Rapor %30 puan etkisindedir.
- Uygulama kısmında kod çalışır ve compile edilebilir olmalıdır. Compile edilemeyen kodlardan 30 puan kırılabilecektir. Test ve kaynak kodu birlikte gönderilmelidir. Sınıflar, yöntemler ve değişken adları anlamlı olmalıdır. Metod ve sınıflarınız basit, genel, yeniden kullanılabilir olmalıdır.
- Kod %70 değerindedir.

- Javadoc Bölümü: sınıflar, yöntemler, değişkenler vb. İçin tüm [javadoc](#) açıklamaları eklenmelidir.tüm açıklamalar anlamlı ve anlaşılır olmalıdır. [standart java kod adı kurallarını](#) kullanılmalıdır. Bu kısım ekstra 20 puan değerindedir.

NOTLANDIRMA:

Kopya: -500

Nesnel tasarım olmaması: -300 (Kalıtım, override,interface, polimorfizm, exception handling, soyut sınıf)

Komut yorumlarının olmaması: -95

Açıklama ve kısıtlamaların dikkate alınmaması: -200

Ödev doya ismi hatalı olması: -250

Sorular için Arş. Gör. Abdullah BAL ile Arş. Gör. Abdulsamet AKTAŞ'a e-mail atılabilir.