

MAKEFILE - processor

Önceki ödevde kullandığım makefile dosyası ve processor sınıfını biraz değiştirerek bu ödevde kullandım.

eList

2. ödevdeki liste sınıfım.

binaryTree

İkili ağaç sınıfım. Düğüm ekleme, maximum ve minimum elemanları bulma ve silme, bütün seviyelerdeki düğümleri level order olarak görüntüleme, istenilen seviyeyi level order olarak görüntüleme, belli bir düğümden n seviye sonra olan düğümleri level order olarak görüntüleme ve ağacın yüksekliğini bulan fonksiyonları barındırmakta. Derste kullandığımız pekiştirme fonksiyonlarını kullanmak adına; Çok verinin bulunduğu bir sisteme göre kodlamadım. Ağaçtaki düğümleri 2 fonksiyon ile yineleyerek taradım ve yüksekliğe ulaştım. Bu sistemden kaçınmak istenirse yapılması gereken tek şey düğüm eklerken ağacın yüksekliğini kayıt etmek olacaktır. Aynı şeyi level order içinde kullandım. Basit bir dizi ile çözülebilir. Başta kök düğümü bellekte tutmak istemedim fakat sonradan kod estetiği için kullanmam gerekti.

node

Düğüm sınıfım. Düğümün alt sağ ve alt sol elemanları, kök düğümü ve sahip olduğu değeri tutan alanlara sahip.

management

Verilen string'e göre obje oluşturan ve gerektiğinde heapte oluşturduğu ağaç yapısını silen createObjects ve destroyObjects fonksiyonlarını barındırıyor. Heap temizliğini

"Herkes kapısının önünü temizlerse tüm mahalle temiz olur." mantığı ile yaptım. Birini silmem heap bellekte oluşan bütün verileri temizlememe yarıyor.

menu

cin ve cout ile menü kodları.

Aşamalar

14.12.18' de başladım ödevde. İlk gün bütün sınıfları oluşturdum. İlk gün zorlandığım yer yoktu.

15.12.18' de kalan yarısını bitirdim. Gün sonunda programda hatalar vardı.

16.12.18' de hataları düzelttim. Önceki 2 ödevde nazaran kolaydı. Her zamanki gibi pointer hatası almadan bitiremedim ödevi. Hata NULL olup olmadığını kontrol etmediğim için oluyormuş.

Fakülte	BİLGİSAYAR VE BİLİŞİM BİLİMLERİ FAKÜLTESİ
Ders Adı	VERİ YAPILARI

ÖĞRENCİNİN

AD: **Eray**

SOYAD: **KAYA**

NO: **B161210069**

GRUP: **C**

ÖĞRETİM: **1**