Veri Yapıları RAPOR

B181210390 Zeynep ALDİNÇ

G161210309 Ömerfaruk YETİM

AMACI:

Hazırlanan programın amacı heap bellek üzerinde oluşturulan kova nesnelerinin, kullanıcın seçeneğine göre içlerine toplar yerleştirilmesi, topların yok edilmesi, seçilen iki farklı kovanın içerisindeki topların yerlerinin değiştirilmesi, seçilen iki kovanın yerlerinin değiştirilmesi, kovaların tersten yerleştirilmesi, topların tersten yerleştirilmesi. İlk kova nesneleri oluşturulduğunda ve her seçimden sonra hem kovaların hem de topların heap bellekteki konumları da kullanıcıya gösterilerek bu nesnelerin konumlarının da değiştiği belirtilmiş olur.

Top Sınıfı:

Kullanılacak değerler Top sınıfı içerisinde private alanda tanımlandı. Tanımlanan bu değerlere Top fonksiyonu içerisinde başlangıç değerleri atandı. TopIsmi değeri char olduğundan dolayı içerisine ‘\0’ (null) değeri, TopVarMi kontrol değerimize de bool olduğundan dolayı false değeri atandı.

TopIsmi değeri IsimAta() fonksiyonu içerisinde kullanıldı ve rand() fonksiyonuyla içerisine A-Z arasında rastgele bir değer gönderildi. TopVarMi değeri de artık TopIsmi belirlendiği için true değerini aldı.

TopDondur() metodu private alanda tanımladığımız ve içine IsimAta() fonksiyonu içerisinde bir değer atadığımız TopIsmi’ni döndürdü.

Private alanda bulunan TopVarMi değerini Top sınıfının dışından değiştirebilmek için değer TopVarMiSet() ve TopVarMiGet() metotlarıyla döndürüldü.

Kova Sınıfı:

Kullanılacak olan kovaRengi değeri Kova sınıfının private alanında oluşturuldu. Kova fonksiyonunun içerisinde topun değeri null yayıldı ve kovaRengi değerine, renkler bu sayılar arasında olduğu için, rand() fonksiyonuyla 1 ve 15 arasında rastgele sayılar atanması sağlandı. kovaRengi değeri private alanda olduğu için kovaRengi’ni döndüren RenkDondur() fonksiyonu yapıldı.

TopOlustur() fonksiyonu içerisinde Top sınıfından bir top nesnesi, heap bellekte oluşturuldu. Aynı fonksiyon içerisinde Top sınıfında oluşturulan IsimAta() fonksiyonu çağırıldı. Topun ismini alabilmek için burada da oluşturulan TopDondur() fonksiyonunun içerisine Top sınıfında oluşturulan TopDondur fonksiyonu çağırıldı.

kovaRengi private alanda oluşturulduğundan dolayı RenkDondur() fonksiyonuyla döndürüldü.

KovaKontrol Sınıfı:

KovaKontrol sınıfının private alanında kovaSayisi değeri ve kovalar nesnesini tutan bir dizi değişkeni oluşturuldu. KovaOlustur() isimli bir fonksiyon oluşturuldu ve içerisinde kullanıcıdan kova sayısı alındı. Alınan kova sayısı kadar elemanı olan kovalar dizisi oluşturuldu. Dizinin her elemanı için yeni bir kovalar nesnesi oluşturuldu. KovaKontrol() fonksiyonu içerisine KovaOlustur() fonksiyonu çağırıldı.

Yazdir() fonksiyonu ile alınan kova sayısı kadar kova nesnesinin şekilleri ve adresleri ekrana yazıldı. Henüz kullanıcı tarafından top eklemesi yapılmadığı için top adresi null olarak yazıldı. ToplariYerlestir() fonksiyonunda kova sayısı kadar kovalar dizisinin içerisine top oluşturuldu. ToplariYoket() fonksiyonunda kovaların içerisinde top varsa toplar silinip null’a eşitlendi. KovaDegistir() fonksiyonunda Kova sınıfından bir geçici değer oluşturuldu. Kullanıcıdan ilk ve ikinci kova için değer alındı. Bu değerlerin sıfır ile kova sayısı arasında olması kontrol edildi. Dizi indisi sıfırdan başladığı için alınan değerler dizide aranırken -1 eklendi. İlk kova değeri geçici değere, ikinci kova değeri ilk kova değerine ve son olarak da geçici değer ikinci kova değerine atanarak iki kovanın yeri değiştirilmiş oldu. TopDegistir() fonksiyonunda aynı işlem top nesneleri için tekrarlandı. KovalariTerstenYerlestir() fonksiyonunda GeciciKovalar adında bir dizi oluşturuldu. Kovalar dizisinin son elemanı gecicikovalar dizisinin ilk elemanına eklenerek döngü bu şekilde devam ettirildi. Ardından gecicikovalar dizisi, kovalar dizisine atıldı. Aynı şey toplar için ToplariTerstenYerlestir() fonksiyonunda yapıldı. KovaKontrol() yıkıcı fonksiyonunda kovalar dizisindeki nesneler silindi.

Test.cpp

Bu bölümde kullanıcının önüne gelecek olan seçenek menüsü ve bu menüdeki seçimlere göre çalışması gereken fonksiyonlar ayarlandı. Menü içerisine “Topları Yerleştir”, ”Topları Yok Et”, “Kova Değiştir”, “Top Değiştir”, “Kovaları Tersten Yerleştir”, “Topları Tersten Yerleştir” ve “Çıkış” seçenekleri eklendi.

Calistir() fonksiyonu içerisinde kullanıcıdan alacağımız seçim değeri tanımlandı. KovaKontrol sınıfındaki fonksiyonları alabilmek için öncelikle KovaKontrol sınıfından bir nesne oluşturuldu. While içerisinde KovaKontrol’ün Yazdir() fonksiyonu çağırılarak kovaların ekranda oluşması sağlandı. Kovalardan sonra ekrana menü gelmesi için Menu() fonksiyonu çağırıldı. secim değeri kullanıcıdan alındı ve switch case döngüsü oluşturularak her seçim için kullanılacak olan fonksiyonlar ilgili seçimlerin altına kovakontrol’den getirildi. Çıkış yapıldığı anda KovaKontrol sınıfından oluşturulan nesne delete yardımı ile silindi(heap’te çöp bırakılmaması amacıyla).

int main() içerisinde Calistir() fonksiyonu çağırıldı. sistemin her çalıştırılışında farklı top isimleri ve farklı renkler oluşturulması için srand() fonksiyonu çağırıldı.