

Ödevde pointer'ların ve C++'daki sınıf yapısının kullanılarak daha iyi anlaşılması hedeflenmiştir. Bu doğrultuda yaptığım projede Basket, Ball ve BasketManager olmak üzere 3 sınıf bulunmaktadır. Ball sınıfında topun değeri char olarak tutulmaktadır ve bu değer Ball sınıfının constructor'ında random olarak (A, B, C... Z) atanır. Değere sınıf dışından atama yapmak engellenmiştir, sadece getValue methodu ile okunabilir. Basket sınıfında, Ball sınıfının nesnesi tutulmuştur. Ayrıca sepetin renk değeri de integer olarak tutulmuştur. Bu integer değer console'un colorSet fonksiyonuna atanarak renk değişikliği Basket sınıfının print fonksiyonunda sağlanır. Daha sonra bu fonksiyonun çağrıldığı ana fonksiyonda renk değeri tekrar beyaza döndürülerek her sepetin farklı renkleri olması sağlanmıştır. Bu renk değerleri ise Basket sınıfının constructor'ında atanır. Son olarak da BasketManager sınıfında Ball ve Basket sınıfının nesneleri bulunur. Programdaki ana algoritmalar ve işlemler bu sınıfta oluşturulmuştur.

Bu ödevde, C++'ın OOP yaklaşımını daha iyi anladım. Geçen sene C# ile programlar yazdıktan sonra C++ ile yazarken biraz zorlandım diyebilirim. Özellikle bellek yönetimi ve pointer konularını anlamakta ve dizi oluşturmakta çok zorlandım. Bunun yanı sıra derleme aşaması da beni zorladı. Sınıfları dahil etme ve çağırma konularını kafamda hala daha tam olarak oturtamadım.

C++'ın pointer yapısını anlamak ne kadar zor olsa da kullanmanın büyük avantajlar sağladığını gördüm.

Öğrendiğim en büyük bilgilerden biri de heap bölgesinde oluşturulan dizi silinmeden önce elemanlarını da tek tek silmem gerektiği idi. Bu zorunluluğun C++ için bir avantaj değil de bir eksiklik olarak görüyorum. Tabii ki daha ileri düzey programlarda performans olarak bir avantaj sağlıyor olabilir. Ancak her elemanı silmek için for döngüsü yazmak pek avantajlı durmuyor.

Sepetleri ve topları sıralarken, daha önce oyun programlarken öğrendiğim temp değişkenini kullandım. Değiştirilmesi istenen değerlerden bir tanesi önce bu temp değişkenine atanır. Daha önce kopyasını temp ile aldığımız değere, diğer değer atanır. Diğer değere de kopyasını aldığımız değer kopyası yani temp değeri atanır. Bu sayede dizideki elemanlar yer değiştirmiş olur. Bu algoritmayı kullanarak dizideki elemanları tersten sıralamayı başardım.

Ana menüyü oluştururken do while döngüsünü tercih ettim. Bunun sebebi önce ekrana yapılacakların yazdırılması daha sonra kullanıcıdan dönen cevaba göre işlem yapılmasıydı. Menüdeki seçimleri de switch case yapısı ile kontrol ettim. Bu yapıyı kullanırken süslü parantez gerektiğini öğrendim. C#'da böyle bir gereklilik olmamasından dolayı bu hatayı çözmem de biraz zamanımı aldı.

Genel olarak bu ödevden öğrendiklerim; C++ Syntax'ı, C++ Sınıf yapısı, Pointers ve Bellek Yönetimi kavramları oldu.