



**T.C.**  
**SAKARYA ÜNİVERSİTESİ**

**BİLGİSAYAR VE BİLİŞİM BİLİMLERİ FAKÜLTESİ**  
**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**  
**PROGRAMLAMA DİLLERİNİN PRENSİPLERİ PROJE RAPORU**

**BİR HABİTATTAKİ EKOLOJİK PİRAMİT**

**Sinem Şaziye Karaca**  
**G211210025**  
**1 - A**

**SAKARYA**  
**Mayıs, 2024**  
Programa Dillerinin Prensipleri

## BİR HABİTATTAKİ EKOLOJİK PİRAMİT

Programlama Dillerinin Prensipleri Dersi kapsamında gerçekleştirdiğimiz ikinci projede bir habitattaki farklı türden canlıların ekolojik piramidini taklit etmemiz istenmektedir. Proje kapsamında kullanacağımız programlama dili C olup dilin yapısında bulunmayan nesneye dayalı tasarımın benzetiminin yapılması beklenmektedir. Benzetimi yapılması istenen en önemli iki yapı nesneye dayalı tasarımdaki kalıtım ve soyut sınıf yapılarıdır. Private yapısı benzetimi de projemde ek olarak benzetimini yaptığım yapıdır. Projede uygulanması istenen canlı hiyerarşisi şu şekilde: Sinek ve Pire sınıfları Böcek sınıfından, Böcek ve Bitki sınıfları ise Canlı sınıfından kalıtım almaktadır. Ekolojik piramit ise şu şekildedir: Bitki, Pire ve Sineği; Böcek, Bitki ve Pireyi; Sinek, Pire ve Böceği yemektedir.

C dilinde bulunmayan kalıtım yapısının benzetimi, C dilinde var olan struct yapısı kullanılarak yapıldı. Burada struct yapısı nesne yönelimli tasarımda class yapısını temsil etmektedir. Fakat class yapısından farklı olarak struct yapısının içinde fonksiyonlar direkt tanımlanamaz. Bunun için, herhangi bir fonksiyonun struct yapısına dahil olduğunu belirtebilmek için, fonksiyon göstericileri kullanıldı. Hiyerarşideki her bir grubu temsil etmesi için kendi isimleriyle tanımlanmış struct'lar oluşturuldu. Bu yapılarda kalıtım hiyerarşisini gerçekleştirebilmek için her bir ebeveyn sınıfın çocuk sınıflarında "super" anahtar kelimesiyle ebeveyn sınıfın üyesi oluşturuldu. Örneğin canlı sınıfından kalıtım alan bitki ve böcek sınıflarında canlı sınıfından, böcek sınıfından kalıtım alan sinek ve pire sınıflarında böcek sınıfından bir değişken oluşturuldu. Bu yapı ebeveyn sınıfların üyelerine ulaşmak için kullanılmaktadır. Her bir canlıda ortak olarak bulunan tür, değer ve konum gibi özellikler canlı sınıfının bir üyesi olarak tanımlanmış, çocuk sınıfların canlı sınıfından kalıtım alarak bu özellikleri kendilerine göre oluşturması sağlanmıştır. Her bir sınıfın içeriğinde nesne yönelimli tasarımın benzetimi için kurucu fonksiyon tanımlanmıştır. Bu fonksiyonlar, ait oldukları sınıflardan yeni nesneler oluşturulduğunda çağrılmakta ve canlıların sahip olduğu özelliklere ilk değerlerini atamaktadır. Nesne yönelimli tasarımda çocuk sınıftan nesne üretildiğinde otomatik olarak ebeveyn sınıftan da nesne üretilmektedir. Projede bu kısmın benzetiminin yapılabilmesi için her bir çocuk sınıf oluşturulduğunda çağrılan kurucularının içinde kalıtım aldıkları ebeveyn sınıflarının da kurucu fonksiyonu çağrılmaktadır. Böylelikle, örneğin bitki sınıfından nesne oluşturmak için özelliklerini parametre olarak gönderdiğimiz bitki kurucu fonksiyonunda, yine aynı özelliklerle canlı kurucu fonksiyonu çağrılmakta ve bitki nesnesi oluşturulduğunda aynı anda canlı nesnesi de oluşturulmaktadır.

Proje çalıştığında bu yapıların aktifleşebilmesi için içinde 1 ile 99 arasında sayıların bulunduğu .txt uzantılı dosya okunmaktadır. Bu dosyanın içindeki değerler canlıların kendi değerlerini temsil etmektedirler. Her bir canlı türü için bir değer aralığı belirlenmiştir. Dosyadan okunan değer 1-9 aralığındaysa bitki, 10-20 aralığındaysa böcek, 21-50 aralığındaysa sinek, 51-99 aralığındaysa pire sınıfından bir nesne üretilmektedir. Üretilen her bir nesneyle birlikte, aynı değerlere sahip bir canlı nesnesi kalıtımın yapısından dolayı otomatik olarak üretilmektedir. Tüm canlı nesneleri, canlının türü fark etmeksizin, canlılar isimli canlı struct pointerları içeren diziye eklenmektedir. Aynı zamanda her bir canlı türünün nesnesi de kendi türünden struct nesnelerini barındıran dizilere eklenmektedir. Canlılar dizisi oluşturulduktan sonra bitki, böcek, sinek ve pire nesneleriyle işlemiz bittiğinden bunları depolayan diziler serbest bırakılmaktadır. Projenin genelinde kurmakta en zorlandığım yapı bu oldu.

Canlılar arası besin zinciri canlılar dizisi üzerinden gerçekleştirilmektedir. Hangi canlının öleceği hangisinin yaşayacağı öncelikle ekolojik piramide bakılarak kararlaştırılır. Eğer karşı karşıya gelen canlılar aynı türdensen değerleri büyük olan, değerleri de aynıysa konumu bitişe en yakın olan canlı yaşayacak şekilde bir yapı oluşturulmuştur. Proje çalıştırıldığında her bir canlı uygun diziye yerleştirilir ve bu görüntü görünüm fonksiyonu ile ekrana basılır. Canlılar arasındaki besin döngüsü klavyeden bir tuşa basıldığında başlar ve en son hayatta kalan canlı ekrana yazılır. Ardından canlılar dizisi bellekten serbest bırakılır ve program sonlanır.