



SAKARYA
ÜNİVERSİTESİ

T.C.

SAKARYA ÜNİVERSİTESİ

BİLGİSAYAR VE BİLİŞİM BİLİMLERİ FAKÜLTESİ

BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

PROGRAMLAMA DİLLERİNİN PRENSİPLERİ ÖDEV RAPORU

**Canlı Türleri Arasındaki Beslenme İlişkilerinin Analizi: C Programlama
Yaklaşımı**

Prof.Dr. AHMET ZENGİN

B211210008-SENA AKSU

MAYIS,2024

Canlı Türleri Arasındaki Beslenme İlişkilerinin Analizi: C Programlama Yaklaşımı

Sena AKSU*B211210008-2B

Özet

Projenin amacı, doğal bir habitat içindeki farklı türler arasındaki etkileşimleri modelleyerek, bu etkileşimlerin sonuçlarını analiz etmektir. Organizmalar arasındaki ilişkiler, beslenme zincirindeki ilişkileri temsil etmektedir ve bu ilişkilerin doğal yaşamdaki etkilerini anlamak için matematiksel bir model oluşturulmuştur. Organizmalar arasındaki etkileşimlerde birbirlerini yeme mantığı şu şekildedir:

Bitki → Pire

Böcek → Bitki

Sinek → Pire

Bitki → Sinek

Sinek → Böcek

Böcek → Pire

Bu problemin çözümü için C programlama dilini tercih ettim ve kalıtım ve soyut sınıf benzetimi kullanarak yapılandırdım. Yazılım geliştirmemde modülerlik ilkesini ve tek sorumluluk prensibini uygulayarak tasarladım. Proje kapsamında, farklı canlı türlerini temsil eden yapıları oluşturmak için struct yapılarını kullandım ve her canlı türünü ayrı başlık altında (header file) tanımladım. Bitki, böcek, sinek ve pire türlerini farklı özellikler ve davranışlarla modellendirdim.

Anahtar Kelimeler: kalıtım , struct yapıları, beslenme zinciri

1. GELİŞTİRİLEN YAZILIM

İlk olarak, Canli.h ve Canli.c dosyalarında temel bir CANLI yapısı tanımlanmıştır. Bu yapı, canlıları temsil eden genel özellikleri içerir. Özellikler arasında canlının değeri (deger), sembolü (sembol), görünümü (gorunum) ve yok edilmesi için bir fonksiyon (yokEt) bulunur. Bitki.h, Bocek.h, Sinek.h, ve Pire.h dosyalarında ise bu genel yapıdan türetilmiş özel canlı türleri için yapılar yer verilmiştir. Örneğin, BITKI yapısı bitkileri, BOCEK yapısı böcekleri temsil eder.

Daha sonra, Habitat.h ve Habitat.c dosyalarında HABITAT yapısı tanımlanmıştır. Bu yapı, habitatı temsil eder ve habitat içindeki canlıları barındırır. Habitatın satır ve sütun sayıları belirli bir matris boyutunu temsil eder. goster fonksiyonu, habitat içindeki canlıları ekrana basar. guncelle fonksiyonu ise habitat içindeki canlıların etkileşimlerini simüle eder ve günceller. yokEt fonksiyonu ise bellek yönetimi için kullanılır.

Ana main fonksiyonunda ise proje işlevselliği gerçekleştirilir. İlk olarak, textVeri.txt dosyasından alınan verilerle bir matris oluşturulur. Bu matris, habitatı ve içindeki canlıları temsil eder. Her bir eleman, CANLI türünden türetilen özel bir canlı nesnesiyle ilişkilendirilir. Bu nesneler, canlıların özelliklerini ve davranışlarını temsil eder.

Daha sonra, matris üzerinde her adımda goster ve guncelle fonksiyonları çağrılarak habitatın görünümü güncellenir ve canlıların etkileşimleri simüle edilir. Etkileşim sonucunda bazı canlılar birbirini yer ve matris üzerinde hareket ederler. Bu işlem adım adım ilerler ve ekranda canlıların etkileşimleri takip edilir. Son olarak, simülasyon tamamlandığında bellek yönetimi için ayrılan alanlar serbest bırakılır ve program sonlanır.

2. ÇIKTILAR

```

10 2 5 4 7
21 6 1 25 8
11 64 33 41 47
8 1 9 5 23|

Microsoft Windows [Version 10.0.22631.3527]
(c) Microsoft Corporation. Tüm hakları saklıdır.

C:\Users\sena\Desktop\PDP SON DENEME\pdp>mingw32-make
gcc -I ./include/ -o ./lib/Canli.o -c ./src/Canli.c
gcc -I ./include/ -o ./lib/Bitki.o -c ./src/Bitki.c
gcc -I ./include/ -o ./lib/Bocek.o -c ./src/Bocek.c
gcc -I ./include/ -o ./lib/Sinek.o -c ./src/Sinek.c
gcc -I ./include/ -o ./lib/Pire.o -c ./src/Pire.c
gcc -I ./include/ -o ./lib/Habitat.o -c ./src/Habitat.c
gcc -I ./include/ -o ./bin/Test ./lib/Canli.o ./lib/Bitki.o ./lib/Bocek.o ./lib/Sinek.o ./lib/Pire.o ./lib/Habitat.o ./src/Test.c
./bin/Test

X X X X X
X X X X X
X X X X X
X X B X X
Kazanan: B : (3,2)
-----
C:\Users\sena\Desktop\PDP SON DENEME\pdp>

```

3.SONUÇ

Simülasyonun sonunda, habitat içindeki canlı türler arasında bir kazanan belirlenmiştir. Bu kazanan, simülasyonun başlangıcında belirlenen kurallar ve canlıların özellikleri temelinde belirlenmiştir. Bellek yönetimi konusunda dikkatli bir şekilde tasarlanmıştır. Bellek sızıntılarına karşı önlemler alınmış ve programın performansı optimize edilmiştir. Projede eksik bulunmamaktadır.

Referanslar

- [1] <https://youtu.be/yMHGt2Bcsts>
- [2] https://youtu.be/ov_GudbL1qo
- [3] <https://stackoverflow.com/questions>