



T.C.
SAKARYA ÜNİVERSİTESİ

BİLGİSAYAR VE BİLİŞİM BİLİMLERİ FAKÜLTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ
PROGRAMLAMA DİLLERİNİN PRENSİPLERİ ÖDEV RAPORU

C Dilinde Nesneye Dayalı Programlamanın
Kavramları Benzetimi

B181210568 - Zidan Omar Hamid

SAKARYA

Mayıs, 2024

C dilinde Nesneye Dayalı Programlamanın Kavramları Benzetimi

Zidan Omar Hamıd

B181210568 -1A

Özet

Nesneye dayalı programlamadaki katılım ve soyut benzetimi uygulayarak C dilinde bir proje geliştirmek

© 2024 Sakarya Üniversitesi.

Bu rapor benim özgün çalışmamdır. Faydalanmış olduğum kaynakları içerisinde belirttim. Her hangi bir kopya işleminde sorumluluk bana aittir.

Anahtar Kelimeler: C, Nesneye Dayalı Programlama, Katılım, Soyutlama

1. Projenin Amacı

Program çalıştırdığı zaman, Veri.txt dosyasından veri okuyacak. Bu dosyada değişken sayıda olabilecek satır ve sütunlara yerleşmiş ve aralarında 1 boşluk bulunan tam sayıları okuyacaktır. Sayılar 1-99 arasındaki sayılardan olabilecektir. Bu sayılar canlıların değerini ifade etmektedir ve bir canlı, belirttiği diğer canlı nesnesinden oluşmaktadır. Sonunda sırayla bir canlı diğer canlıyı yiyip en son kalan canlıyı ekrana yazılacaktır.

2. Projede Kullanılan Yöntemler

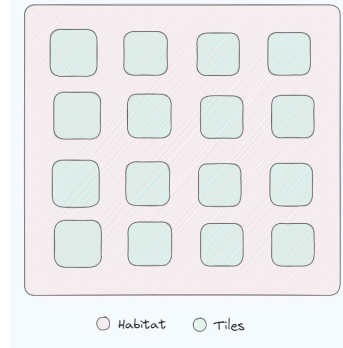
Veri.txt dosyası oluşturduktan sonra bir habitat oluşturuyor. Habitat içinde *tile* denilen bir struct var. C dilinde generic veri tipi tanımlanmadığı için bir array da farklı veri tipi veya her zaman değişebilen veri tipi oluşturamaz. Bunun yüzünden bir array içinde farklı bir canlı tutmak için ve o canlı hangi türünden temsil ettiğini tanımlayabilir. *Tile* struct canlının kendi verisi ve canlı tip birleşiminden (Union) oluşturuldu.

```
Habitat.h

enum SpeciesType
{
    TYPE_BITKI,
    TYPE_BOCEK,
    TYPE_SINEK,
    TYPE_PIRE,
};

union SpeciesUnion
{
    Bitki bitki;
    Bocek bocek;
    Sinek sinek;
    Pire pire;
};

struct Tile
{
    enum SpeciesType type;
    union SpeciesUnion data;
};
```



Diğer yöntem ise projede kullanmamız gereken NDP kavramı benzetimidir. Canlı nesnesi doğrudan kullanmak veya aynı fonksiyon farklı canlı sınıfı tekrar yazmak yerine soyut sınıf olarak tanımlandı. Ve örnek olarak bitki ve böcek alt sınıfı oluşturup canlı sınıfın özellikleri kullanabiliriz. Bu yöntemle katılım ve soyut kavramı uygulanmış olur.

```
Bocek.c

void PrintBocek(const Bocek this)
{
    printf("%s", this->super->GetCanli(this->super));
};

void KillBocek(const Bocek this)
{
    this->super->KillCanli(this->super);
};
```

```
Pire.c

void PrintPire(const Pire this)
{
    printf("%s", this->super->super->GetCanli(this->super->super));
};

void KillPire(const Pire this)
{
    this->super->super->KillCanli(this->super->super);
};
```

Yukarıdaki kod parçası gibi, Böcek canlıının alt sınıfıdır. Böceği ekrana göstermek için canlıdan bir fonksiyon çağırıldı çünkü böcek, kendine ifade eden canlı türü bilgisi tutmadı sadece canlı sınıfında veriyi saklandı ve aynı şekilde pire sınıfı, böceğin alt sınıfıdır. Kendi yaşayıp yaşamadığının durumu veya kendi gösterge fonksiyonu oluşturulmasına gerek yok direk üst sınıfın fonksiyonu (böcek sınıfı) çağırıp otomatik olarak o da canlı sınıfında bilgileri erişebiliyor.

3. Örnek Ekran Çıktısı

```
odev-2 : x + v
→ odev-2 ⚡ (P main) 4 days ago
▶ make
rm -f lib/*.o bin/Test
gcc -Iinclude -c src/Bitki.c -o lib/Bitki.o
gcc -Iinclude -c src/Bocek.c -o lib/Bocek.o
gcc -Iinclude -c src/Canli.c -o lib/Canli.o
gcc -Iinclude -c src/FileUtil.c -o lib/FileUtil.o
gcc -Iinclude -c src/Habitat.c -o lib/Habitat.o
gcc -Iinclude -c src/Pire.c -o lib/Pire.o
gcc -Iinclude -c src/Sinek.c -o lib/Sinek.o
gcc -Iinclude -c src/Test.c -o lib/Test.o
gcc lib/Bitki.o lib/Bocek.o lib/Canli.o lib/FileUtil.o lib/Habitat.o lib/Pire.o lib/Sinek.o lib/Test.o -o bin/Test -lm
bin/Test

Current Habitat:
C B B B B
S B B S B
C P S S S
B B B B S

Press any key to continue ...
|
```

```
X X X X X
X X X X X
X X X X X
X X B X X

Kazanan: B : (3,2)
Time taken: 0.001428
→ odev-2 ⚡ (P main)
▶ |
```

Referanslar

- [1] "Java Abstract Classes and Methods." W3Schools. (Eriřim tarihi: 5 Mayıs 2024) https://www.w3schools.com/java/java_abstract.asp
- [2] "Java Interfaces." W3Schools. (Eriřim tarihi: 5 Mayıs 2024) https://www.w3schools.com/java/java_interface.asp
- [3] "Create an array with elements of different types" yanıtı. Stack Overflow. Yanıt tarihi: 4 Ekim 2011, 10:19. Ed Heal tarafından yanıtlandı. (Eriřim tarihi: 5 Mayıs 2024) <https://stackoverflow.com/a/7646384>