



T.C.
SAKARYA ÜNİVERSİTESİ

**BİLGİSAYAR VE BİLİŞİM BİLİMLERİ FAKÜLTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

PROGRAMLAMA DİLLERİNİN PRENSİPLERİ ÖDEV RAPORU

2. ÖDEV - C DİLİNDE CLASS BENZETİMİ

B221210064-Tuğra YAVUZ

SAKARYA

MAYIS, 2024

Programlama Dillerinin Prensipleri Dersi

Özet

Ödevde bizden istenen C dilinde abstract class benzetimi yaparak ilgili ".txt" dosyasından gelen verilere göre bitki, böcek, pire ve sinek yapılarını oluşturup karşılaştırmaktır. Bu benzetimi yapmanın yanında bize bir kalıtım hiyerarşisi oluştururmak da istenmektedir ve abstract class kullanımı burada önemli bir değer arz etmektedir. Ayrıca dosyayı okuduktan sonra oluşturduğum yapıları saklamak için bir "void*" işaretçi tutan dizi veya bağlı liste kullanılması gerekmektedir. Ben Veri Yapıları dersinden kalan alışkanlıklarımdan dolayı bağlı liste kullanmayı tercih ettim. Kodum bu bağlı liste üzerinde gezerek önce nesneleri soyut fonksiyon olan "gorunum()" fonksiyonu ile ekrana yazdırıyor. Ardından raporda istenen parametreler ile ilgili karşılaşmaları yapıyor ve elenen nesneyi bellekten serbest bırakıp bir sonraki adıma geçiyor. Son olarak ise kazanan nesneyi ekrana ilk baştaki konumuna göre yazıyor. Bu işlemlerin sonunda da bağlı liste düğümleri bellekten boşaltılıyor ve program sonlanıyor. Ben bu işlemleri "habitat" sınıfı benzetimi içinde yapmayı tercih ettim ve "main" içerisinde "habitat" nesnesi oluşturma ve fonksiyon çağrımları yaparak olabildiğince sade tuttum.

© 2024 Sakarya Üniversitesi.

Bu rapor benim özgün çalışmamdır. Faydalananmış olduğum kaynakları içerisinde belirttim. Herhangi bir kopya işleminde sorumluluk bana aittir.

Anahtar Kelimeler: class, struct, benzetim, c dili

1. GELİŞTİRİLEN YAZILIM

Ödevime öncelikle literatür taraması ile başladım. Abstract classlar üzerine Java, C++ gibi dillerde kendimi geliştirdim. Bunun ardından C dilinde hakimiyetimi artırmak için "Programlama Dillerinin Prensipleri" kitabımdan çeşitli örnekler göz gezdirdim [1]. Tarama bittiğinden sonra algoritmayı geliştirmeye hazır hale gelmiştim. Kağıt defter alarak karşılaştırma algoritmayı hazırlamaya koyuldum. Bu algoritma tüm düğümleri iki gezici ile çalışacak ve düğümleri karşılaştırarak devam edecekti. Kodu yazmadan önce makefile dosyamı ve .h-.c dosyalarımı oluşturdum. Öncelikle canlı, bitki, pire, böcek ve sinek sınıflarının C dilinde benzetimini yaptım. Burada abstract class benzetimi yapmaya özen gösterdim ve abstract class olarak "canlı" sınıfını kullandım. Soyut fonksiyon olarak ise ekrana ilgili nesnenin türünü yazdırımı yarayan "gorunum" fonksiyonunu kullandım. Ayrıca "canlı" sınıfı içinde tüm yapıalarda kullanacağım ve kalıtima uygun olan kapışma değeri ("txt" dosyasından okunan değer), düğümün x ve y eksenlerini tutmama yarayacak değişkenleri sakladım. Nesnelerin sınıflarını yazdıktan sonra ".txt" dosyasından veri okumamı sağlayacak kodu yazıp ".txt"

dosyasından okunan sayıya göre nesne oluşturup bağlı liste düğümlerinde sakladım. Bu işlemlerin sonunda verilerim hazırdı, bu yüzden "habitat" sınıfı benzetimini yazmaya başlayabildirdim. Bu sınıf içinde kıyaslama fonksiyonlarını ve bu kıyaslama fonksiyonuna yardımcı olacak fonksiyonları yazıp kazanarı da bu struct içinde sakladım. Yardımcı fonksiyonlara örnek olarak bağlı liste düğümünü alıp verisini cast işlemeye tabii tutarak x, y, kapışma değeri gibi fonksiyonları söyleyebilirim. Sıra karşılaştırma fonksiyonunu yazmaya gelince öncelikle bir temp değişkeni atayarak koda başladım. Çünkü bağlı liste değişkenlerim global değişkendi ve herhangi bir hatam ilk düğümün kaybı ile sonuçlanabilirdi. Temp'leme işleminden sonra 2 adet gezici düğüm değişkeni tanımladım. Bu kodların çalışma prensibi ise şu şekildeydi: İki veri kapışma değerlerine göre, nesne tiplerine göre ve bitiş noktasına uzaklıklarına göre kıyaslanıyor. Eğer 2. gezicinin değeri elenirse, 2. gezici diğer geziciye geçiriliyor. Eğer ki elenen 1. gezici ise, 1. gezici eleyen 2. geziciye eşitlenip 2. gezici bir sonraki basamağa geçirilip bu işlemler tüm veriler karşılaştırılana kadar devam ediyor. Her karşılaştırma işleminin sonunda bellekten silinen elemanın verileri tipine göre ilgili "yoket" metodu çağırılarak bellekten yok ediliyor, bu sayede bellek sızıntısının da önüne geçirilmiş oluyor. Bu kıyaslama fonksiyonu geri dönüş değeri olarak ise kazanan düğümü dönüyor. Kodun "main" fonksiyonu içerisinde ise şu işlemleri yapıyor: Dosya oku methodu çağrılarak dosyayı okuyor. Ardından ekrana "gorunum" fonksiyonları ile verileri yazıp işlemlere başlamak için bir tuşa basılmasını istiyor. Tuşa basıldıktan sonra kazanan bulunup ekrana eelenenler "X" olacak şekilde tekrar yazdırılıyor. Bunun yanında kazanan düğümün türü ve matristeki konumunu da yazdırmayı unutmuyorum. Yazdırma işleminin ardından kıyas fonksiyonu esnasında bellekten silinemeyen düğümün tuttuğu void işaretçisinin tuttuğu verinin tipini tespit edip uygun şekilde bellekten serbest bırakıyorum. Son olarak ise bağlı liste düğümlerini bellekten serbest bırakıyorum ve programı sonlandırıyorum.

2. ZORLANDIĞIM KİSİMLAR

Öncelikle c dili orta seviyeli bir dil olduğu için koud denetlemek c++, java gibi daha yüksek seviyeli dillere göre daha çok dikkat gerektiriyor. Bu sebepten dolayı özellikle karşılaştırma fonksiyonunda çeşitli dikkat hataları ile çok zaman kaybettim. Bunun dışında karşılaştırma fonksiyonu sonrası ekrana yazdırırken de epey zorluk zorluk yaşadım çünkü kodun ilk versiyonunda bazı yerlerde veriler kayboluyordu. Bu sorunu ise fonksiyonun çağrı yöntemini değiştirmek çözümdüm.

3. ÇIKTILAR

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

Kodum bu girdiye aşağıdaki çıktıyı vermektedir:

```
XXXXX
XXXXX
XXXXX
XXBXX
```

kazanan:B(3,2)
96 58 78 61

87 54 23 14

15 68 48 23

Kodum bu girdiye aşağıdaki çıktıyı vermektedir:

XXXX

XXXX

XXSX

kazanan:S(2,2)

4. SONUÇ

Bu kod c dilinde abstract class benzetiminde ve pointer kullanımı konusunda öğrenciyi geliştiren bir ödevdi.

Ödevi yaparken yaptığım pratik sayesinde c dilinde daha rahat kod yazabilmekteyim.

Referanslar

- [1] Yumuşak, N., & Adak, M. F. (2021). Programlama Dillerinin Prensipleri: Mantığı Kavra – İşleyişi Öğren – Kodu Yaz