



Sakarya Üniversitesi
Bilgisayar Mühendisliği

Programlama Dilleri Prensibleri

2.Ödev Ödev Raporu

Lütfü Orhun İNAN
B201210397
1/B
orhun.inan@ogr.sakarya.edu.tr

C Dili Kullanarak Nesne Yönelimli Benzetimi Yapılı Program Tasarımı

Lütfü Orhun İNAN¹
¹B201210397 1.B Grubu

Özet

Ödevin amacı C dilinde nesne yönelimli benzetimili uygulama tasarlamak, ve bu uygulama ile dosya okumaktır. Ödev kontrolü için Kisiler.txt ve Sayilar.txt adlı iki dosya okunacaktır. Uygulama içinde Kişi yapısı, Oyun yapısı, Dosya okuma yapısı ve Test uygulama kendilerine ait dosyalar üzerinde, bu dosyalarda istenen klasör düzeninde olmalıdır. Okunan dosyalar içerisindeki verilerle işlemler yapılmalı, bu işlemlerin sonuçları istenen şekilde ekran çıktıları olmalıdır. Ödev MinGW gcc compiler ile derlenebilmelidir.

Anahtar Kelimeler: C, OOB, File Reading

© 2022 Sakarya Üniversitesi.

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

İçindekiler

1. Kodun Çalışma Yöntemi	2	3. Karşılaşılan Problemler ve Çözümleri	3
2. Yazılım Tasarım Açıklamaları	2	4. Program Çıktıları ve Debug Özelliği	3
1 Nesneler ve Tasarımları	2	5. Sonuç	3
1.A PlayerArray	2	6. Kullanılan Uygulamalar ve Kaynakça	3
1.B DiceArray	2		
2 Fonksiyonlar ve Amaçları	2		

1. Kodun Çalışma Yöntemi

Uygulama beş dosyadan ve iki Nesne yönelimli türetilmiş veri yapısından oluşmaktadır. Okunması istenen iki dosyanın içerikleri bu türetilmiş nesneler içine yerleştirilir ve diğer dosyalara parametre olarak gönderilir. Dosya.c içinde bulunan fonksiyonlara verilen nesneler içine, Dosyada bulunan veriler satır satır okunup ayrıştırılarak eklenmektedir. Veriler okunduktan sonra istenen kurallara göre tasarlanmış fonksiyona parametre olarak verilirler. Sonuçlar ekrana çıktı olarak yazılır.

2. Yazılım Tasarım Açıklamaları

1. Nesneler ve Tasarımları

1.A PlayerArray

Kisi.h içinde belirlenmiştir. 6 farklı özellikleri vardır. (char name[128], double health, double changeRate, int diceNumber, next ve prev pointer) bununla beraber 3 methodu vardır. (PlayerInsert, PlayerDelete, PlayerPrintAll). Bunlarla beraber bu nesneyi başlatan PlayerInitilize fonksiyonu bulunmaktadır. Bu nesne doubly linked array görevi görmektedir ve "Kisiler.txt" dosyasındaki verileri tutmaktadır.

1.B DiceArray

Oyun.h içinde belirlenmiştir. 2 özelliği (int diceNumber, next pointer) ve 2 methodu (DiceInsert ve DicePrintAll) vardır. İsminin aksine bu nesne sadece linked list görevi görmektedir. Amacı "Sayilar.txt" dosyasındaki verileri tutmaktır.

2. Fonksiyonlar ve Amaçları

2.A ConsolePrinter.gotoxy(int x, int y)

Amaç konsol imlecinin yerini ayarlamaktır.

2.B ConsolePrinter.printBorders(int satir, int sutun) ve ConsolePrinter.printBordersWith2(int satir, int sutun)

Amaç verilen satır ve sütun girdileri ile konsol ekranında kenarlık oluşturmaktır.

2.C ConsolePrinter.printGameScreen ve printEndOfTheGameScreen

Amaçları oyun sırasında ve sonundaki sonuçları ekrana göstermektir.

2.D Kisi.PlayerInsert, Kisi.PlayerPrintAll, Kisi.PlayerDelete ve Kisi.PlayerHighest

Amaçları sırasıyla, verilen PlayerArray içine nesne eklemek, içindeki bütün nesnelerin çıktısını almak, içindeki bir nesneyi silmek ve verilen PlayerArray içindeki en yüksek değere sahip nesneyi döndürür.

2.E Dosya.rfReadPlayers ve Dosya.rfReadDice

Amaçları verilen dosyalar içindeki verileri verilen nesneler içine yerleştirmektir. önlerindeki rf read file anlamına gelmektedir. rfReadPlayers aynı zamanda verilen '#' ayracına göre satırları bölmektedir.

2.F Oyun.DiceInitilize ve Kisi.PlayerInitilize

Amaçları istenen nesnelerin yenilerini oluşturmaktır. new ile aynı amaçlıdır.

2.G Oyun.PlayTheGame

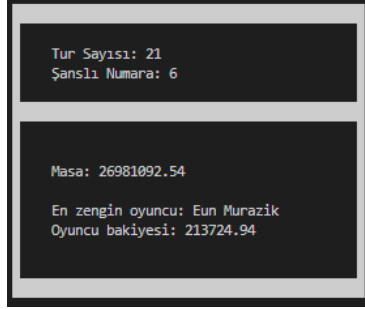
Amaç verilen kurallar içerisinde, nesnelerdeki verileri kullanarak çıktılar üretip gereken fonksiyonlara vermektir.

3. Karşılaşılan Problemler ve Çözümleri

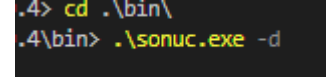
İlk karşılaşılan problem fonksiyonlar arası nesnelerin aktarımı ve bu verilerin işlenmesi. Fonksiyonlar içinde eğer nesnenin elemanlarından herhangi biri silinirse program Segmentation Fault yada segfault hatası vermektedir. Bunun için tasarlanan delete fonksiyonları denemelerde hatasız nesne elemanlarını silsede başka bir fonksiyon içinden çağrıldığında segfault erroru yaşanmaktaydı. Buna çözüm olarak nesne elemanlarını silmek yerine 1000.00tl altındaki kişilerin hesaplara katılması if kullanarak engellenmiştir. İkincisi ise nesnelerin kullanılması sırasında başlatılmadıkları için segfault hataları oluşmakta idi. buna çözüm olarak nesneler başlatılmadan önce nesnelere özel Inilizer fonkisonları yapılmıştır.

4. Program Çıktıları ve Debug Özelliği

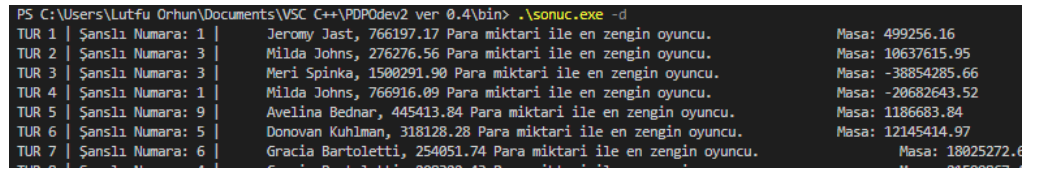
Program çalıştırılmadan önce “-d” parametresi verilirse, program ekrana UI çizimi yapmadan direkt çıktıları gösterir. Amacı daha hızlı çıktıları görmektir.



şekil 1.a: Normal ekran çıktısı



şekil 1.b: Debug parametre kullanımı



şekil 1.c: Debug ekran çıktısı

5. Sonuç

Yazılan uygulama istenen görevleri yerine getirmesi için yazılmıştır, ancak birkaç değişikliklerle bu kod birçok alanda kullanılabilir. Örnekler olarak, Online otomatik yazı moderatörlüğü, Online chat uygulamalarında filtre, Genel kelime arayıcısı, Yasaklı terimlerin kullanımı engelleme gösterilebilir.

6. Kullanılan Uygulamalar ve Kaynakça

- Visual Studio Code
Uygulama yazımı
- Adobe Illustrator 2021
Kapak Tasarımı
- Adobe Indesign 2020
Rapor Tasarımı
- Visual Studio Code C\C++ Plugin
Direkt derleme
- gcc.exe (Rev10, Built by MSYS2 project) 11.2.0
MinGW Derleyicisi
- <https://docs.microsoft.com/tr-tr/cpp/c-runtime-library/reference/setmode?view=msvc-170>
Konsol içinde Unicode çıktı almak için kullanılan _setmode kullanımının açıklaması için
- <https://www.rapidtables.com/code/text/unicode-characters.html>
Unicode karakterler tablosu. Ekran çıktısı için