

Taş Kağıt Makas

Proje Başlangıç Tarihi 27 Ekim 2022
Proje Bitiş Tarihi 19 Kasım 2022

1 Giriş

Proje ile beraber öğrencilerin nesneye yönelik programlama yapısını anlaması ve çözüm sağlayabilmesi amaçlanmaktadır.

Proje C++ veya JAVA dili kullanılarak gerçekleştirilecektir.

Görsel olarak oynanan bir oyun tasarlanacaktır. Görselleştirme için belirtilen programlama dillerine uygun kütüphaneler kullanılabilir.

2 Projenin İçeriği

Bir oyuncunun diğer bir oyuncuyla rekabet edebileceği taş kağıt makas oyunu tasarlanacaktır.

Oyun kullanıcı-bilgisayar ya da bilgisayar-bilgisayar şeklinde iki seçenekli olarak oynanabilmelidir. Oyunun başlangıcında seçim yapılmalıdır.

Tasarlanacak oyunda, her oyuncunun elinde oyunun başlangıcında taş, kağıt ve makas olmak üzere 3 farklı tipte olan nesnelerden toplam 5 nesnesi bulunacaktır.

Kullanıcı-bilgisayar oyununda kullanıcı nesne seçimini oyunun başında kendisi yapacak, bilgisayar ise rastgele bir şekilde nesneleri seçecektir.

Bilgisayar-bilgisayar oyununda tüm seçimler otomatik olarak yapılmalı ve oyunun adımları hızlı bir şekilde ilerlemelidir.

Örnekler:

1. Kullanıcı: taş, kağıt, makas, taş, taş

Bilgisayar: kağıt, kağıt, makas, taş, makas

2. Bilgisayar: makas, kağıt, makas, taş, taş

Bilgisayar: taş, kağıt, makas, taş, kağıt

Oyun sırasında kullanıcı bilgisayarın elindeki nesnelerin içeriğini görmemelidir. Ancak her seçim yapıldığında seçilen nesneler arayüzde görülebilmelidir. Kullanıcı elindeki kalan nesnelerden herhangi birini her hamle başında kendi seçebilmelidir.

Kullanıcı-bilgisayar oyununda oyun temelde şu şekilde ilerleyecektir: Önce kullanıcı elindeki nesneler arasından bir nesne seçimi yapar daha sonra bilgisayar rastgele bir seçim yapar en son nesneler karşılaştırılarak skorlar hesaplanır ve yeni seçimle oyun devam eder. Bilgisayar-bilgisayar oyununda her iki oyuncu içinde rastgele seçimlerle ilerler. Oyun aşağıdaki iki koşuldan biri sağlandığında sona erer.

- İki taraftan birinin elindeki nesneler tükendiğinde,
- Başlangıçta belirlenen toplam hamle sayısı tamamlandığı zaman, (başlangıç toplam hamle sayısı: 10).

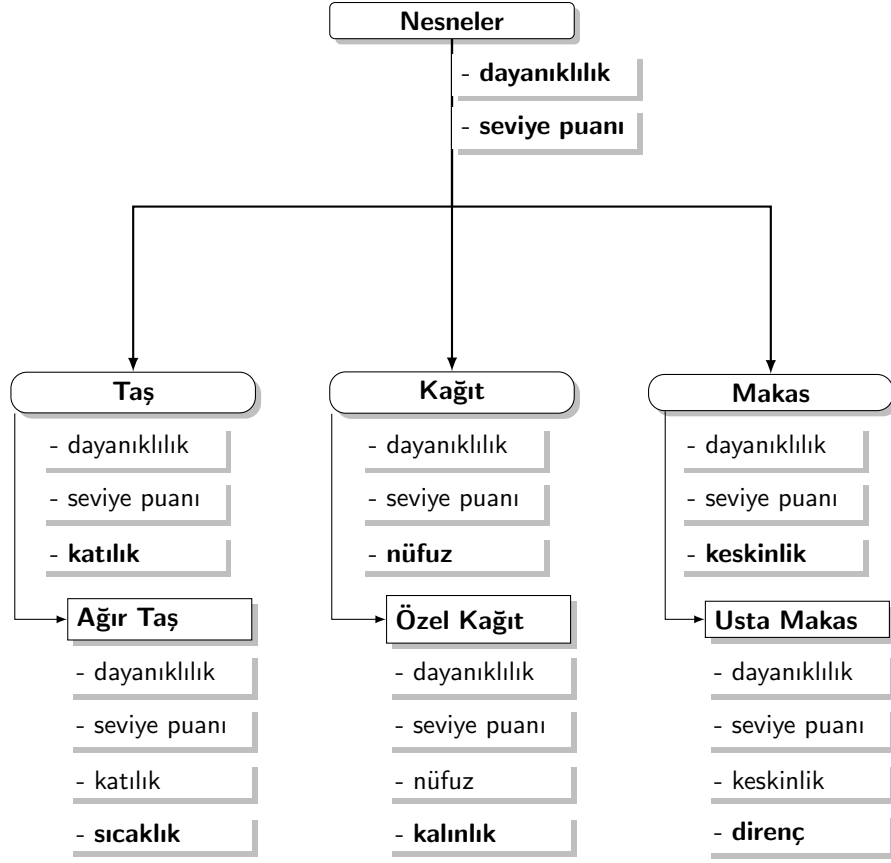
Bütün nesneler seçilmeden aynı nesne 2. defa seçilmemelidir. 5 nesnenin tamamı seçildiğinde aynı nesne tekrar seçilebilmelidir.

Örnek senaryo şu şekilde olabilir:

Hamle	Kullanıcı ya da Bilgisayar	Bilgisayar
1	taş1	kağıt1
2	taş2	taş1
3	kağıt	taş2
4	makas1	makas
5	makas2	kağıt2
6	kağıt	taş2
...

Oyunda kullanılan tüm nesneler için dayanıklılık ve seviye puanı özellikleri ortaktır.

Nesneler kendi içerisinde taş, kağıt ve makas olarak 3 alt sınıftan birine ait olmalıdır. Taşların diğer nesnelerden ayrı olarak katılık özelliği, kağıtlarda nüfuz özelliği ve makaslarda ise keskinlik özelliği yalnızca kendi sınıflarına ait özelliklerdir. Ayrıca bu 3 sınıfın bir de kendi içerisinde alt sınıfları bulunmaktadır. Bunlar; taş sınıfının altında ağır taş sınıfı, kağıt sınıfının altında özel kağıt ve makas sınıfının altında ise usta makas sınıfları olacaktır. Sırasıyla ağır taş için sıcaklık, özel kağıt için kalınlık ve usta makas için direnç özellikleri bu sınıflara özgü özelliklerdir. Aşağıda sınıflar ilişkisel olarak gösterilmektedir.



Kullanıcı ve bilgisayarın seçmiş olduğu nesnelerin karşılaştırılması ve skorların hesaplanması aşağıdaki formüllere göre yapılacaktır. Buna göre bir hamlede karşı karşıya gelen iki nesnenin birbirine karşı etkileri ayrı ayrı hesaplanmalıdır. Sonrasında bir nesnenin etki miktarı kadar rakip nesnenin dayanıklılık değeri azaltılacaktır. Dayanıklılık değeri 0 veya 0'ın altına düşen nesneler oyundan elenecektir. Bir nesneyi yenen rakip nesnenin seviye puanını 20 artırılacaktır.

Seviye puanı 30'un üstüne ulaşan nesneler terfi edeceklerdir. (Taş nesnesi ağır taş nesnesine, kağıt nesnesi özel kağıt nesnesine, makas nesnesi usta makas nesnesine dönüşecektir)

Oyunculardan birinin elindeki tüm nesneler tükendiğinde rakip oyuncu oyunu kazanacaktır. Eğer maksimum 10 hamle (sunum sırasında farklı değerler için denenebilmeli) sonrasında her iki oyuncunun da elinde en az bir nesne varsa bu durumda oyunculardan elindeki nesnelerin toplam dayanıklılık miktarı en fazla olan kazanacak ve bu toplam dayanıklılık değeri kazananın skorunu verecektir.

Kullanıcı ve bilgisayarın seçtiği nesneler için aşağıdaki formüllere göre etki puanları hesaplanacaktır. Her bir formüldeki pay kısmı atak yapan tarafı, payda ise rakip tarafın değerlerine karşılık gelmektedir. Süslü parantezin içindeki değişkenler o nesnenin özellikleri arasında varsa kullanılmalıdır aksi durumda hesaba katılmamalıdır.

Örneğin taş ve kağıt karşılaştırılıyorsa, taş için taş_etkisi, kağıt için kağıt_etkisi hesaplanmalıdır. Taş nesnesi için pay kısmında taşın katılık değeri alınır ancak taş ağır taşa dönüşmüşse katılık değeri sıcaklık değeri ile çarpılır. Payda kağıdın değerlerini temsil eder. Kağıdın keskinlik değişkeni olmadığı için 0 alınır ve nufuz değeri kağıdın kendi değeri olarak alınır. Kağıt özel kağıt ise kalınlık değeri ile çarpılır. a değeri 0,2 olarak alınmalıdır. Pay paydaya bölündüğünde taş nesnesi için kağıda karşı etki değeri hesaplanmış olur. Benzer hesaplama kağıt_etkisi için de hesaplanır ve bu iki değer rakiplerin dayanıklılık değerlerinden eksiltir.

$$makas_etkisi = \frac{keskinlik\{*direnc\}}{(a * nufuz_r\{*kalinlik_r\} + (1 - a) * katilik_r\{*sicaklik_r\})} \quad (1)$$

$$kagit_etkisi = \frac{nufuz\{*kalinlik\}}{(a * katilik_r\{*sicaklik_r\} + (1 - a) * keskinlik_r\{*direnc_r\})} \quad (2)$$

$$tas_etkisi = \frac{katilik\{*sicaklik\}}{(a * keskinlik_r\{*direnc_r\} + (1 - a) * nufuz_r\{*kalinlik_r\})} \quad (3)$$

Formüldeki değerler için her nesnenin kendi özellikleri başlangıçta 2 olarak alınabilir. Başlangıç dayanıklılık değeri tüm nesneler için 20, başlangıç seviye puanı 0'dır.

3 Sınıflar

3.1 Nesne Sınıfı

Bir nesne sınıfı oluşturun. Sınıf tanımı şunları içermelidir:

- Bu sınıf abstract class yapısıyla tanımlanmalıdır.
- Yapıcı (constuctor) metotları (parametrelili ve parametresiz olarak en az iki) yazılmalıdır. Parametreler dayanıklılık ve seviyePuanı olmalı.
- Nesnelerin dayanıklılık ve seviye puanı özelliklerini oyunda göstermek için nesnePuanıGoster metodu yazılmalıdır.
- Nesnelerin rakip nesneye karşı atak etkisini hesaplamak ve için etkiHesapla metodu yazılmalıdır.
- Nesnelerin karşılıklı atakları sonucunda etki değerleri kadar dayanıklılık değerlerinin azaltılması ve seviye puanı güncellemeleri için durumGuncelle metodu yazılmalıdır.

3.2 Taş Sınıfı

Taş nesneleri taş sınıfı üzerinden türetilmelidir ve sınıf şu tanımlamaları içermelidir:

- Nesne sınıfından kalıtım alacaktır.
- Yapıcı (constuctor) metotları (parametrelili ve parametresiz olarak en az iki) yazılmalıdır. Taş sınıfında bulunan dayanıklılık ve seviyePuanı özelliklerine atama yapmak için super() kullanılmalıdır.
- Bu sınıfın ayrıca katılık özelliği olmalıdır.
- nesnePuanıGoster, etkiHesapla ve durumGuncelle metodları override edilerek özelleştirilecektir.

3.3 Ağır Taş Sınıfı

Ağır Taş nesneleri ağır taş sınıfı üzerinden türetilmelidir ve sınıf şu tanımlamaları içermelidir:

- Taş sınıfından kalıtım alacaktır.
- Yapıcı (constuctor) metotları (parametrelili ve parametresiz olarak en az iki) yazılmalıdır. Ağır Taş sınıfında bulunan dayanıklılık, seviyePuani ve katılık özelliklerine atama yapmak için `super()` kullanılmalıdır.
- Bu sınıfın ayrıca sıcaklık özelliği olmalıdır.
- `nesnePuaniGoster`, `etkiHesapla` ve `durumGuncelle` metodları `override` edilerek özelleştirilecektir.

3.4 Kağıt Sınıfı

Kağıt nesneleri Kağıt sınıfı üzerinden türetilmelidir ve sınıf şu tanımlamaları içermelidir:

- Nesne sınıfından kalıtım alacaktır.
- Yapıcı (constuctor) metotları (parametrelili ve parametresiz olarak en az iki) yazılmalıdır. Kağıt sınıfında bulunan dayanıklılık ve seviyePuani özelliklerine atama yapmak için `super()` kullanılmalıdır.
- Bu sınıfın ayrıca nüfuz özelliği olmalıdır.
- `nesnePuaniGoster`, `etkiHesapla` ve `durumGuncelle` metodları `override` edilerek özelleştirilecektir.

3.5 Özel Kağıt Sınıfı

Özel Kağıt nesneleri ÖzelKağıt sınıfı üzerinden türetilmelidir ve sınıf şu tanımlamaları içermelidir:

- Kağıt sınıfından kalıtım alacaktır.
- Yapıcı (constuctor) metotları (parametrelili ve parametresiz olarak en az iki) yazılmalıdır. Özel Kağıt sınıfında bulunan dayanıklılık, seviyePuani ve nüfuz özelliklerine atama yapmak için `super()` kullanılmalıdır.
- Bu sınıfın ayrıca kalınlık özelliği olmalıdır.
- `nesnePuaniGoster`, `etkiHesapla` ve `durumGuncelle` metodları `override` edilerek özelleştirilecektir.

3.6 Makas Sınıfı

Makas nesneleri makas sınıfı üzerinden türetilmelidir ve sınıf şu tanımlamaları içermelidir:

- Nesne sınıfından kalıtım alacaktır.
- Yapıcı (constuctor) metotları (parametrelili ve parametresiz olarak en az iki) yazılmalıdır. Makas sınıfında bulunan dayanıklılık ve seviyePuani özelliklerine atama yapmak için `super()` kullanılmalıdır.
- Bu sınıfın ayrıca keskinlik özelliği olmalıdır.
- `nesnePuaniGoster`, `etkiHesapla` ve `durumGuncelle` metodları `override` edilerek özelleştirilecektir.

3.7 Usta Makas Sınıfı

Usta makas nesneleri UstaMakas sınıfı üzerinden türetilmelidir ve sınıf şu tanımlamaları içermelidir:

- Makas sınıfından kalıtım alacaktır.
- Yapıcı (constuctor) metotları (parametrelili ve parametresiz olarak en az iki) yazılmalıdır. Usta makas sınıfında bulunan dayanıklılık, seviyePuani ve keskinlik özelliklerine atama yapmak için `super()` kullanılmalıdır.
- Bu sınıfın ayrıca hız özelliği olmalıdır.
- `nesnePuaniGoster`, `etkiHesapla` ve `durumGuncelle` metodları override edilerek özelleştirilecektir.

3.8 Oyuncu Sınıfı

Bilgisayar ve kullanıcı olmak üzere oyunu oynayan iki oyuncu olacaktır. Bu iki oyuncunun farklı ve aynı özellikleri olacaktır. Aynı özelliklerini temsil etmek için Oyuncu temel sınıfı oluşturulacaktır.

Bu sınıfta bulunması gereken özellikler ve fonksiyonlar:

- Bu sınıf abstract class yapısıyla tanımlanmalıdır.
- `oyuncuID`, `oyuncuAdi` ve `skor` özellikleri olmalı.
- Yapıcı (constuctor) metotları (parametrelili ve parametresiz olarak en az iki) yazılmalıdır. Parametreler `oyuncuID`, `oyuncuAdi` ve `skor` olmalı.
- `nesneListesi` özelliği ile oyuncuların elinde bulunan nesneler listede tutulacaktır.
- `SkorGoster` fonksiyonu ile oyuncuların skorları gösterilecektir.
- `nesneSec` fonksiyonu yazılmalı fakat bu sınıf bilgisayar ve kullanıcı için farklı durumlarda çalışacağı unutulmamalıdır.

3.9 Bilgisayar Sınıfı

- Oyuncu sınıfından kalıtım alacaktır.
- Yapıcı (constuctor) metotları (parametrelili ve parametresiz olarak en az iki) yazılmalıdır. Oyuncu sınıfında bulunan `oyuncuID`, `oyuncuAdi` ve `Skor` özelliklerine atama yapmak için `super()` kullanılmalıdır.
- Oyuncu sınıfında bulunan `nesneSec` metodu override edilecektir. Bilgisayar başta random olarak aldığı nesneler arasından yine random nesne seçerek hamle yapacaktır.

3.10 Kullanici Sınıfı

- Oyuncu sınıfından kalıtım alacaktır.
- Yapıcı (constuctor) metotları (parametrelili ve parametresiz olarak en az iki) yazılmalıdır. Oyuncu sınıfında bulunan oyuncuID, oyuncuAdi ve Skor özelliklerine atama yapmak için super() kullanılmalıdır.
- Oyuncu sınıfında bulunan nesneSec metodu override edilecektir. Kullanıcı başta seçtiği nesneler arasından yine kendi istediği nesneyi seçerek hamle yapacaktır.

3.11 Oyun Sınıfı

main metodu bu sınıf içerisinde yer alacaktır. Nesnelerin oluşturulması ve özelliklerinin atanması bu sınıf içerisinden yapılmalıdır.

Her sınıf için ortak olan özellikler:

- Projede Encapsulation, Inheritance, Polymorphism, Abstraction yapılarının ihtiyaç olanlarının mutlaka kullanılması gerekmektedir.

4 Özet

- Sınıf tanımlamaları, değişken tanımlamaları, override edilmesi gereken yöntemler, get ve set metotları, yapılandırıcılar (Constructor), çağırılması gereken yöntemler vb. belirtilen kurallar eksiksiz olmalıdır.
- Oyun dinamik bir şekilde ilerlemelidir.
- Dayanıklılık ve seviye puanı hesaplamaları, oyuncuların her adımdaki etki puanları, seviye atlama aşaması, nesnelerin eksildiği adımlar ve diğer bütün adımlar anlaşılır bir şekilde bir dosyada gösterilmelidir.
- Oyun yukarıda anlatılan tüm kurallara uygun çalışmalıdır.
- NYP mantığına uygun olmayan projeler eksik değerlendirilecektir.

Notlandırma

Aşağıdaki notlandırma bilgisi projeyi yaparken planlama yapabilmeniz açısından paylaşılmıştır.

Sunumda sorulara verilen cevaplar, kod hakimiyeti ve dokümanda belirtilen maddeler projenin genel kriterleri içerisinde değerlendirilecektir.

NYP: Tanımlanması gereken tüm yapı, içerikler ve NYP'ye uygunluğu.....	50
Görsellik: Oyunun görsel tasarımı ve doğru çalışması	35
Rapor: UML, İÇERİK, kazanımlar, yorumlar	15

Toplam: 100

5 Proje Teslimi

Rapor LATEX kullanılarak yazılmalıdır.

UML diyagramı çizilmelidir. UML diyagramı olmayan raporlar eksik kabul edilecektir.

Rapor IEEE formatında en az 4 sayfa (times new roman 12 pt), özet, giriş, yöntem (detaylı), tablo, katkılar, sonuç, kaynakça bölümünden oluşmalıdır.

Raporda her bir grup üyesinin katkıları belirtilmelidir.

Proje ile ilgili sorular edestek2.kocaeli.edu.tr sitesindeki forum üzerinden Arş. Gör. Abdurrahman Gün veya Arş. Gör. Furkan Göz'e sorulabilir.

Demo tarihleri daha sonra duyurulacaktır.

Demo sırasında algoritma, geliştirdiğiniz kodun çeşitli kısımlarının ne amaçla yazıldığı ve geliştirme ortamı hakkında sorular sorulabilir.

Kullandığınız herhangi bir satır kodu açıklamanız istenebilir.

Proje grupları her öğretim kendi içinde olmak üzere en fazla iki kişiden oluşturulmalıdır.