

SPORCU KART OYUNU

Bir oyuncunun (bu kullanıcı olacak) otomatik oyuncuyla (bilgisayara karşı) rekabet edebileceği basit bir kart oyunu tasarlanacaktır.

Tasarlanacak oyunda, toplamda 16 sporcu kartı bulunacaktır. Bu kartlardan 8'i futbolcu, 8'i basketbolcu olmalıdır.

Her bir kullanıcıya rastgele 4 basketbolcu, 4 futbolcu kartı dağıtılacaktır.

Futbolcuların penaltı, serbest vuruş ve kaleciyle karşı karşıya özellikleri bulunmaktadır.

Basketbolcuların üçlük, ikilik ve serbest atış özellikleri bulunmaktadır.

Kullanıcı ve bilgisayar kendilerine dağıtılan 8 karttan her hamlede birini seçerek ortaya koyacaklardır. Her hamle ardışık sırayla bir futbolcu bir basketbolcu kartlarıyla oynanacak şeklinde olmalıdır.

Hamle	Kullanıcı	Bilgisayar
1	Futbolcu	Futbolcu
2	Basketbolcu	Basketbolcu
3	Futbolcu	Futbolcu
4	Basketbolcu	Basketbolcu
5	Futbolcu	Futbolcu
6	Basketbolcu	Basketbolcu
7	Futbolcu	Futbolcu
8	Basketbolcu	Basketbolcu
.....

Her oyuncu öncelikle birbirlerinin kartlarını görmeden bir kart seçmelidir. Kart seçimi yapıldıktan sonra hamle sırası futbolcudaysa futbolcu, basketbolcudaysa basketbolcu için pozisyon bilgisi alınmalıdır.

Her iki oyuncunun ortaya koyduğu iki kart her hamlenin sonunda karşılaştırılacaktır. Her iki oyuncunun kartı pozisyon bilgisine göre yüksek değerli karta sahip kullanıcıya 10 puan kazandıracaktır.

Değerler aynıysa kartlar geri alınmalıdır. Bu durumda hamle sırası diğer sporcu özelliğinden (futbolcuysa basketbolcuya, basketbolcuysa futbolcuya) devam etmelidir.

Eldeki kartlar bitene kadar oyun devam edecektir.

Eğer oyuncuların ortaya koydukları ellerindeki **son kartlarda** pozisyon bilgisine göre aynı puandaysa rastgele farklı bir pozisyon olarak yeni karşılaştırma yaparak ilerlenir. Üç pozisyon bilgisi aynıysa eldeki kartlar ile oyun sonlanır.

Amaç: Proje gerçekleştirimi ile beraber öğrencilerin nesneye yönelik programlama yapısını anlamasını ve çözüm sağlayabilmesini amaçlamaktadır.

Programlama Dili: Proje C++, C# veya Java dili kullanılarak gerçekleştirilecektir. Görselleştirme için belirtilen programlama dillerine uygun kütüphaneler kullanılabilir.

ÖZET

Oyunu ve oyuncuların ellerinde bulunan kartların görülebileceği ve takip edilebileceği bir arayüz tasarlanmanız beklenmektedir.

- Destede 16 adet sporcu kartı bulunmalıdır.
- Oyuncu desteden 8 (4 futbolcu, 4 basketbolcu) rastgele kart alır.
- Bilgisayar desteden geri kalan diğer 8 kartı alır.
- Hamle sırasında bilgisayar elinde bulunan 8 karttan birini rastgele seçerken kullanıcı kendi elindeki kartlardan istediği birini seçebilir. Burada kullanıcı bilgisayarın hangi kartı seçtiğini bilmemeli ve bilgisayarın elindeki kartlar kullanıcıya kapalı görünmelidir. Futbolcu kartı seçilmesi gerekirken basketbolcu kartının seçilmesine ya da tersine izin verilmemelidir. Bilgisayar kullanıcısı da aynı hatayı yapmamalıdır.
- Daha sonra ortaya konan kartlar açık bir şekilde gösterilir.
- Kartın seçilen pozisyon puanları aynıysa kartlar geri alınıp yeniden sıradaki sporcu için seçim yapılmalıdır.
- Kazanılan bir hamle içerisinde oyuna sürülen kartlar tekrar kullanılamayacaktır. Ancak son kartlarda üstünlük sağlanamaması durumunda yeni değerlendirme yapılmalıdır.
- Oyun tüm kartlar bitene kadar devam edecektir.
- Tüm kartlar bittikten sonra skoru yüksek olan oyuncu oyunu kazanır.

Sporcu Sınıfı

Bir sporcu sınıfı oluşturun. Sınıf tanımı şunları içermelidir:

- Yapıcı (constructor) metotları (parametrelili ve parametresiz olarak en az iki) yazılacak. Parametreler sporcuİsim ve sporcuTakım olmalı.
- Sporcuların kart puanını göstermek için sporcuPuanıGoster() metodu yazılmalıdır.

Futbolcu Sınıfı

Futbolcuların her biri birer sınıf olarak tanımlanacaktır. Sınıflar şu tanımlamaları içermelidir:

- **Sporcu sınıfından kalıtım alacaktır.**
- Yapıcı (constructor) metotları (parametrelili ve parametresiz olarak en az iki) yazılacak. Futbolcu sınıfında bulunan futbolcuAdi ve futbolcuTakım özelliklerine atama yapmak için super() kullanılacaktır.
- Bu sınıfın penaltı, serbestAtis ve kaleciKarsiKarsiya özellikleri olmalıdır.
- sporcuPuanıGoster() metodu override edilerek her bir futbolcu kartı için özelleştirilecektir.
- boolean veri tipinde kartKullanildiMi bilgisi tutulmalıdır.

Basketbolcu Sınıfı

Basketbolcuların her biri birer sınıf olarak tanımlanacaktır. Sınıflar şu tanımlamaları içermelidir:

- **Sporcu sınıfından kalıtım alacaktır.**
- Yapıcı (constuctor) metotları (parametrelili ve parametresiz olarak en az iki) yazılacak. Futbolcu sınıfında bulunan basketbolcuAdi ve basketbolcuTakim özelliklerine atama yapmak için super() kullanılacaktır.
- Bu sınıfın ikilik, ucluk ve serbestAtis özellikleri olmalıdır.
- sporcuPuaniGoster() metodu override edilerek her bir futbolcu kartı için özelleştirilecektir.
- Boolean veri tipinde kartKullanildiMi bilgisi tutulmalıdır.

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:

- oyuncuID, oyuncuAdi ve Skor özellikleri olmalı.
- Yapıcı (constuctor) metotları (parametrelili ve parametresiz olarak en az iki) yazılacak. Parametreler oyuncuID, oyuncuAdi ve Skor olmalı.
- kartListesi özelliği ile oyuncuların elinde bulunan kartlar listede tutulacaktır.
- SkorGoster() fonksiyonu ile oyuncuların skorları gösterilecektir.
- kartSec() fonksiyonu yazılmalı fakat bu sınıf bilgisayar ve kullanıcı için farklı durumlarda çalışacağı unutulmamalıdır.

Bilgisayar Sınıfı

- **Oyuncu sınıfından kalıtım alacaktır.**
- Yapıcı (constuctor) metotları (parametrelili ve parametresiz olarak en az iki) yazılacak. Oyuncu sınıfında bulunan oyuncuID, oyuncuAdi ve Skor özelliklerine atama yapmak için super() kullanılacaktır.
- Oyuncu sınıfında bulunan kartSec() metodu override edilecektir. Bilgisayar random olarak aldığı kartlar arasından yine random kart seçerek ortaya koyacaktır.

Kullanıcı Sınıfı

- **Oyuncu sınıfından kalıtım alacaktır.**
- Yapıcı (constuctor) metotları (parametrelili ve parametresiz olarak en az iki) yazılacak. Oyuncu sınıfında bulunan oyuncuID, oyuncuAdi ve Skor özelliklerine atama yapmak için super() kullanılacaktır.
- Oyuncu sınıfında bulunan kartSec() metodu override edilecektir. Kullanıcı random olarak aldığı kartlar arasından **kendi istediği kartı** seçerek ortaya koyacaktır.

Test Sınıfı

Futbolcu ve basketbolcuların isimleri ve özellikleri burada tanımlanmalıdır. Pozisyon bilgisi için bir metot içermelidir.

Futbolcu seçimi yapılacağı zaman penaltı, serbest vuruş ve kaleciyle karşı karşıya pozisyonlarından; basketbolcu seçimi yapılacağı zaman üçlük, ikilik ve serbest atış pozisyonlarından **rastgele** birinin seçimi oyuncuların kart seçimi yapıldıktan sonra burada yapılmalıdır.

Futbolcu ve basketbolcuların özelliklerin ataması için getter ve setter metotları da, Constructor'da kullanılabilir.

Oyunun ilerleyişi buradan takip edilmelidir.

NOT: Oyuncu isimleri ve özelliklerini dilediğiniz gibi tanımlayabilirsiniz. Özelliklerin değerlerini oyunu test edecek parametreleri düşünerek planlamanız gerekmektedir.

ÖRNEKLER:

Futbolcu seçimi hem bilgisayar hem kullanıcı tarafından yapılır. Cristiano Ronaldo ve Lionel Messi seçilmiş olsun.

Cristiano Ronaldo: Penaltı: 95, Serbest Vuruş: 80 Kaleciyle karşı karşıya: 90

Lionel Messi: Penaltı: 100, Serbest Vuruş: 75 Kaleciyle karşı karşıya: 90

Rastgele pozisyon seçilir. Kaleciyle karşı karşıya seçilmiş olsun.

Kartlar açılır.

SONUÇ: Kimse puan alamaz. Kartlar geri alınır.

Basketbolcu seçimi hem bilgisayar hem kullanıcı tarafından yapılır. Lebron James ve Stephen Curry seçilmiş olsun.

Lebron James: İkilik: 85, Üçlük: 80 Serbest Atış: 95

Stephen Curry: İkilik: 85, Üçlük: 95 Serbest Atış: 90

Rastgele pozisyon seçilir. Üçlük seçilmiş olsun.

Kartlar açılır.

SONUÇ: Stephen Curry kartını atan oyuncu puan alır.

Her sınıf için ortak olan özellikler:

- **Projede Encapsulation, Inheritance, Polymorphism, Abstraction yapılarından ihtiyaç olanlarının mutlaka kullanılması gerekmektedir.**
- **Yapıcı (constuctor) metotları (parametrelili ve parametresiz olarak en az iki şekilde) yazılmalıdır.**
- **Tüm özellikler için get, set metotları tanımlanmalıdır.**