

GAMEZ OYUN İŞ AKIŞI

Karelere ayrılmış bir oyun tahtasında (satranç tahtası gibi) bir veya birden fazla oyuncu karelerde yer alan paraları toplayarak en yüksek kazanca ulaşmaya çalışacak.

Programınızda Oyuncu ve Tahta adında iki adet sınıf bulunmalıdır. Bu sınıflara ait UML sınıf diyagramları aşağıda verilmiştir. Bazı metotların ne yapması gerektiği ise diyagramların altında belirtilmiştir.

Oyuncu
+oyuncuAdi : String
-x : int
-y : int
-toplamPuan : int
+Oyuncu(String,int,int)
+getX() : int
+setX(int) : void
+getY() : int
+setY(int) : void
+getToplamPuan() : int
+setToplamPuan(int) : void
+puanEkle(int) : void
+hareketEt(char,int,int) : void

Tahta
-oyunMatrisi : int[][]
-oyuncular : Oyuncu[]
+Tahta(int,int)
+getOyunMatrisi() : int[][]
+setOyunMatrisi(int[][]) : void
+getOyuncular() : Oyuncu[]
+setOyuncular(Oyuncu[]) : void
+oyunBittiMi() : boolean
+oyunuOynat() : void
+tahtayıYazdir() : void

Oyuncu.puanEkle(int) : Dışarıdan aldığı parametre değerini toplamPuan değerine ekler.

Oyuncu.hareketEt(char,int,int) : Dışarıdan aldığı yön bilgisini (b, k, d, g – Batı, Kuzey, Doğu ve Güney olarak) kullanarak oyuncunun mevcut koordinatlarını değiştirir. Dışarıdan alınan iki int parametresi ise oyunun oynandığı tahtadaki satır ve sütun sayısıdır. Bu parametreler kullanılarak belirtilen yöne doğru bir hareket yapılmalıdır. Eğer belirtilen yöne doğru bir hareket geçerli bir hareket değilse (tahtanın dışına çıkılması) oyuncu hareket etmemeli ve eski yerinde kalmalıdır.

Tahta.Tahta(int,int) : Parametre olarak gelen iki int değişkeni tahtanın satır ve sütun sayısını belirtir. Bu satır ve sütun sayısına göre tahtanın matrisi oluşturulmalıdır. Tahtadaki karelerdeki değerler ise 1 ile 9 arasında üretilmiş rastgele sayılar olmalıdır.

Tahta.setOyuncular(Oyuncu[]) : Parametre olarak gelen Oyuncu dizisini oyuncular değişkenine atar. Bunun yanında oyuncuların tahtada başladıkları ilk yerdeki kare değerlerinin 0 olması gerekmektedir. Bu metot içerisinde bu işlemin de yapılması gerekir. Örneğin bir oyuncu (0,0) noktasında oyuna başlayacaksa ve tahta oluşturulurken (0,0) için üretilen değer 5 ise bu değer 0 olarak güncellenmelidir.

Tahta.oyunBittiMi() : Oyun oynanmaya başladıktan sonra tahtanın durumu sürekli değişecektir. Ancak oyun bir noktada bitmelidir. Bu metot tahtanın mevcut durumunu kontrol edip oyunun bitip bitmediğine karar verir. Oyunun bitme koşulu ise tahta üzerinde alınabilecek hiç para kalmamasıdır.

Tahta.tahtayıYazdir() : Tahtanın mevcut durumunu anlaşılır bir şekilde ekrana yazdıran metot. Bu metot tahtanın iki boyutlu görünümünü düzgün bir şekilde ekrana yazdırır ve oyuncuların puan durumlarını da tahtanın altına yazar.

Tahta.oyunuOynat() : Oyunu oynatan asıl metot. Tahtadaki paralar ve oyuncuların yerleri ayarlandıktan sonra oyunu başlatabilirsiniz. Bu metot algoritmik olarak aşağıdaki gibi çalışmalıdır.

1) Oyun sırası hangi oyuncudaysa o oyuncuya hangi yöne gitmek istediğini sor.

2) Gelen yön bilgisine göre oyuncuyu hareket ettir.

3) Oyuncunun yeni yerindeki parayı oyuncunun toplam puanına ekle ve o karedeki para değerini 0 yap.

4) Oyun bitti mi diye kontrol et. Oyun bittiyse Adım 5'e geç. Oyun bitmediyse oyuncu sırasını değiştir ve Adım 1'e geç.

5) Oyunun sonucunu ekrana yazdır.

PROGRAM KISITLARI ve KURALLAR

1) Kullanıcının yön bilgisini String olarak gireceği varsayılacak.

2) Kullanıcının ekrandan yanlış veri girişini kontrol etmenize gerek yok. Kullanıcının her zaman doğru veri türünü girdiğini varsayabilirsiniz.

3) Ancak kullanıcının girdiği yön bilgisinin tutarlı bir yön olup olmadığını kontrol etmelisiniz. Eğer tutarsız bir yön bilgisi girildiyse kullanıcıya hata mesajı vermeli ve kullanıcının hiç bir hareket yapmadan yerinde kalmasını sağlayarak sırayı diğer oyunculara geçirmelisiniz.

4) Birden fazla oyuncu aynı karede yer alabilir. Böyle bir durumda oyunculardan yalnızca birinin adını o kareye yazdırmanız yeterli.

5) Sınıflarınızın başına birkaç yorum satırında adınızı ve numaranızı lütfen belirtin.

6) Kodunuza yorum satırları eklemelisiniz. Yazdığınız metotları kısaca açıklamalısınız.

ÖRNEK PROGRAM AKIŞI

A 7 3 7

9 9 5 4

6 9 5 1

2 7 8 B

Oyuncu A: 0

Oyuncu B: 0

Sayın A, lütfen hareket yönünüzü seçiniz (b, k, d, g):

bööö

Geçersiz bir yön girdiniz!.. Yerinizde kaldınız!..

A 7 3 7

9 9 5 4

6 9 5 1

2 7 8 B

Oyuncu A: 0

Oyuncu B: 0

Sayın B, lütfen hareket yönünüzü seçiniz (b, k, d, g): g

Belirttiğiniz yöne gidemezsiniz!.. Yerinizde kaldınız!..

A 7 3 7

9 9 5 4

6 9 5 1

2 7 8 B

Oyuncu A: 0

Oyuncu B: 0