

# ALTIN TOPLAMA OYUNU

Faruk Arıç      Sefa Öztürk  
180202009      180202036

Kocaeli Üniversitesi Bilgisayar Mühendisliği Bölümü  
Yazılım Lab.1 – Proje 1

## 1.Giriş

Altın toplama oyununda amaç; x ve y eksenini kullanıcından alınan bir board üzerindeki birbirinden farklı 4 adet oyuncunun, oyun tahtası üzerindeki altınları toplamasıdır.

## 2.Özet

Altın toplama oyununu özetlemek gerekirse 4 adet oyuncu kendine özel algoritmalar ile board üzerindeki altınları aramaktadır. Board üzerinde random olarak iki çeşit altın dağıtılır. Bu altınlardan bir çeşidi her oyuncunun görebileceği normal altınlar; diğer çeşit ise yalnızca C oyuncusunun görebileceği ya da belli durumlarda gözükebilecek gizli altınlardır. Bu 4 oyuncu sırayla kendine özel algoritmalar ile altınları bulmaya çalışır. Board üzerindeki altın bitince ya da tüm oyuncuların altını tükendiğinde oyun sonlanmaktadır.

## 3.Yöntem

### 3.1 Proje geliştirme ortamı

Projemiz Java Programlama dilinde NetBeans IDE üzerinde yazılmıştır. Arayüz tasarımı için JFrame, JLabel kullanılmıştır.

### 3.2 Oyun tahtası oluşturma yöntemi

Oyun tahtasını oluştururken JFrame Class'ı ve matrisler kullanılmıştır. JButton classından üretilen buton nesneleri, iki boyutlu matrisin Her bir indexine atanır. Dolayısıyla kullanıcının başlangıç ekranında gelen default boyut, altın ve Altın miktarı gibi verilen değerler ile oyuna start verilmesi ile oyun başlar. Math.random() fonksiyonu ile rastgele altın bulunan matrisler üretilir.

### 3.3 Oyun tahtasına altın atama yöntemi

Oyun tahtasına altın atamak için iki adet matris ürettik. Bu matrislerden bir tanesi altınların türünü tutmaktadır. Yani hangi altın gizli hangi altın görünür olacak bilgisi birinci matriste tutulur. İkinci matrisin amacı ise dağıtılan altınların miktarını tutmaktır. Birinci matriste, matrisin değeri if(isHidden) ise o indexteki altın gizli altındır. Eğer matrisin değeri else ise indexteki altın gizli altındır.

```
public void createRandomGold(boolean isHidden){
    int randomY = new Random().nextInt(settings.sizeY);
    int randomX = new Random().nextInt(settings.sizeX);

    if(
        (randomY==0 && randomX==settings.sizeX-1) ||
        (randomY==settings.sizeY-1 && randomX==0) ||
        (randomY==settings.sizeY-1 && randomX==settings.sizeX-1) ||
        (randomY==0 && randomX==0)
    ){
        createRandomGold(isHidden);
        return;
    }
}
```

Resim 1 : Altın oluşturma işlemi

### 3.4 Oyuncu hareket ettirme yöntemi

#### 3.4.1 A oyuncusu için hareket ettirme

A,B,C,D oyuncuları en başta matrisin köşelerine yerleştirilir. İki boyutlu matrisi koordinat sistemi olarak düşünürsek; iki nokta arası en kısa mesafe aşağıdaki formül ile bulunur.

$$\sqrt{((x_2 - x_1)^2 + (y_2 - y_1)^2)}$$

Resim 2 : İki nokta arası mesafe bulma formülü

İç içe for döngüsü ile bütün matris dolaşılır. Tüm uzaklıklar bir diziyeye aktarılır ve daha sonra o dizideki en küçük değer en kısa mesafe olarak kabul edilir.

### 3.4.2 B oyuncusu için hareket ettirme

B oyuncusu A'dan farklı olarak altın miktarlarını eldeki uzaklığa böler ve altın/mesafe oranı en karlı olan altına doğru ilerler. Bazı durumlarda altın/mesafe oranı eşit olan iki değer bulabilir. Bu durumda ilk döndürülen değere gider.

### 3.4.3 C oyuncusu için hareket ettirme

C oyuncusu için  $matris[i][j]$  değerlerini tek tek dolaşır her el bunu tekrarlar. Buna göre kendine hedef belirler. Burada amaç gizli olan altını görünür hale getirmektir. Daha sonra C oyuncusu bütün altınlar içerisinde B oyuncusu gibi en karlı olan hedefe doğru ilerler.

### 3.4.4 D oyuncusu için hareket ettirme

D oyuncusu diğer oyuncuların yapacağı hamleleri önceden sezebilmektedir. Bundan dolayı ki diğer oyuncuların hedeflediği altınları onlardan önce ulaşamıyorsa o altınları hedef listesine eklemeyi hedef olarak geriye kalan altınlardan en karlı olana doğru ilerler.

## 3.5 Yalancı Kod

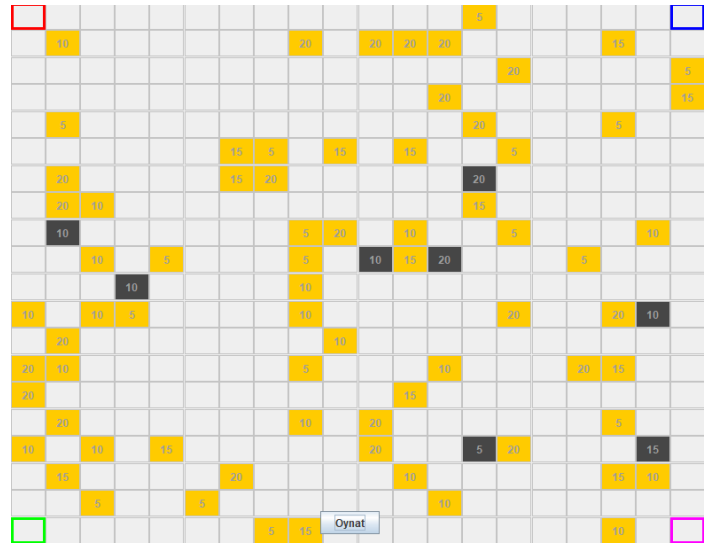
1. Başla
2. Kullanıcıdan x ve y eksenini al ya da default oyun ayarlarıyla oynat.
3. Başlat butonuna bas.
4. Random altınları matris değerleri default ekrana göre dağıt
5. Random altın değerlerini (5,10,15,20) olarak dağıt.
6. Her hamle için oynata bas.
7. A oyuncusunu oynat
8. B oyuncusunu oynat
9. C oyuncusunu oynat
10. D oyuncusunu oynat
11. Altın bitmediyse 6.adıma geri dön
12. Altın bittiyse oyunu bitir
13. Oyuncuların değerlerini uygulama ekranında yazdırmak için istatistikleri getir butonuna bas.
14. Bitir .

## 4. Deneyisel Sonuçlar

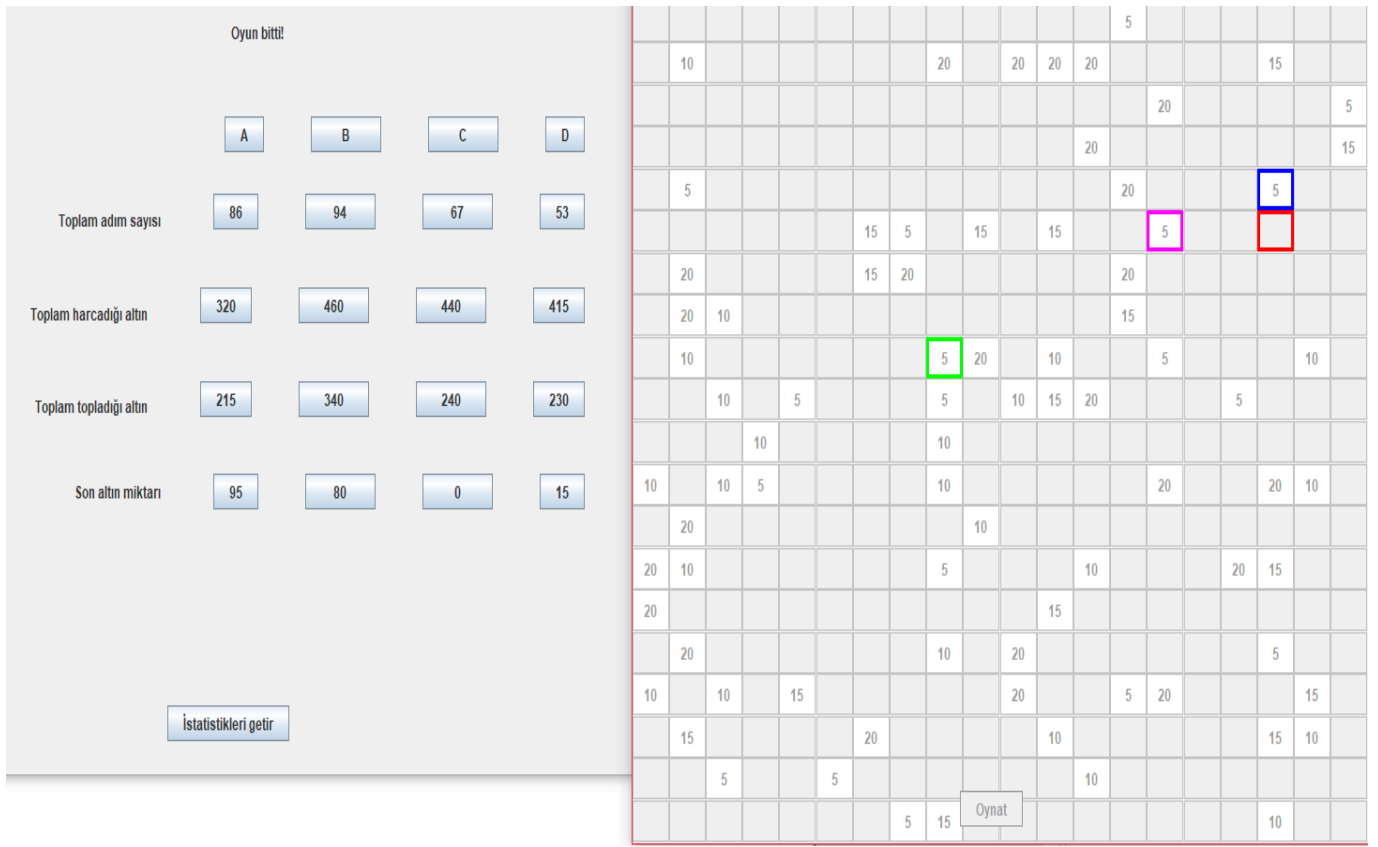
The screenshot shows a game settings window titled 'Altın Toplama Oyunu'. It contains four sections for player settings, each with a red, blue, green, and purple border respectively. Each section has two input fields: 'Hamle Maliyeti' (Action Cost) and 'Hedef Belirleme Maliyeti' (Target Determination Cost). A 'Başlat' (Start) button is at the bottom.

| Player | Hamle Maliyeti | Hedef Belirleme Maliyeti |
|--------|----------------|--------------------------|
| A      | 5              | 5                        |
| B      | 5              | 10                       |
| C      | 5              | 15                       |
| D      | 5              | 20                       |

Resim 3 : Başlangıç ekranı



Resim 4 : Oyun tahtası



Resim 5 : Oyunun bitmiş hahali

```
Output - GoldDigger (run) x | Block.java x |
run:
A oyuncusu hedefini belirledi: 1-2
Aoyuncusu altın aldı: 15
B oyuncusu hedefini belirledi: 1-18
Boyuncusu altın aldı: 15
C oyuncusu hedefini belirledi: 17-1
Coyuncusu altın aldı: 20
D oyuncusu hedefini belirledi: 14-18
A oyuncusu hedefini belirledi: 2-3
Aoyuncusu altın aldı: 20
B oyuncusu hedefini belirledi: 9-18
C oyuncusu hedefini belirledi: 18-1
Coyuncusu altın aldı: 20
Doyuncusu altın aldı: 20
```

Resim 6 : Oyun sonunda oyuncular için yazılan Output çıktısı.

## 5.Sonuç

Bu uygulamayı yaparken zorlandığımız noktalar;

- Projeyi yapmaya ilk başladığımızda Board oluşturamadık, oluşturduğumuz matrise altın atama işlemleri ya da oyuncuları hareket ettirme işlemlerini terminal üzerinde görebiliyorduk. İlk olarak board oluşturma kısmında zorlandık.
- İkinci olarak bizi en çok zorlayan şey board üzerindeki oyuncuları hareket ettirme işlemi oldu. Board üzerindeki her bir matris indexini bir butona göndermiştik ancak butonların nasıl hareket ettirileceğini bilmiyorduk.

Sonuç olarak bu projenin bize kazandırdıkları sırasıyla;

- Java programlama dili ile proje geliştirmek
- JFrame, JButton gibi class yapılarını kullanarak Grafiksel kullanıcı arayüzü tasarlamak
- Farklı arama algoritmalarını öğrenmek
- Bu arama algoritmalarını bir uygulama içerisinde kullanmak
- Dinamik özelliklere sahip bir program geliştirmek
- Nesneye yönelik programlama becerilerini geliştirmek

## 6.Kaynakça

- <https://stackoverflow.com/questions/40746687/drawing-a-game-board-in-java>
- <https://docs.oracle.com/javase/7/docs/api/javax/swing/JButton.html>
- <https://www.geeksforgeeks.org/java-math-random-method-examples/>
- [https://www.tutorialspoint.com/java/number\\_sqrt.htm](https://www.tutorialspoint.com/java/number_sqrt.htm)
- <https://www.javatpoint.com/java-jtextfield>
- <https://stackoverflow.com/questions/11927963/get-input-from-jframe>