****

**KOCAELİ ÜNİVERSİTESİ  
 BİLGİSAYAR MÜHENDİSLİĞİ**

**TIC TAC TOE**

Selin Gizem Özkan

150201167

Bilgisayar Mühendisliği Bölümü

**TIC TAC TOE**

Selin Gizem Özkan

150201167

Bilgisayar Mühendisliği Bölümü

Kocaeli Üniversitesi

ozkanselingizem@gmail.com

**1.1.Problem Tanımı**

Tic tac toe adlı projemizde amaç kullanıcının bilgisayara karşı bu oyunu oynamasıdır. Program ilk başta yeni oyun olarak başlatılıyor. Daha sonra kullanıcı oyundan çıkış yapmak isterse, oyunun son hali oyun.txt dosyasına yazdırılıyor. Program tekrar çalıştırıldığında kullanıcıya yeni bir oyun açmak için 1’i kaydedilen son oyundan devam etmek için 2’yi seçme sansı sunuyor. Programda kullanıcıdan oyunun boyutunu, kullanıcı ismini ve hangi harfle oynamak istediğini alıyoruz. Bu bilgileri kullanıcının seçtiği seçeneklere göre gerekli sınıflardaki metod veya yapılandırıcılara göndererek oyunu başlatıyoruz. Ayrıca bu bilgiler bilgiler.txt dosyasında tutuluyor.

**1.2.Yapılan Araştırmalar**

Projemizin başlangıç aşamasında, öncelikle tic-tac-toe oyunun nasıl oynandığına

baktım. Kazanma, berabere kalma veya

karşı tarafın bizi yendiği durumlarda

nasıl bir oyunla karşılaşıyorum bunları inceledim.

Kendi oyunum için bir algoritma oluşturdum ve oluşturduğum bu algoritmaya göre programı geliştirdim. Burada java programlama dilinde sınıflarda yapıcılar nasıl oluşturuluyor, oluşturulan yapıcılarda parametre olarak değer alınırsa bunlara nasıl erişim sağlanıyor ve nasıl bilgiler güncelleniyor bunların araştırmasını yaptık.

Main nasıl nesne oluşturulur ve oluşturulan nesnelere nasıl değer gönderilir bunları öğrendim. Javada dosya işlemlerinin nasıl gerçekleştiğini araştırdım.

**1.4.Genel Yapı**

Programımız da ana Main, Oyuncu

ve OyunTahtasi olmak üzere üç tane class vardır. Program Main içerisinde kullanıcıya iki seçenek sunar. Kullanıcı yeni bir oyun tahtasıyla oyuna başlayabilir , veya kaldığı son oyun üzerinden devam edebilir. int secim olarak tanımladığım değişkenle kontrolü sağladıktan sonra oyunu seçime göre başlar.

Eğer seçimimiz 1 ise yani kullanıcı yeni bir oyun tahtasıyla oyuna başlamak istiyorsa kullanıcıdan ismini, oyun tahtasının boyutunu ve hangi harfle oynamak istediğini alır. Kullanıcının girdiği harfe göre bilgisayar olarak Oyuncu sınıfından oluşturduğumuz oyuncuya harf atıyoruz.

Ardından oyuncu tahtasını oluşturuluyor. Kullanıcıdan hamle yapması isteniyor. Kullanıcı hamlesini girdikten sonra bilgisayar da random olarak herhangi

bir yere harf atıyor. Burada kullanıcıya çıkmak isteyip istemediği sorulur.

Kullanıcı oyundan çıkar ise oyunun son halini OyuncuTahtasi sınıfının içerisinde oluşturduğumuz dosyaya yaz metoduyla oyun.txt dosyasına yazar. Kullanıcı devam ederse oyun bitene kadar hamle alınır. Oyun tahtasının son durumuna göre oyunda kazanan var ise kazananın ismini,

beraberlik durumu var ise beraberlik ekrana yazdırılıyor.

Eğer kullanıcı seçime 2 girerse oyun

son kaydedilen oyun üzerinden başlatılıyor.

1 ve 2 dışındaki bütün seçimler için program hata mesajı verir ve program sonlanır.

Oyun verilen bu parametrelerle devam ediyor.

Oyuncu class’ında kullanılan metod ve yapıcılar:

1.public Oyuncu() : Parametresiz yapıcı.Kullanıcı değer girmezse insan olma true, oynayacağı harfe X atıyor.

2.public Oyuncu(): Aldığı isme göre oyuncuyu oluşturuyor.

3.public Oyuncu(boolean insan, String isim):insan ise X değil ise O harfi atıyor.

4.public Oyuncu(char tip, boolean insan, String isim) : Aldığı parametlerin

this ile atamasını yapıyor.

5.char karakteriAl() : Alınan

karakteri döndürüyor.

6.boolean oyuncuturunuAl() : insan

olarak döndürüyor.

7.String bilgisayarHamlesiUret(): Random olarak bilgisayar hamlesi üretiyor.

8.String insanOyuncuHamlesiKontrol(): Alınan hamleyi kontrol ediyor.

9.String oyuncununHamlesiniAl():

Boş döndürür (kullanılmamiştir.).

10.public String getIsim(): Private

olarak tanımlanan nesnenin ismine ulaşmamızı sağlar. Oyuncunun ismini döndürüyor.

OyunTahtasının metod ve yapıcıları :

İki tane yapıcısı var:

1.Oyuntahtasi(int boyut):Kullanıcıdan alınan boyuta göre boş bir oyun tahtasi oluşturur.

2.OyunTahtasi(char[][]o\_t,int boyut):Aldığı matris ve boyuta göre oyun tahtası oluşturuyor.

3.char[][]oyunTahtasiniAl(): Parametresiz.

Oyun tahtasını döndürüyor.

4.char[][]oyunTahtasiniAl(char[][]o\_t):

Aldığı diziyi döndürüyor.

5.void oyunTahtasiniYazdir():Tahtayi yazdırıyor.

6.boolean hamleyiyaz(String koordinat,char oyuncu): Koordinatı string olarak alıp integera dönüştürüyor. Oyuncunun hamlesini yazıyor.

7.boolean kazanan(char tip): Parametre olarak oynanan harfi alıyor. Gerekli kontrolleri yapıyor.

8.boolean beraberlikKontrol(): Aldığı matriste tüm elemanlar doluysa ve

kazanan yoksa beraberlik durumu belirliyor.

9.void dosyayayaz(): Oyunu oyun.txt dosyasına yazdırır.

**1.5.Referanslar**

**[1]-**<https://www.youtube.com/watch?v=nSLpoW2zkk4&list=PLh9ECzBB8tJPFTpuHKhYayis0H9pS6_rI&index=6>

**[2]-**[**https://gelecegiyazanlar.turkcell.com.tr/konu/android/egitim/android-101/javada-dosya-islemleri**](https://gelecegiyazanlar.turkcell.com.tr/konu/android/egitim/android-101/javada-dosya-islemleri)

**[3]-**<http://serkansakinmaz.blogspot.com.tr/2013/08/java-snfclass-ve-nesneobject-mantg.html>

**[4]-http://www.ugurkizmaz.com/YazilimMakale-588-Java-Text-Dosyasi-Olusturma---Okuma.aspx**

**5-**<http://www.ahmetsayar.com/lecturenotes/>

**6-http://www.yazilimmutfagi.com/10252/windows/java/java-string-sinifi.aspx**

**7-http://umiitkose.com/2015/08/java-string-sinifi-ve-metodlari/**

**8-http://www.kodcu.net/yazi/programi-sonlandirma**

**9-https://www.google.com.tr/search?client=ubuntu&hs=sdV&q=tic+tac+toe+oyunu&spell=1&sa=X&ved=0ahUKEwilmqvU7IvYAhWB7hoKHezoAEkQvwUIKSgA&biw=1920&bih=965**