

# PROGRAMLAMA LABORATUVARI I

## PROJE 4

**PROJE TESLİM TARİHİ: 15.12.2017**

### NESNEYE YÖNELİK TİC TAC TOE

**Projenin Amacı:** Nesneye yönelik programlama mantığını kullanarak Tic Tac Toe oyununu geliştirmek. **Projede  $n \times n$  ( $n$  değeri 3,5 ya da 7 olabilir) tahta üzerinde klasik tic tac toe oyunu gerçekleştirilecektir.** Oyunda bir kullanıcı bilgisayara karşı oynayacaktır. Oyun konsol uygulaması olarak gerçekleştirilecektir. **Oyunun başlangıcında kullanıcıdan oyun tahtasının boyutu ( $n$  değeri), kullanıcının ismi ve hangi harfi (X ya da O) kullanacağı bilgisi alınacaktır.** Harf bilgisi kullanıcı tarafından girilmek zorunda değildir. Girilmemesi durumunda aşağıda açıklamalarda belirtilen ilgili yapıcı (constructor) çalıştırılacaktır.

Aynı zamanda oyunda yarım kalan bir oyunu kaydetme özelliği de bulunacaktır. Sadece bir oyun kaydedilecektir ve oyun bitmeden çıkış yapılmak istenirse oyuncuya oyunu kaydetmek isteyip istemediği sorulacaktır. Oyun kaydedilmek istenirse oyun tahtasının son durumu ve kullanıcı bilgileri bir metin belgesine kaydedilecektir. Oyun başlangıcında da kullanıcıya yeni bir oyun mu açmak istediği yoksa kayıtlı oyundan mı devam etmek istediği sorulacaktır. Yeni oyun açmak isterse tahta boyutu, kullanıcı ismi ve seçmek istediği harf bilgisi alınacaktır. Aksi durumda bilgiler aşağıda listelenen oyun tahtası ve kullanıcı yapıcılarından uygun olanlar seçilerek aktarılacaktır.

**Önemli Not: Oyun aşağıda belirtilen fonksiyonlar üzerinden nesneye yönelik programlama mantığı kullanılarak gerçekleştirilecektir.**

**Programlama Dili:** Uygulama C++ Java ya da C# dili kullanılarak gerçekleştirilecektir.

#### Projede İstenen Nesneye Yönelik Programlama Dizayını:

- Oyunun oynanacağı oyun tahtasını bir nesne olarak düşünmelisiniz. Bu nesnenin iki boyutlu bir karakter dizisi olarak tahtayı temsil eden bir özelliği ve oyun tahtasının başlangıç değerlerinin atanacağı bir yapıcısı olmalıdır. Yapıcının iki farklı tipte tasarlanması istenmektedir.
  - ✓ Parametresiz olan yapıcı oyun tahtasını boş olarak başlatacaktır.  
*oyunTahtasi()*
  - ✓ Parametrelili olan yapıcı ise almış olduğu iki boyutlu oyun tahtası dizisini oyun tahtası nesnesine atayacak ve oyunu bu tahta üzerinden başlatacaktır.  
*oyunTahtasi(char [][] oynTahtasi)*
- Aynı zamanda oyun tahtası nesnesinin aşağıdaki işlevleri yerine getirecek metotları bulunmalıdır.
  - ✓ Oyun tahtasını döndürecek bir metot  
*char [][] oyunTahtasiniAl()*
  - ✓ Oyun tahtasını konsol ekranında göstermeye yarayacak bir metot  
*void oyunTahtasiniYazdir()*

- ✓ Oyuncu tarafından hamle yapılacak alanın boş olup olmadığını kontrol eden boş ise true döndürüp boş olan alana ekleme işlevini gerçekleştiren, dolu ise false döndüren bir metot.  
**boolean hamleyiYaz(string koordinat, char oyuncu)**
- ✓ Oyunu kazanan oyuncuyu belirleyecek bir metot. Bu metot parametre olarak oyuncu id sini alacak ve yaptığı kontroller sonucunda bool olarak, oyuncu oyunu kazandıysa true aksi durumda false değerini döndürecek.
- boolean kazanan(char oyuncu)**
- ✓ Oyunun berabere kaldıysa true aksi durumda false döndürecek bir metot.  
**boolean beraberlikKontrol()**
- Oyunda tanımlanması gereken ikinci bir nesne yapısı da oyuncu nesnesidir. Bu nesnenin hangi harfi kullandığını belirten (X mi O mu) bir özelliği, oyuncunun bilgisayar mı yoksa insan mı olduğunu belirten ikinci bir özelliği ve kullanıcı adını tutan bir isim özelliği bulunmalıdır.
- Oyuncu nesnesinin üç farklı tipte yapıcısının tanımlanması istenmektedir.
  - ✓ Parametresiz olan ilk yapıcı insan olan oyuncuya X harfini ve insan olma özelliğine true değerini atar.  
**oyuncu()**
  - ✓ Tek parametrelili olan yapıcı bool olarak oyuncunun insan mı yoksa bilgisayar mı olduğunu belirten bir parametre alacaktır. Almış olduğu bu değeri İnsan olma özelliğine atayacaktır. Eğer bu bool değeri true ise harfe X değerini aksi durumda harfe O değerini atayacaktır.  
**oyuncu(boolean insanmiKontrolu)**
  - ✓ İki parametrelili olan yapıcı, hem bool bir değer olan insan olma özelliğini hem de harf değerini girilen parametrelere göre belirleyecektir.  
**oyuncu(boolean insanmiKontrolu, char kr)**
- Oyuncu nesnesinin sahip olacağı beş temel metot bulunmaktadır.
  - ✓ Oyuncunun hangi harf ile oynadığını döndüren bir metot,  
**char karakteriAl()**
  - ✓ oyuncunun insan mı yoksa bilgisayar mı olduğunu döndüren metot,  
**boolean oyuncuTurunuAl()**
  - ✓ oyuncunun hamlesinin hangi hücreye olduğunu döndüren bir metot  
**string oyuncununHamlesiniAl()**
  - ✓ insan olan oyuncuyu hamle yapması için uyarın ve klavyeden kullanıcı tarafından girilen hamleyi alıp, döndüren bir metot  
**string insanOyuncuHamlesiniKontrol()**
  - ✓ random olarak bilgisayar için oyun tahtası üzerinde bir hamle koordinatı döndüren bir metot.  
**string bilgisayarHamlesiUret()**

Yukarıda oyunu geliştirmek için ihtiyaç duyacağınız en temel metotlar verilmiştir. İhtiyaç durumunda sınıflara farklı özellikler ve metotlar ekleyebilirsiniz ya da metot parametrelerinde değişiklikler yapabilirsiniz.

Aşağıda örnek oyun tahtaları gösterilmiştir.

	A	B	C
1			
2			
3			

	A	B	C	D	E
1					
2					
3					
4					
5					

	A	B	C	D	E	F	G
1							
2							
3							
4							
5							
6							
7							

**Demo günü teslim edilecekler:**

-Programınızın CD'si. CD içinde kodunuz, programınızın çalıştırabilmesi için gerekli olan instructionlar (readme.txt) ve raporunuz (aşağıda belirtilen yönergelerle göre hazırlanacaktır.) olmalıdır.

-Proje'nin rapor çıktısı (hard-copy).

**Proje Değerlendirme Yönergesi Açıklamalar**

**1. PROJE RAPORU:** Her proje sonunda teslim edilmesi gereken projenizi her yönüyle açıklayacağınız bir dokümandır.

**1.1. Problem Tanımı:** Proje kapsamında sizden çözüm bulmanız beklenen problem ile ilgili açıklama yapılması gerekmektedir. Burada amaç projenin ne kadar anlaşıldığını test etmektir.

**1.2. Yapılan Araştırmalar:** Proje geliştirilmesi aşamasında karşılaşılan sorunlara nasıl çözümler bulunduğu ve bu konularda yapılan araştırmalar açıklanmalıdır.

**1.3. Tasarım**

**1.3.1. Akış şeması:** Proje içerisinde yer alan algoritma ve işlemleri şekilsel olarak ifade edecek şema oluşturmanız beklenmektedir.

**1.3.2. Yazılım mimarisi:** Projenin kodlanması aşamasında kullanılacak kod yapısı ve geliştirme aşamalarını gösteren bir yapı hazırlanması beklenmektedir.

**1.4. Genel Yapı:** Projenizi genel yapısı bakımından her yönüyle özetlemeniz gerekmektedir.

**1.5. Referanslar:** Proje geliştirilirken ve araştırma aşamasında faydalanan kaynaklar rapor dokümanının en altında listelenmeli ve doküman içerisinde de ilgili yerlerde indekslenmelidir.

**Referans formatı aşağıda verilen örneklerle uygun olmalıdır.**

**Kitap, çok yazarlı**

Larson, G. W., Ellis, D. C., & Rivers, P. C. (1984). Essentials of chemical dependency counseling. New York: Columbia University Press.

**Report from a private organization (author & publisher same)**

National League for Nursing. (1990). Self-study report for community health organizations (Pub. No. 21-2329). New York: Author.

**Unpublished master's thesis**

Paulosky, K. A. (1997). Knowledge and attitudes of pain and activities of nurse administrators. Unpublished master's thesis, Northern Michigan University, Marquette, Michigan.

**Article in a journal (continuous pagination throughout volume)**

Burke, R. J., Shearer, D., & Deszca, E. (1984). Correlates of burnout phases among police officers. Group and Organizational Studies, 9, 451-466.

### **Article in a Popular Magazine**

Caloyianis, N. (1998, September). Greenland sharks. National Geographic, 194, 60-71.

### **Web Site**

[http://en.wikipedia.org/wiki/Neural\\_network](http://en.wikipedia.org/wiki/Neural_network) (Access date: 07.10.2013)

**İNTİHAL: İNTERNETTEN ALINAN KOD PARÇACIKLARI KOPYA OLARAK DEĞERLENDİRİLECEKTİR. KOPYA ÇEKTiĞİ YA DA KOPYA VERDİĞİ TESPİT EDİLEN ÖĞRENCİLER SUNUMA ALINMAYACAKTIR.**

**PROJELER İKİŞER KİŞİLİK GRUP OLUŞTURULARAK YAPILACAKTIR!**