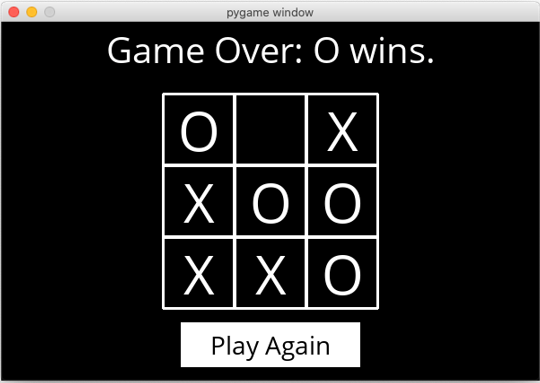
**Tictactoe Oyunu Ödevi**

İndireceğiniz *tictactoe.zip* dosyasında dosya bulunmaktadır; runner.py ve tictactoe.py.

* *tictactoe.py* oyunu oynamak ve en uygun hamleleri yapmak için gereken tüm kod akışını içermektedir. Dosyanın içindeki kodlara bakıldığında;
  + Oyunun olası hareketlerini temsil etmek için üç değişken tanımlandığı görülmektedir: *X, O* ve *EMPTY*.
  + initial\_state fonksiyonu, başlangıç durumunu döndürmektedir.
* *runner.py* oyunun grafik arayüzünü çalıştıracak tüm kodları içermektedir.

Tictactoe oyununu MINIMAX algoritmasını kullanarak tasarlamanız gerekmektedir. Geliştirme sonucunda *tictactoe.py* dosyasında aşağıda verilen maddelerde bulunan tüm komutları/boşlukları tamamladıktan sonra, yapay zekaya karşı oynamak için runner.py dosyasını çalıştırabilirsiniz. Aşağıdaki oyunun ekran görüntüsüne BOARD ismi verilmiştir ve açıklamalardan “board” geçen kısımlarda O ve X ekranına referans verilmektedir.



***player, actions, result, winner, terminal, utility*,** ve ***minimax*** fonksiyonlarını **tamamlayın**.

* **Player** fonksiyonu girdi olarak *board*un durumunu almalı ve sıranın hangi oyuncuda olduğunu (X veya O) döndürmelidir.
  + Oyunun başlangıç durumunda ilk hamleyi X alır. Daha sonra oyuncu her ilave hamlede değişiklik yapar.
  + Giriş olarak bir terminal boardu verilmişse (yani oyun zaten bittiyse) herhangi bir dönüş değeri kabul edilebilir.
* **Actions** fonksiyonu, belirli bir *board*da gerçekleştirilebilecek tüm olası eylemlerin bir kümesini döndürmelidir.
  + Her eylem bir tuple (i, j) olarak temsil edilmelidir; burada i, hareketin satırına (0, 1 veya 2) karşılık gelir ve j, satırdaki hangi hücrenin harekete karşılık geldiğine karşılık gelir (ayrıca 0, 1 veya 2) 2).
  + Olası hamleler, tahtada halihazırda X veya O bulunmayan hücrelerdir.
  + Giriş olarak bir terminal ekranı sağlanmışsa herhangi bir dönüş değeri kabul edilebilir.
* **Result** fonksiyonu, giriş olarak bir board ve bir action alır ve orijinal boardu değiştirmeden yeni bir board statei döndürmelidir.
  + Yapılacak Action geçerli bir eylem değilse programınızın bir hata/exception ‘fırlatması’ gerekir.
  + Döndürülen board durumu, orijinal giriş boardunun alınmasından ve sırası gelen oyuncunun giriş actionı ile belirtilen hücrede hamle yapmasına izin verilmesinden kaynaklanan board olmalıdır.
  + Daha da önemlisi: Minimax sonuçta hesaplama sırasında birçok farklı board durumunun dikkate alınmasını gerektireceğinden orijinal board değiştirilmeden bırakılmalıdır. Bu, boarddaki bir hücrenin basitçe güncellenmesinin *Result* fonksiyonunun doğru bir uygulaması olmadığı anlamına gelir. Herhangi bir değişiklik yapmadan önce muhtemelen boardun bir kopyasını almak faydalı olacaktır.
* **Winner** fonksiyonu bir boardu girdi olarak kabul etmeli ve varsa boardun kazananını döndürmelidir.
  + Eğer X oyuncusu oyunu kazandıysa fonksiyonunuz X değerini döndürmelidir. Eğer O oyuncusu oyunu kazandıysa fonksiyonunuz O değerini döndürmelidir.
  + Yatay, dikey veya çapraz olarak art arda üç hamle yaparak oyunu kazanabilirsiniz.
  + En fazla bir kazanan olacağını varsayabilirsiniz
  + Oyunun kazananı yoksa (oyun devam ettiği için veya berabere bittiği için), fonksiyon *None* değerini döndürmelidir.
* **Terminal** fonksiyonu bir boardu giriş olarak kabul etmeli ve oyunun bitip bitmediğini gösteren bir *boolean* değeri döndürmelidir.
  + Birisi oyunu kazandığı için veya hiç kimse kazanmadan tüm hücreler dolduğu için oyun bittiyse, fonksiyon *True* değerini döndürmelidir.
  + Aksi takdirde, oyun hala devam ediyorsa fonksiyon *False* değerini döndürmelidir.
* **Utility** fonksiyonu, bir terminal boardunu giriş olarak kabul etmeli ve board Utilityi çıkış olarak kabul etmelidir.
  + X oyunu kazandıysa değer 1'dir. Eğer oyunu O kazandıysa değer -1'dir. Eğer oyun berabere biterse değer 0'dır.
* **Minimax** fonksiyonu girdi olarak bir boardu almalı ve oyuncunun o boarddan hareket etmesi için en uygun hareketi döndürmelidir.
  + Geri dönen hamle, boardda izin verilen eylemlerden biri olan en uygun action (i, j) olmalıdır. Birden fazla hamle eşit derecede optimalse, bu hamlelerden herhangi biri kabul edilebilir.
  + Board bir terminal boardsu/ekranı ise *minimax* fonksiyonu *None* değerini döndürmelidir.
* Bir board giriş olarak kabul eden tüm fonksiyonlar için, bunun geçerli bir board olduğunu (yani, her biri X, O veya EMPTY olmak üzere üç değere sahip üç satır içeren bir liste olduğunu) varsayabilirsiniz. Fonksiyonların parametrelerinin sırasını değiştirmemelisiniz.
* Tüm fonksiyonlar doğru şekilde uygulandıktan sonra python runner.py'yi çalıştırıp yapay zekanıza karşı oynayabilirsiniz ☺