

Dersin Adı	: Algoritma ve Programlama - I
Proje - 2	: Kilit Oyunu Simülasyonu
Projenin Verilme/Duyurulma Tarihi	: 27.12.2022 Salı, Saat 22:59
Grupların Oluşturulması İçin Son Tarih	: 30.12.2022 Cuma, Saat 23:59
Kaynak Kod ve Rapor Teslim Tarihi	: 06.01.2023 Cuma, Saat 23:59
Geç Kaynak Kod ve Rapor Teslim Tarihi	: 08.01.2023 Pazar, Saat 23:59

### GENEL BİLGİLER

Kilit oyunu 2 kişi ile oynanan bir strateji oyunudur. Oyun alanı, 8X8 boyutlarındadır ve 64 kareden oluşmaktadır. Oyunda 2 oyuncunun da 8'er taşı bulunmaktadır. Oyunun amacı, rakibin bütün taşlarını dışarı çıkarmaktır. taş sayısını 2'nin altına indirmektir.

Oyun başlangıcında oyuncular taşlarını kendilerine en yakın satırdaki karelere yerleştirirler. Daha sonra oyuncular sırayla kendilerine ait bir taşı hareket ettirerek rakibin taşlarını kilitlemeye çalışırlar. Taşlar oyun alanı içinde yatay ya da dikey istenildiği kadar uzağa hareket ettirilebilir ancak taşların üzerinden atlanamaz.

Yataydaki 2 komşu karesinde rakip taş yer alan bir taş, dikeydeki 2 komşu karesinde rakip taş yer alan bir taş veya yataydaki ve dikeydeki komşu karelerinde rakip taş yer alan köşedeki bir taş kilitlenmiş sayılır. Sırası gelen oyuncu, yaptığı hamle sonucunda rakip oyuncunun bir veya daha fazla taşını kilitlerse o taşı/taşları dışarı çıkarır. Bir başka deyişle bir oyuncu, hamle sırası kendisindeyken hareket ettirdiği kendi taşı ile kilitlediği rakip oyuncuya ait taşı/taşları dışarı çıkarır. Rakibin tüm taşlarını dışarı çıkarmayı taş sayısını 2'nin altına indirmeyi başaran oyuncu oyunu kazanır.

Örnek bir oyun için bakınız: [https://www.youtube.com/watch?v=GxulU4jEP\\_s](https://www.youtube.com/watch?v=GxulU4jEP_s)

### PROBLEM TANIMI

Yukarıda tarif edilen Kilit oyununun simüle edilmesini sağlayacak bir program geliştirilmesi istenmektedir. Oyun alanı en az 4X4, en çok 8X8 boyutlarında olmalıdır. Oyun alanının satırları sayma sayıları, sütunları İngilizce'deki büyük harfler ile temsil edilmelidir.

Program çalıştığında ilk olarak, oyuncuları temsil etmek için kullanıcıdan iki karakter alınmalıdır:

1. oyuncuyu temsil etmek için bir karakter giriniz: X
2. oyuncuyu temsil etmek için bir karakter giriniz: Y

Oyun, tekrar tekrar oynanabilmelidir ve her oyunun başlangıcında, oyun alanının satır/sütun sayısı (aynı zamanda, her oyuncunun sahip olacağı taş sayısı) [4-8] kullanıcıdan alınarak başlangıç oyun alanı görüntülenmelidir. Örneğin 4X4 boyutlarındaki bir başlangıç oyun alanı aşağıdakine benzer şekilde görüntülenebilir:

Oyun alanının satır/sütun sayısını giriniz(4-8): 4

	A	B	C	D	
1	Y	Y	Y	Y	1
2					2
3					3
4	X	X	X	X	4
	A	B	C	D	

Oyuncu X, lütfen hareket ettirmek istediğiniz kendi taşınızın konumunu ve hedef konumu giriniz:

Oyuncular hareket ettirmek istediği kendi taşının şimdiki konumunu ve hedef konumu, ilgili konumun satır numarası (örnekte [1-4]) ile sütun harfi (örnekte [A-D]) bitişik olacak şekilde (örneğin 3C) ve bu iki konum arasında bir boşluk bırakarak (örneğin 3C 1C) girmelidir.

Her hamleden sonra oyun alanı tekrar görüntülenmelidir ve bu hamle ile kilitlenip dışarı çıkarılan bir taş olduysa bu taşın konumu belirtilmelidir. Örneğin yukarıdaki örnekteki oyun alanı, oyunun herhangi bir aşamasında aşağıdakine benzer şekilde görüntülenebilir:

	A	B	C	D	
1	Y		X		1
2	X				2
3			X		3
4					4
	A	B	C	D	

2C konumundaki taş kilitlendi ve dışarı çıkarıldı.

Oyuncu Y, lütfen hareket ettirmek istediğiniz kendi taşınızın konumunu ve hedef konumu giriniz:

Oyun sonunda, hangi oyuncunun oyunu kazandığı ekrana yazdırılmalıdır ve kullanıcıya tekrar oynamak isteyip istemediği sorulmalıdır:

Oyuncu X oyunu kazandı.  
Tekrar oynamak ister misiniz(E/H)?:

#### **Notlar:**

1. Veri girişleri sırasında, kullanıcının hatalı veya oyun kurallarına göre geçersiz veri girişleri, çalışma zamanı hatasına neden olabilecek veri girişleri (istisna işleme (exception handling)) kontrol edilmeli ve uygun veri girişi yapıncaya kadar beklenilmesi sağlanmalıdır. Ayrıca veriler mutlaka belirtilen sıra ve şekilde alınmalıdır.
2. Bir oyuncunun oyun kurallarına uygun bir hamle yapma şansı olmaması durumunun ele alınmasına gerek yoktur. Bir başka deyişle, sırası gelen oyuncunun mutlaka geçerli bir hamle yapma şansı olacağını varsayabilirsiniz.
3. Bu projenin temel amacı, şimdiye kadar işlenen konular dahilinde tek/iki boyutlu liste, sözlük ve string veri yapılarının kullanımını pekiştirmektir. Ayrıca programın modüler yapıda olması (fonksiyonlardan oluşması) ve global değişken (bir fonksiyonun içinde, o fonksiyonda tanımlanmamış/yaratılmamış bir değişken) kullanılmaması beklenmektedir.
4. Python dilinin standart kütüphanesindeki tüm modüller (The Python Standard Library: <https://docs.python.org/3/library/>) kullanılabilir, ancak üçüncü-parti başka kütüphaneler kullanılmamalıdır.
5. Proje, iki kişilik gruplar halinde yapılacaktır, isteyenler tek kişi de yapabilir. Ancak dersi tekrar alan ve haftalık uygulama ödevlerinden sorumlu olmayanlar (uygulama derslerine girmeyenler), projeyi tek kişi yapmak zorundadır.
6. Projeyi iki kişilik grup olarak yapacak olanların oluşturdukları grup bilgisini, 30.12.2022 Cuma, Saat 23:59'a kadar <https://forms.gle/X2t4RZ6n9FZhMDez9> adresindeki forma girmeleri gerekmektedir.
7. Herhangi bir sorunuz olursa, <https://egeders.ege.edu.tr/> web sitesindeki ilgili ders sayfasında bu ödev ile ilgili açılan forumu kullanınız. Ayrıca olası güncelleme ve/veya açıklamalar için forumu takip ediniz.

8. Projeyi vakit kaybetmeden hemen yapmaya başlayınız, kodlamaya başlamadan önce algoritmanızı tasarlamanız (pseudocode yazmanız) önerilir.
9. Projeyi kendiniz yapınız, arkadaşlarınızla kod paylaşımı yapmaktan özellikle kaçınınız.
10. Projeyi teslim etmeden önce farklı durumları test etmeyi sağlayacak örnek girdi ve çıktılar oluşturarak programınızı test ediniz. (Girdileri tek tek girmek yerine, konsola toplu olarak yapıştirarak zaman kazanabilirsiniz.)

#### **Raporda Bulunması Gerekenler:**

- Kapak ve içindekiler sayfaları
- Programcı Kataloğu:
  - yarattığınız veri yapısı (liste, sözlük v.b.) nesnelerini listeleyiniz ve kullanım amaçlarınızı açıklayınız.
  - oluşturduğunuz fonksiyonların başlıklarını/arayüzlerini listeleyiniz ve işlevlerini açıklayınız.
- Test Kataloğu: Programın olası farklı kullanım durumları (test senaryoları) için girdileri, beklenen çıktılar, programınızın ürettiği çıktılar ve başarı durumlarını belirtiniz.
- Rapor bilgisayar ortamında hazırlanmalı ve biçimlendirmeye özen gösterilmelidir: sayfa düzeni (kenar boşlukları, sayfa no v.b.), yazı biçimi (başlıklar, paragraflar v.b.)

#### **DİKKAT EDİLMESİ GEREKEN NOKTALAR:**

##### **Projenin Teslim Edilmesi:**

- Kaynak kod dosyası (.py uzantılı), dosya adı grup üyelerinin öğrenci numaralarının alt çizgi karakteri ile birleşiminden (örneğin 05090004219\_05090004235.py) oluşacak şekilde, grup üyelerinden birisi tarafından <https://egeders.ege.edu.tr/> web sitesindeki ilgili ders sayfası kullanılarak sisteme yüklenmelidir.
- Rapor dosyası (.doc ya da .pdf uzantılı), dosya adı grup üyelerinin öğrenci numaralarının alt çizgi karakteri ile birleşiminden (örneğin 05090004219\_05090004235.py) oluşacak şekilde, grup üyelerinden birisi tarafından <https://egeders.ege.edu.tr/> web sitesindeki ilgili ders sayfası kullanılarak sisteme yüklenmelidir.
- Kaynak kod ve rapor dosyaları sisteme tekrar tekrar yüklenebilir, ancak sistemde sadece en son yüklenen dosyanın saklandığı unutulmamalıdır. Ayrıca yükleme işlemi tamamlandıktan sonra dosyaları kontrol ediniz ve dosyaların sorunsuz bir şekilde sisteme yüklendiğinden emin olunuz.
- Kaynak kod ve rapor tesliminde en fazla 2 günlük gecikmeler kabul edilecek, ancak son teslim tarihinden sonraki her gün için kaynak kod ve/veya rapor notundan %20 kesinti yapılacaktır.

##### **Projenin Değerlendirilmesi:**

1. Projenin değerlendirmesinde; programın doğru ve eksiksiz çalışmasının yanında, yapısal programlama ilkelerine uygunluk (sabit (constant) kullanma, anlamlı değişken/sabit isimleri, gerekli görülen yerlerde açıklamalar (comments) v.b.), etkinlik (gereksiz işlemlerden ve gereksiz kod tekrarından kaçınma v.b.) ve modüler programlama ilkelerine uygunluk (programın fonksiyonlardan oluşması ve global değişken kullanılmaması v.b.) ta dikkate alınacaktır. Buna göre puanlama şu şekildedir:
  - Yapısal programlama ilkelerine uygunluk: 5 puan
  - Etkinlik: 5 puan
  - Modüler programlama ilkelerine uygunluk: 10 puan
  - Programın doğru ve eksiksiz çalışması: 70 puan
  - Rapor: 10 puan
2. Kaynak kodları arasında belirli bir oranın üstünde benzerlik tespit edilen projelerin notunda aynı oranda kesinti yapılacaktır veya bu projeler sıfır alacaktır.