

2012-2013 Güz Yarıyılı Bilgisayar Bilimlerine Giriş 1 Dersi Projesi

Bu ödevde *Game of Life* simülasyonu yazılacaktır. Game of Life 1970 yılında John Conway tarafından geliştirilmiştir. 2-D bir dünyada yaşayan organizmaların belirli kurallara göre bir sonraki jenerasyonda doğum, hayata devam veya ölüm durumları belirlenerek yaşam döngüsü simüle edilir.

Konu ile ilgili detaylı bilgiyi ve örnekleri aşağıdaki linklerden inceleyebilirsiniz:

http://en.wikipedia.org/wiki/Conway%27s_Game_of_Life.

<http://www.math.com/students/wonders/life/life.html>

2-D dünya, bir matris ile gösterilir. Matrisin her hücresinde bir organizma yaşayabilir. İlk jenerasyonda bazı hücrelerde yaşayan organizmalar vardır. Bunların dışında kalan diğer hücreler boştur.

Her hücrenin sağ, sol, üst, alt ve diagonallerinde etrafını çevreleyen 8 komşusu bulunur. Matrisin kenarlarında yaşayan hücrelerin komşu sayısının 8'den az olacağına (örneğin [1,1] adresindeki hücrenin sağında, altında, sağ alt diagonalinde olmak üzere toplam 3 komşusu vardır) dikkat ediniz.

Bir sonraki jenerasyonda organizmaların durumları aşağıdaki kurallara göre belirlenir:

1. **Ölüm** : Eğer bir organizmanın *sıfır* veya *bir* komşusu varsa yalnızlıktan, *üçten fazla* komşusu varsa kalabalıktan dolayı ölür.
2. **Doğum** : Eğer boş bir hücrenin *üç* tane yaşayan komşusu varsa, boş hücrede yeni bir organizma doğar.
3. **Yaşamaya devam** : Canlı bir organizmanın *iki* veya *üç* adet canlı komşusu varsa, organizma yaşamaya devam eder.

Her doğum veya ölüm, komşu hücrelerin o anki durumunu değiştirmez, bir sonraki jenerasyonu etkiler. Kurallar bütün organizmalara *aynı anda* uygulanır. Yani her organizmanın durumu, bir önceki jenerasyona ait matris değerlendirilerek belirlenir. Bundan dolayı, örneğin üç canlı komşusu olan canlı bir organizmanın dördüncü bir komşusu doğarsa bu organizma ancak bir sonraki jenerasyonda ölür.

Game of Life aşağıdaki koşullardan biri sağlanmışsa sona erer:

1. Bütün organizmalar ölmüşse,
2. Ardışık iki jenerasyonda organizmaların hepsinin durumu aynı kaldıysa,
3. Bazen HAYAT döngüsü osilasyona girebilir. Yani bir jenerasyon n-jenerasyon ileri-geri aynı şekilde tekrarlanır. Eğer bir jenerasyon *iki önceki* jenerasyonun tamamen aynısı olur ve yaşam döngüsü ileri-geri doğru aynı şekilde tekrarlanırsa,
4. Kullanıcı tarafından giriş bilgisi olarak verilen sayıda jenerasyon tamamlandı ise

Giriş Bilgileri :

1. HAYAT matrisinin satır ve sütun sayısını ve oyunun bitmesi için yukarıda verilen 4. şart olan maksimum jenerasyon sayısını kullanıcıdan alınız.
2. HAYAT matrisini okuyarak ilk jenerasyonda hangi adreslerde yaşayan organizmalar olduğu bilgisini kullanıcıdan alınız.

3. Her jenerasyon sonunda HAYAT matrisinin ekrana yazdırılıp yazdırılmayacağını kullanıcıya sorunuz.

Çıkış Bilgileri :

1. Matrisi ekrana yazdırırken yaşayan hücreleri **x**, ölü hücreleri **o**, boş hücreleri – ile gösteriniz.
2. Her jenerasyon sonunda durumun ekranda gösterilmesi istendiyse, her jenerasyon sonundaki durumu ekrana yazdırınız.
3. Her jenerasyon sonunda durumun ekranda gösterilmesi istenmediyse, sadece ilk 3 jenerasyonu ve son 3 jenerasyonu gösteren matrisleri ekrana yazdırınız.
4. Oyunun kaç jenerasyon sürdüğünü ve oyunun bitme sebebini ekrana yazdırınız.

Teslim Edilecek Rapor:

Teslim edeceğiniz raporunuz aşağıdaki şekilde hazırlanmalıdır.

Rapor 2 bölümden oluşmalıdır :

1. **Program :** PASCAL dilinde program yazınız. Kullandığınız değişkenlere kullanım amaçları ile uygun anlamlı isimler veriniz. Gerekli yerlere (örneğin ornaizmanın doğumunun kontrolü, yeni jenerasyona geçiş durumunda yapılan işlemler vb.) açıklama satırları koyarak yapılan işlemleri açıklayınız.
2. **Uygulama :** Yukarıda verilen 4 durma şartını sağlayan 4 ayrı başlangıç matrisi belirleyiniz. Bu başlangıçlar için son 5 jenerasyonun ekran çıktısını veriniz. Oyunun bitiş sebebini ve toplam jenerasyon sayısını yazdırınız.

Teslim Tarihi:

Raporlarınızı **final haftası** yapılacak laboratuvar da teslim ediniz ve bilgisayar başında çalışmasını gösteriniz. Teslim tarihi, laboratuvardan sorumlu asistanların web sayfasında ilan edilecektir.

Puanlama :

Algoritma Tasarımı ve Programın Çalışması: (%80)

1. Ödev, istenilen işlerin tamamını yerine getirmelidir.
2. Gereksiz kontrollerden ve işlemlerden arınmış bir tasarım yapılmalıdır.
3. Program hatasız çalışmalıdır.
4. Programın çalışması sırasında, konuyu bilmeyen kişilerin rahatlıkla anlayabilmesi için, giriş ve çıkışlarda mesajlarla bilgi verilmelidir.

Rapor Dokümantasyonu: (%20)

1. Raporun kapak sayfasında, dersin adı, öğrencinin ad, soyad ve numarası, ödev konusu bilgileri yer almalıdır.
2. Rapor anlaşılır yazılmalı, gerekli bütün bilgileri içermeli ama gereksiz detaylarla uzatılmamalıdır.
3. Ödev A4 kağıda, yazıcıdan çıkış alınarak verilmelidir.
4. Gereksiz kod tekrarı olmamalıdır.
5. Kaynak kodun formatı düzgün olmalıdır.