EGE UNIVERSITY
FACULTY of ENGINEERING
COMPUTER ENGINEERING DEPARTMENT
204 DATA STRUCTURES (3+1)
2019-2020 FALL SEMESTER

PROJECT 1 : (Arrays, Matrices, Methods, Random Numbers)

1.1 BIRTHDAY PARADOX EXPERIMENTS
1.2 CLASSIFICATION USING K-NEAREST NEIGHBORS (KNN) ALGORITHM

Date Given : **Control Date:** (will be announced by the session assistants)

1.1) Birthday Paradox experiments

The birthday paradox says that the probability that two people in a room will have the same birthday is more than half, provided n, the number of people in the room, is more than 23. This property is not really a paradox, but many people find it surprising.

(from Goodrich and Tamassia)
Example Applet : http://www-stat.stanford.edu/~susan/surprise/Birthday.html
Demo: http://demonstrations.wolfram.com/SimulatingTheBirthdayProblem/

*Bu problemin haftanın aynı günü doğanlar için bir sürümünü düşününüz. Örneğin; Biri 02.03.1994 (Çarşamba), diğeri 09.06.1993 (Çarşamba) gününde doğan iki kişi olsun. Problemin bu versiyonunda bu iki kişinin haftanın aynı gününde doğmuş olması dikkate alınacaktır (farklı doğum tarihlerine sahip olmalarına rağmen).

15 points

Write a C#/Java/Python program that can test this paradox by a series of experiments on randomly generated birthdays, which test this paradox for **n = 2, 3, 5, 10, 20 people in room.**

*n'in her bir değeri için deneyleri **15'er** kere tekrarlayıp ortalama çakışma-eşleşme sayısını da tablo halinde yazdırınız. Örnek olarak 10 kişilik bir ortamda (ortalama) kaç doğum günü çakışması vardır? İlgili deneyleri yapan metodu yazarak hesaplatınız.*

- **Çakışma:** 2 kişinin doğum tarihi **haftanın aynı gününe** geliyorsa 1 çakışma vardır. Bunun için doğulan **gün adının** aynı olması yeterlidir. **Doğum yılının, ayının ve gün numarasının eşit olmasına gerek yoktur.** 3 kişi aynı günde doğdularsa 2 çakışma vardır. 7 kişinin aynı doğum gününe sahip olmaları durumunda 6 çakışma vardır.

- Programınız aşağıdaki **iki listelemeyi de içermelidir**:
  i) Simülasyonlar yapılarak **15 deney** için sadece istatistiksel değerler (çakışma sayıları) 16 x 5'lik ayrı bir tablo olarak ekrana verilebilir. **n'in** farklı değerleri (n = 2, 3, 5, 10, 20 kişi) için 5 sütun, her deneydeki ortalama çakışma sayıları için 15 satır ve bu 15 değerin ortalaması için 16. satır yazdırılır.

  ii) Tek tek denemelerdeki çakışmalar gösterilmelidir. Her deneme için 7 elemanlı bir dizide gösterilen haftanın günlerindeki çakışma sayıları yazdırılmalıdır. n'in her bir değeri için de bu işlem 15 kere tekrarlanır. Ardından i)'deki 16x5'lik özet liste de verilmelidir.

- 15 deneme sıfırdan yapılacaktır. Yani örnek olarak 10 kişi için 150 doğum tarihinin çakışmasına bakılmamalıdır. İstatistikten bildiğiniz şekilde her 10 kişi için çakışmalar

sıfırlanıp yeni bir deney yapılacaktır. 15 deneyin ortalaması daha gerçekçi bir değer verecektir.

- Programlarınız **konsol uygulaması** olarak yapılabilir. Dileyenler Form Uygulaması şeklinde de tasarlayabilirler.

RAPOR (10 puan) FORMATI – Bölüm 1

1.1.1. Gerçekleştirilen Platform ve Dil (C#/Java/Python/Diğer) ve Sürüm Adı
1.1.2. Program 1.1'in kısa tanımı (projeden)
1.1.3. Program 1.1 için kullanılan bileşik veri tiplerinin ve metotların kısa açıklamaları
1.1.4. Elde edilen örnek sonuçlar (Ekran görüntüleri) (1.1.'de elde edilen 16x5'lik tablo, n=2, 3, 5, 10 ve 20 için 15 denemenin sonuçları yani ekran görüntüleri)
1.1.5. Proje 1.1 Yazılım Geliştirme İçin Harcanan Süreler (kişi ve saat bazında)

PROJE TESLİMİNE İLİŞKİN BİLGİLER

1) Data Structures dersinin Projeleri için (ortak çalışma imkanınız bulunan kişilerle) **2 veya 3 kişilik çalışma grupları** oluşturmanız önerilir. Dileyen öğrenciler asıl A projelerini tek kişi yaparak da teslim edebilirler. *Alternatif B projeleri Proje 2'den itibaren verilecektir, daha basit projeler olup tek kişiliktir, grup olarak teslim edilemez.*

2) **Raporun sonuna, 4. sayfadaki Özdeğerlendirme raporunu doldurarak ekleyiniz. Rapor ve her iki programın açıklama satırları destekli kaynak kodları (.cs uzantılı),** çalışma grubundan bir öğrenci tarafından (**dersin duyurularında belirtilen formata uygun olarak**) son teslim tarihine kadar ilgili seçenekten **sisteme yüklenmelidir.**

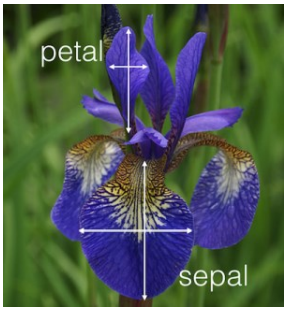
3) **MOSS üzerinden belli ölçüde kod benzerlikleri görülen gruplara Proje notu olarak 0 atanacaktır.**

1.2 CLASSIFICATION USING K-NEAREST NEIGHBORS (KNN) ALGORITHM
K EN YAKIN KOMŞU YÖNTEMİ İLE SINIFLANDIRMA

Doğada gördüğümüz bir çiçeğin, Zambak (veya süsen) bitkisi olduğu biliniyor. Üç farklı türünden hangisine ait olduğunu bulduran bir algoritmanın yazılması istenmektedir. Elimizde her bir çiçek türünden 150 örnek üzerinden ölçülerek alınan veriler bulunmaktadır. Her bir örnek için 4'er adet özellik (çanak yaprak uzunluğu, çanak yaprak genişliği, taç yaprak uzunluğu, taç yaprak genişliği) ve hangi sınıfta (tür) olduğu bilgisi hazır olarak verilmektedir. Tablo 1'de 6 tanesine yer verilmiştir. Veriseti: https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data

Tablo 1: Çiçek verisetinden alınmış 6 adet bitki örneğine ilişkin bilgiler

Örnek No	Çanak Yaprak Uzunluğu	Çanak Yaprak Genişliği	Taç Yaprak Uzunluğu	Taç Yaprak Genişliği	Tür
0	5.1	3.5	1.4	0.2	*I. setosa*
1	4.9	3.0	1.4	0.2	*I. setosa*
2	7.0	3.2	4.7	1.4	*I. versicolor*
3	6.4	3.2	4.5	1.5	*I. versicolor*
4	6.3	3.3	6.0	2.5	*I. virginica*
5	5.8	2.7	5.1	1.9	*I. virginica*
…	…	…	…	…	…



Şekil 1: Zambak çiçeğinde çanak (sepal) ve taç (petal) yapraklar.

10 puan a) Bulduğumuz ancak türünü bilmediğimiz bir **çiçeğin hangi türe ait olduğunu tespit eden algoritmayı (k en yakın komşu yöntemi) yazınız** (hazır kNN kullanmayınız). k değerini ve yeni çiçeğin **belirlenen** özellik(ler)ini girdi olarak aldırarak bu yöntemle hangi sınıftan (*I. Setosa, I. Versicolor, I. Virginica)* olduğunu bulduran kNN algoritmasını kendiniz yazınız. Yöntem:

Elinizdeki türü bilinmeyen çiçeğin özelliklerini, verisetindeki tüm kayıtlarla karşılaştırarak özellikleri uzaklık d (distance) formülüne göre en yakın olan k tane çiçeği bulmalısınız. Bulduğunuz bu k tane çiçeğin türlerine bakarak en çok sayıda hangi türden çiçek varsa çiçeğinizi o türden sayacak ve sınıflandıracaksınız.

$A = (x_1, x_2, \ldots, x_m)$ ve $B = (y_1, y_2, \ldots, y_m)$ özellik vektörleri, m özellik sayısı olmak üzere iki çiçek (A ve B) arasındaki uzaklığı (distance) hesaplayan

d (A,B) Formülü: $\sqrt{\sum_{i=1}^{m}(x_i - y_i)^2}$

Tablo 1'deki ikinci yani 1 numaralı çiçeğin özellikleri sırası ile 4.9, 3.0, 1.4 ve 0.2'dir. Programınızı özellik sayısı değişken olacak şekilde geliştirmelisiniz.

Örnek olarak K değerini kullanıcı 3 girdiyse, verdiğiniz öznitelik dizisine uzaklığı en yakın (az) olan 3 çiçeği tespit etmelisiniz. İki tanesi *I. Versicolor,* bir tanesi de *I. Virginica* ise oy çokluğu ile bitkiyi *I. Versicolor* olarak sınıflandıracaksınız. Eğer oy çokluğu konusunda 1'den fazla bitki arasında eşitlik olursa en yakın bitkinin türünde sınıflandırabilirsiniz (k=1 için).

25 puan b) **Bitki sınıflandırma:** Yazdığınız kNN algoritmasının k değerini:

(i) yeni çiçeğin 4 adet özelliğini girdi olarak aldırarak,
(ii) yeni çiçeğin sadece 2 adet **çanak** yaprak özelliğini (uzunluk ve genişlik) girdi olarak aldırarak,
(iii) yeni çiçeğin sadece 2 adet **taç** yaprak özelliğini (uzunluk ve genişlik) girdi olarak aldırarak,

en yakın k adet bitkinin **özelliklerini, uzaklıklarını ve hangi sınıflardan olduklarını** bir tablo olarak **ekrana listeleyiniz**. Bağlantısı önceki sayfada verilen 150 satırlı verisetini kullanarak kNN yöntemi ile bitkinizin de türünü **tahminleyiniz** ve **ekrana yazdırınız**.

10 puan c) **Başarı Ölçümü:** Verisetinde her bir tür bitki örneğinin sonunda yer alan 10'ar veriyi test verisi olarak ayırınız. k değerini kullanıcıdan aldırarak, test verilerinden herbirini, a maddesindeki yöntemi ve 4 özelliğin tümünü kullanarak kalan 80% veri üzerinden sınıflandırınız, b'deki listelemeleri yapınız. Test verilerinin gerçek sınıfları ile, kNN ile tahminlediğiniz sınıflarını karşılaştırınız (gerçek ve tahminlenen türlerin / sınıfların her ikisini de yazdırınız). Başarı oranını,

doğru sınıflandırılan bitki sayısı / verisetinde test amaçlı kullandığınız toplam bitki sayısı

olarak hesaplayarak yazdırınız.

5 puan d) **Ekleme ve Silme İşlemleri:** Klavyeden bellekteki verisetine yeni örnek çiçek verisi (öznitelikler ve sınıf) ekleyen metodu yazınız. Sadece indisi verilen veriyi ve tüm verileri silen metotları yazınız.

5 puan e) **Listeleme:** Bellekteki verisetindeki değerleri görüntüleyen kodu yazınız.

- İlk proje olduğundan her projede tek sınıf kullanılması yeterlidir. Dileyenler, her bir çiçeğin bilgilerini birer nesnede tutup işlemlerini yürütebilirler. Listeler için dizi, matris, ArrayList gibi önereceğiniz veri yapılarını kullanabilirsiniz. Dosya kullanımı anlatılmadığı için dileyen öğrenciler, veriyi programa yapıştırarak ayrıştırabilirler veya https://docs.microsoft.com/en-us/dotnet/csharp/programming-guide/file-system/how-to-read-from-a-text-file.
- Kontrol sırasında, yeni örnek çiçek bilgileri eklenebilecek, programınız farklı değerler ile (örnek olarak k=1 ile 5 arasında değerler verilerek) test edilebilecektir. Programınız bunlara uygun şekilde cevap verebilmelidir. Varsayımlarınızı raporda belirtmelisiniz. Sınıf (tür) sayısının 3 olarak sabit alındığını varsayabilirsiniz.

RAPOR (10 puan) FORMATI – Bölüm 2

1.2.1. Gerçekleştirilen Platform ve Dil (C#/Java/Diğer) ve Sürüm Adı
1.2.2. Program 2'nin kısa tanımı (projeden)
1.2.3. Program 2 için kullanılan bileşik veri tiplerinin, sınıf ve metotların kısa açıklamaları
1.2.4. Elde edilen örnek sonuçlar (Ekran görüntüleri) (b,c,d,e seçenekleri için istenen listeler)
1.2.5. Proje 1.2 Yazılım Geliştirme İçin Harcanan Süreler (kişi ve saat bazında)
Ek 1: Program 1.2 a seçeneği: Sözdekod / Algoritma / Yöntem (Mantık) Anlatımı

10 puan **Özdeğerlendirme Tablosu**

Proje 1 Maddeleri	**Not**	**Tahmini Not**	**Açıklama**
1.1	15		
1.1 Rapor	10		
1.2.a	10		
1.2.b	25		
1.2.c	10		
1.2.d,e	10		
1.2 Rapor	10		
Özdeğerlendirme Tablosu	10		
Toplam	100		

Açıklama kısmında yapıldı, yapılmadı bilgisi veya hangi maddelerin nasıl yapıldığı kısaca yazılabilecektir.