DATA STRUCTURES PROJECT – 1A.1

BIRTHDAY PARADOX

* + 1. – Gerçekleştirilen Platform ve Dil : Eclipse Kepler / JAVA - jre 1.8
    2. – Rastgele seçilen n sayıda insanların aynı gün doğma olasılıklarının incelenmesi.
    3. – Fonksiyonlar ve Veri Tipleri :

FONKSİYONLAR -----------------------------------------------------------------------

anaMenu : Program açıldığında menüyü gösteren fonksiyon.

kizErkekDeney : Deneyin herkes için mi, kız erkek için ayrı ayrı mı gösterileceğini soran fonksiyon.

istatistikYaz : Deneylerin sonuçlarını istatistiksel olarak gösteren fonksiyon.

nCakisma : Toplam çakışma sayılarını 11X3 lük diziye alan fonksiyon.

herDeneydekiCakisma : Düzensiz diziye tüm çakışmaları alan fonksiyon.

deneyTekrarla : Deneyleri 10 kere tekrar eden fonksiyon.

herDeneyiGoster : Deney sonuçlarını ekrana bastıran fonksiyon.

generateRandomBD : n tane doğum günü üreten fonksiyon.

generateRandomBDBetweenTwoDates : Verilen yıllar arasında rastgele tarih üreten

fonksiyon.

kucukYilHesapla : Verilen yılların küçüğünü döndüren fonksiyon.

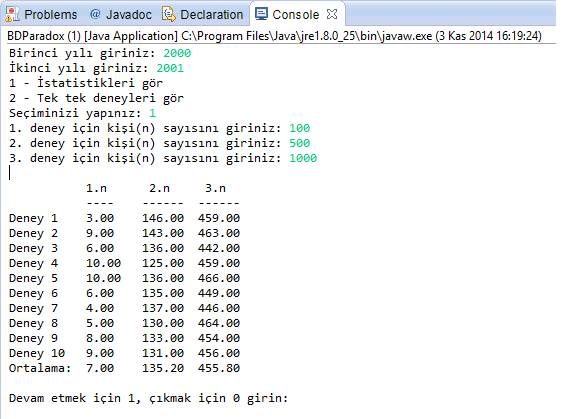
toplamYil : Verilen yıllara göre toplam yılı döndüren fonksiyon.

VERİ YAPILARI-----------------------------------------------------------------------

cakismaDizisi[][] : 11X3 lük tablonun tutulduğu dizi

sayac[][][] : Doğum tarihlerinin tutulduğu 3 boyutlu düzensiz dizi.

* + 1. – Ekran Görüntüleri :



* + 1. – Harcanan süreler :

Merve Gülcan Tecik – 96 saat

Çağrı YILDIZ – 96 saat

Tolgahan Akgün – 96 saat

DATA STRUCTURES PROJECT – 1A.2

RESERVATION SYSTEM

1.2.1 - Gerçekleştirilen Platform ve Dil : Eclipse Luna / JAVA jre\_1.8

1**.**2.2 - 0’la 100 arasında seçilen rastgele sayı kadar kişi üretip dengeli bir şekilde masalara oturtan rezervasyon sistemi.

1.2.3 - Masa class’ında kullanılan metodlar:

1. masayaOturt : string olarak parametre kişisini masaya oturutr
2. masadaKimlerVar : masada kimlerin olduğunu string dizisi olarak döndürür
3. kacKisiVar : masadaki kişi sayısını döndürür
4. masaDolulukYuzdesi : masanın doluluk yüzdesini döndürür
5. masaDizisiOlustur : önceden belirlenmiş sayıda masa nesnelerinden oluşmuş dizi döndürür
6. masalaraYerlestir : string dizisini parametre olarak alıp buradaki kişileri masaya oturtur
7. masalariYazdir : masaları GUI'ye yazdırır

Masa class’ında kullanılan veri tipleri:

1. String[] kisiler : Kişiler için string referans dizisi
2. int masadakiKisiSayisi : Masadaki kişi sayısını tutar

RandomName class’ında kullanılan metodlar:

1. isimleriBirlestir : isimlerin kartezyen kümesi oluşturuluyor
2. isimleriOku : oluşturulan isimleri string dizisi olarak döndürür

RandomName class’ında kullanılan veri tipleri:

1. String[] isimler: Daha önceden belirlenen isimleri tutan string dizisi
2. String[] soyIsimler: Daha önceden belirlenen soyisimleri tutan string dizisi
3. String[] karistirilmisIsimleString: İsimlerin kartezyen kümesi

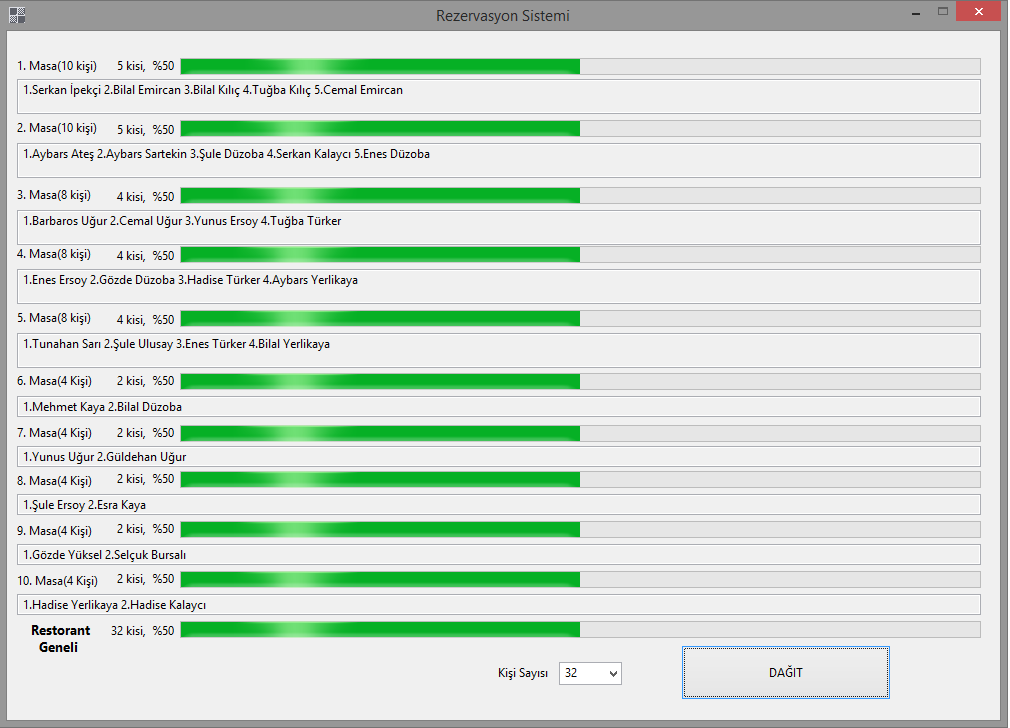
Main’de kullanılan veri tipleri:

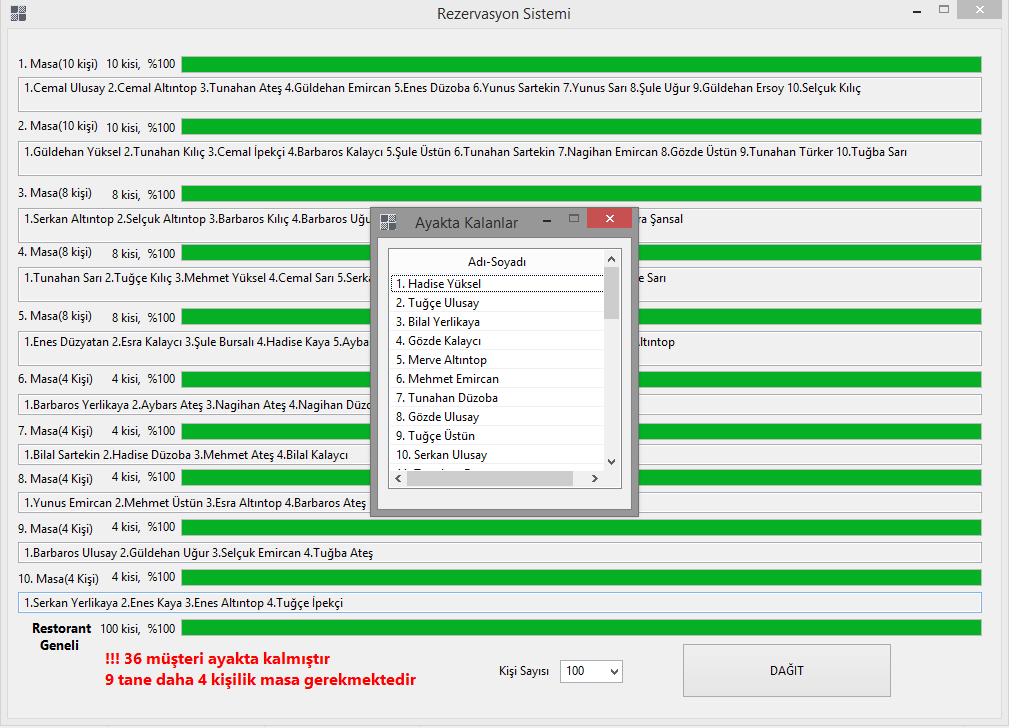
1. ArrayList<ProgressBar> pbarList : İlerleme çubuklarını public bir arraylist’de tutar
2. ArrayList<Text> txtKisler : Ekrana yazdırılacak kişileri public bir arraylist’de tutar
3. ArrayList<Label> lblList : Masa yüzdelerini ve kişi sayılarını public bir arraylist’de tutar.

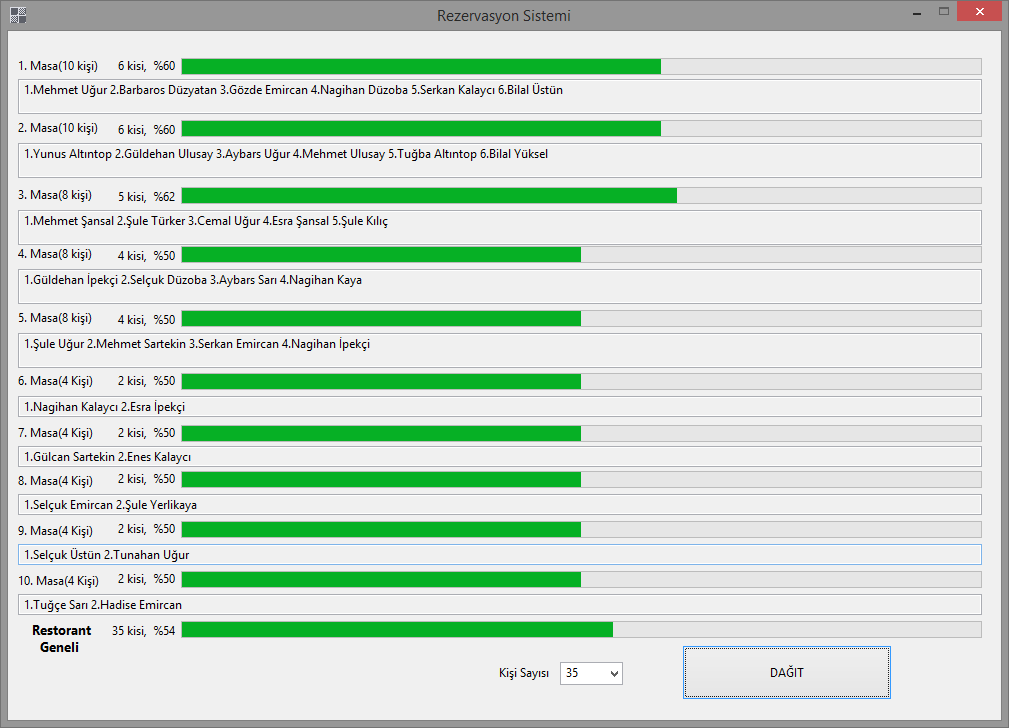
ArrayList Nedir ?

* Veri eklenip silindikçe ArrayList kendi uzunluğunu otomatik olarak ayarlar.
* ArrayList listesine erişim işlemi O(1), araya ekleme işlemi O(n) ve silme işlemi O(n) zaman karmaşasına sahiptir.
* ArrayList araya ekleme, silme ve arama eylemlerini yapan metotlara sahiptir.
* ArrayList üzerinde foreach döngüsü, iteratörler ve indexler yardımıyla gezilebilir.

1.2.4 – Ekran Görüntüleri :







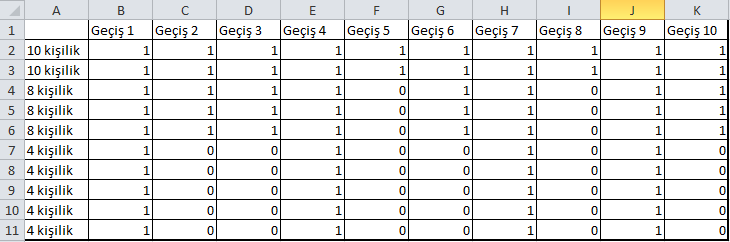
1.2.5 – Harcanan süreler :

Merve Gülcan Tecik – 96 Saat

Tolgahan Akgün – 96 Saat

Çağrı Yıldız – 96 Saat

Ek 1 –



For döngüsüyle bu matrisin içinde dolaşılır ve eğer karşılaşılan hücrenin değeri 1 ise hücrenin denk geldiği masaya kişi oturtulur. 0 ise bir sonraki hücreye bakılır ve bu şekilde kişi sayısı 0 olana kadar ve ya matris bitene kadar yerleştirme işlemi yapılır ve sonuç olarak da daha önceden belirlenen matristeki önceliklerine göre kişiler masalara oturtulur.

Ek 2 –

Sistemi tekrardan kapsamlı bir şekilde tasarlayacak olursak restorana ayrıca toplu olarak oturulabilecek salon ayrılırdı. Eğer gelen müşteriler belirli bir sayının(8) üzerindeyse toplu gelenler için ayrılan salona öncelikli olarak bu grupları yerleştirirdik.

10 kişilik masaları 5’er kişilik olarak ikiye ayrılabilir ve 8 kişilik masaları da 4’er kişilik masalara ayrılabilir şekilde tasarlardık. Bu şekilde restoran dolu iken gelen küçük grupları daha etkin bir şekilde oturtabiliriz. Bu sistemin daha profesyonel olması için görselliğe daha çok önem verirdik. Eğer büyük masalardan birisi boşalırsa ve yeni gelen müşteriler kalabalık değilse gelen müşteriler kısa süreli olarak öncelik kuyruğunda tutulabilir ve küçük masaların boşalması beklenebilir. Bu şekilde gelen grupların boyutuna göre boşalan masalarımızı dinamik ve etkin bir şekilde doldurabiliriz.