­**MultiThread ile Sayı Bulma**

B200109031 – Yavuz Selim ŞAHİN

Multi Thread ile Sayı Bulma Programı

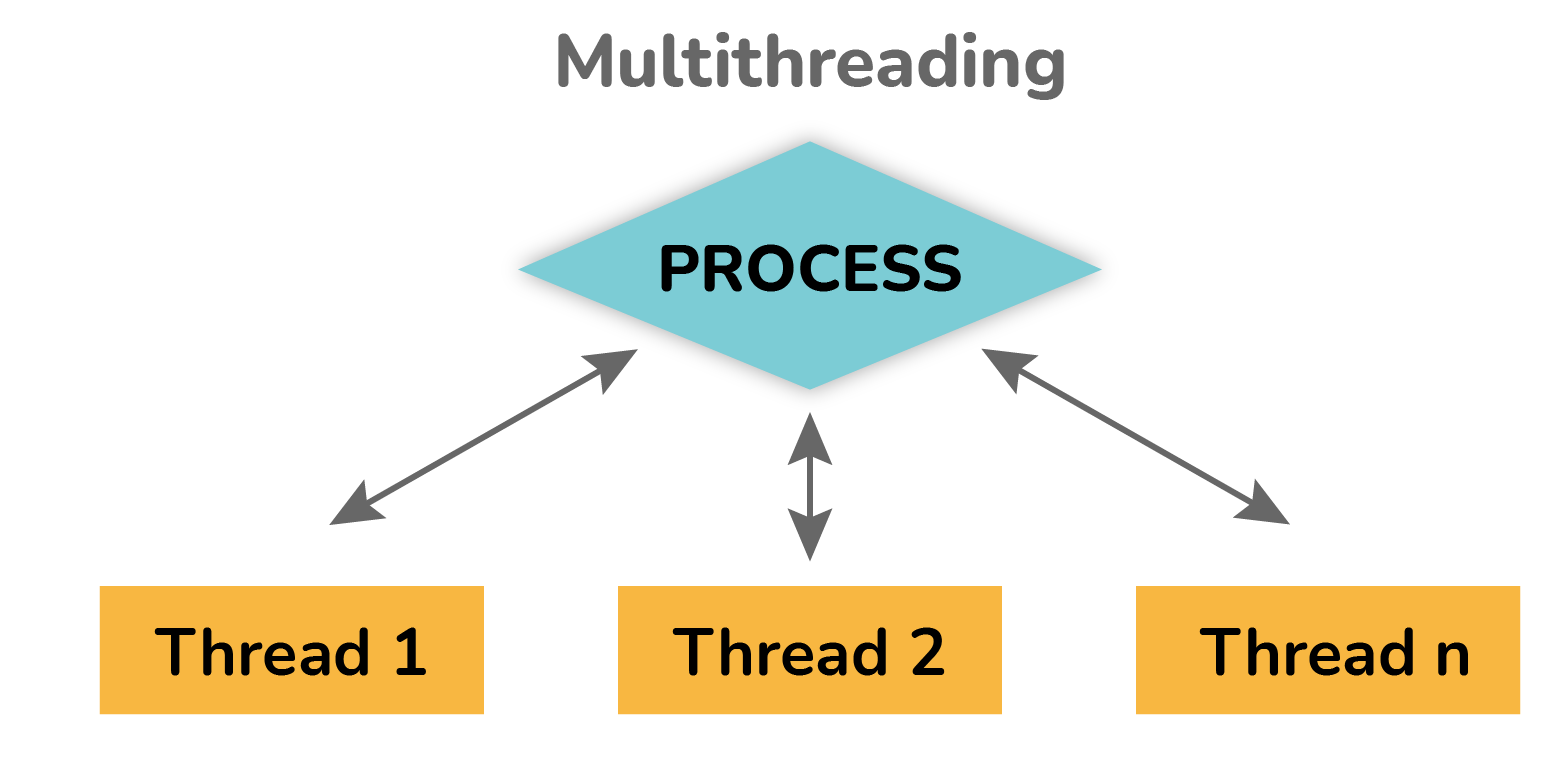
**İŞLETİM SİSTEMLERİ**

Teslim Tarihi:

**18.05.2023**

**Multi Thread :** Çoklu iş parçacığı (multithreading), bir program içinde birden fazla iş parçacığının aynı anda yürütülmesine olanak sağlayan bir programlama ve yürütme modelidir. Bir iş parçacığı (thread), bağımsız olarak çalışabilen ve programın belirli bir bölümünü yürüten küçük bir işlem birimidir. Çoklu iş parçacığı programlaması kullanılarak birden fazla iş parçacığı aynı anda yürütülerek işlem yapılabilir. Bu, programın performansını artırabilir ve işlem gücünü daha verimli bir şekilde kullanabilir. İş parçacıkları, eşzamanlı olarak çalışan farklı işlemleri gerçekleştirebilir. Örneğin, bir iş parçacığı dosya okurken diğer iş parçacığı veriyi analiz edebilir.

Şekilde bir işlemin threadlere bölünmesi verilmiştir.



**Multi Thread ile Sayıyı Bulma Programı**

Bu ödev kapsamında şöyle bir problem düşündüm. Öncelikle 0 ile 100 arasında , Java’nın Random sınıfı kullanılarak rastgele bir sayı üretilecektir. Daha sonra programımız MultiThread’lerle bu sayıyı bulmaya çalışacaktır. Bu random üretilen sayıyı bulmak için 3 farklı sınıf oluşturulmuş ve Runnable arayüzünden implement edilmiştir ardından run() methodu ezilmiştir ve run metodu içine rastgele üretilen sayıyı bulmak için geliştirilen yöntemler yazılmıştır. Rastgele üretilen sayı bulunduğunda o thread durur . Tüm thread’ler sayıyı rastgele üretilen sayıyı bulunca program sonlanır.

Thread 1 : LineerSearch sınıfı . Bu sınıf 0 ile 100 arasındaki üretilen rastgele sayıyı bulmak için 0’dan başlar ve her değer için sayının bulunup bulunmadığı kontrolünü yapar. Sırayla tüm sayılara bakacağı için mutlaka doğru sayıyı bulacak ve thread sonlanacaktır. Bu sınıf Thread sınıfından extends almaktadır.

**Şekil 1 : Thread 1 - LineerSearch Sınıfı**

metin, ekran görüntüsü içeren bir resim

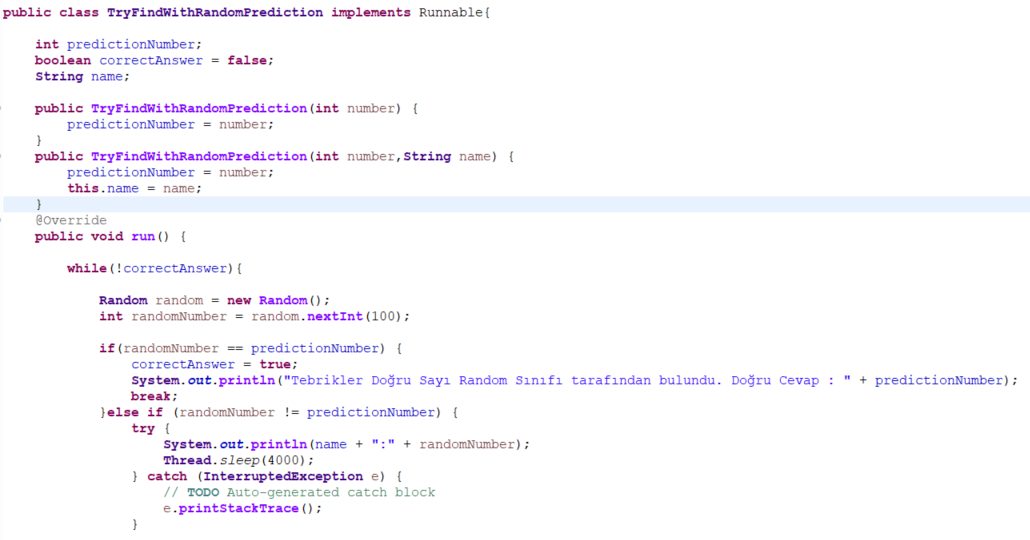
Açıklama otomatik olarak oluşturuldu

**Thread 2 :** TryFindWithRandomPrediction sınıfı , Bu sınıf problemimiz olan 0 – 100 arası rastgele sayıyı bulmak için kendisi de 0 ile 100 arası rastgele bir sayı üretir ve bu sayının eşit olup olmadığını kontrol eder. Tahmin edilecek sayıyı bulana kadar Thread 2 çalışmaya devam eder. Sayı bulunduğunda program durur.

Sınıfın içinde boolean correctAnswer = false ; tanımlanmıştır. While(!correctAnswer) yani doğru sayı bulunana kadar döngü devam eder ve Random sayı üretilerek işlemler tekrarlanır.

Kodun içindeki Thread.Sleep(4000); MultiThread işlemlerini görebilmemiz için yazılmıştır. Kodu 4 saniye kadar bekletir bu sayede kullanıcı olarak MultiThread işlemlerini görebiliriz. Aksi taktirde program çok hızlı şekilde sayıyı bulacaktır.

**Şekil 2 : Thread 2 - TryFindWithRandomPrediction Sınıfı**

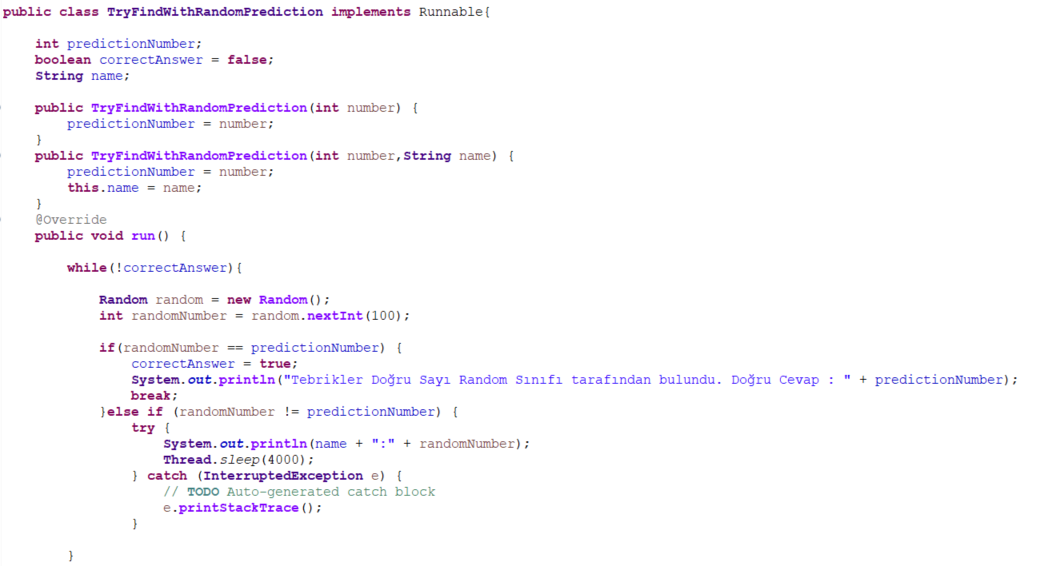


**Thread 3 :** WaitInput sınıfı , Runnable arayüzünü implement eder ve bu arayüzün run methodunu override eder. Bu sınıfta Random üretilen sayıyı bulmak için kullanıcı imputu girilmesi gerekmektedir.

Kullanıcıdan 0 ile 100 arasında bir input beklenir ve doğru sayıyu bulana kadar input girme işlemleri yapılabilir.

Kullanıcıdan input almak için Scanner sınıfından bir nesne üretilmiştir ve alınan input değeri integer bir değişkene atanmıştır. Ardından program bu integer değerin , tahmin edilecek sayıya eşit olup olmadığını kontrol eder . Eşit olana kadar program Thread devam eder.

**Şekil 3 : Thread 3 – WaitInput Sınıfı**



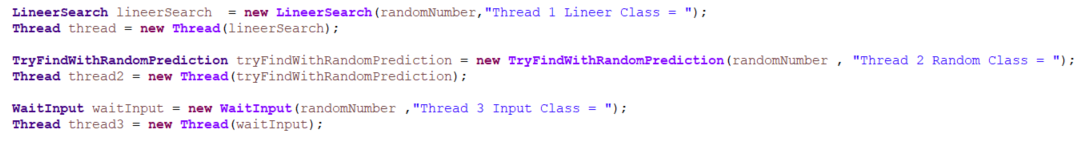
**Main Metot**

Main metotta öncelikle Random sınıfından 0 ile 100 aralığında rastgele bir sayı üretilir.



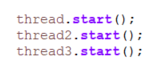
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Ardından 3 thread için oluşturduğumuz 3 sınıftan nesneler üretilir ve tahmin edilecek sayımız olan random sayısı , bu nesnelerin kurucu fonksiyonuna parametre olarak geçilir. Daha sonra bu sınıfların nesneleri Thread sınıfı kurucusuna parametre olarak geçilir ve Thread nesneleri oluşturulur.



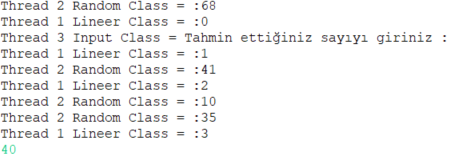
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Main metotta son olarak oluşturulan bu 3 thread çalıştırılır ve sayının tahmin edilmesi beklenir.



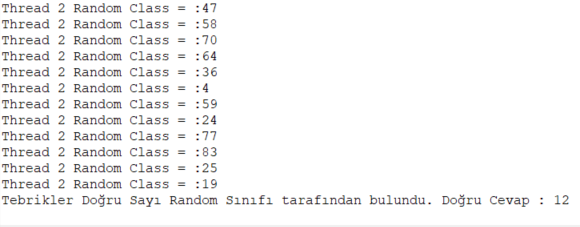
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**Ekran Çıktıları :**

**1)Çıktı-2metin, yazı tipi, ekran görüntüsü, beyaz içeren bir resim

Açıklama otomatik olarak oluşturuldu**

**metin, makbuz, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu**

**Çıktı-3 Çıktı-4**

Thread 2 Random Class = :68

Thread 1 Lineer Class = :0

Thread 3 Input Class = Tahmin ettiğiniz sayıyı giriniz :

Thread 1 Lineer Class = :1

Thread 2 Random Class = :41

Thread 1 Lineer Class = :2

Thread 2 Random Class = :10

Thread 2 Random Class = :35

Thread 1 Lineer Class = :3

40

Thread 2 Random Class = :37

Thread 1 Lineer Class = :4

Thread 3 Input Class = Tahmin ettiğiniz sayıyı giriniz :

Thread 2 Random Class = :57

Thread 1 Lineer Class = :5

50Thread 2 Random Class = :13

Thread 1 Lineer Class = :6

Thread 2 Random Class = :80

Thread 1 Lineer Class = :7

Thread 3 Input Class = Tahmin ettiğiniz sayıyı giriniz :

30Thread 2 Random Class = :18

Thread 1 Lineer Class = :8

Thread 2 Random Class = :4

Thread 1 Lineer Class = :9

Thread 3 Input Class = Tahmin ettiğiniz sayıyı giriniz :

20Thread 2 Random Class = :75

Thread 1 Lineer Class = :10

Thread 2 Random Class = :64

Thread 1 Lineer Class = :11

Thread 3 Input Class = Tahmin ettiğiniz sayıyı giriniz :

Thread 2 Random Class = :8

Tebrikler LineerSearch sınıfı sayıyı buldu , Sayı : 12

11

Thread 2 Random Class = :64

Thread 3 Input Class = Tahmin ettiğiniz sayıyı giriniz :

12

Thread 3 Input Class = Tebrikler Sayıyı buldunuz. Doğru Cevap : 12

Thread 2 Random Class = :10

Thread 2 Random Class = :36

Thread 2 Random Class = :10

Thread 2 Random Class = :98

Thread 2 Random Class = :47

Thread 2 Random Class = :58

Thread 2 Random Class = :70

Thread 2 Random Class = :64

Thread 2 Random Class = :36

Thread 2 Random Class = :4

Thread 2 Random Class = :59

Thread 2 Random Class = :24

Thread 2 Random Class = :77

Thread 2 Random Class = :83

Thread 2 Random Class = :25

Thread 2 Random Class = :19

Tebrikler Doğru Sayı Random Sınıfı tarafından bulundu. Doğru Cevap : 12

**EK :** MultiThread sayesinde 3 farklı işlemi aynı anda , birinin tamamlanmasını beklemeden yapabildik. Öneğin thread 3 teki kullanıcıdan input alınmazsa , program thread 1 e ve thread 2 ye bakacak. Input alınmasını bekleyip diğer işlemleri sonlandırmayacak. .Thread 1 , 2 ve 3 işlemcinin ayarlamasıyla çalıştırıldı . Doğru sonucu bulan thread sonlandırıldı. Ekran çıktısında en son Thread 2 Random Class kaldı . Bu thread’da random sayı üreterek en sonunda doğru cevabı buldu ve program sonlandı.

Thread yöntemi Thread sınıfını extends ederek yahut Runnable arayüzünü implment ederek kullanılabilir. Ben 2 thread için implement 1 thread için extend yaptım.