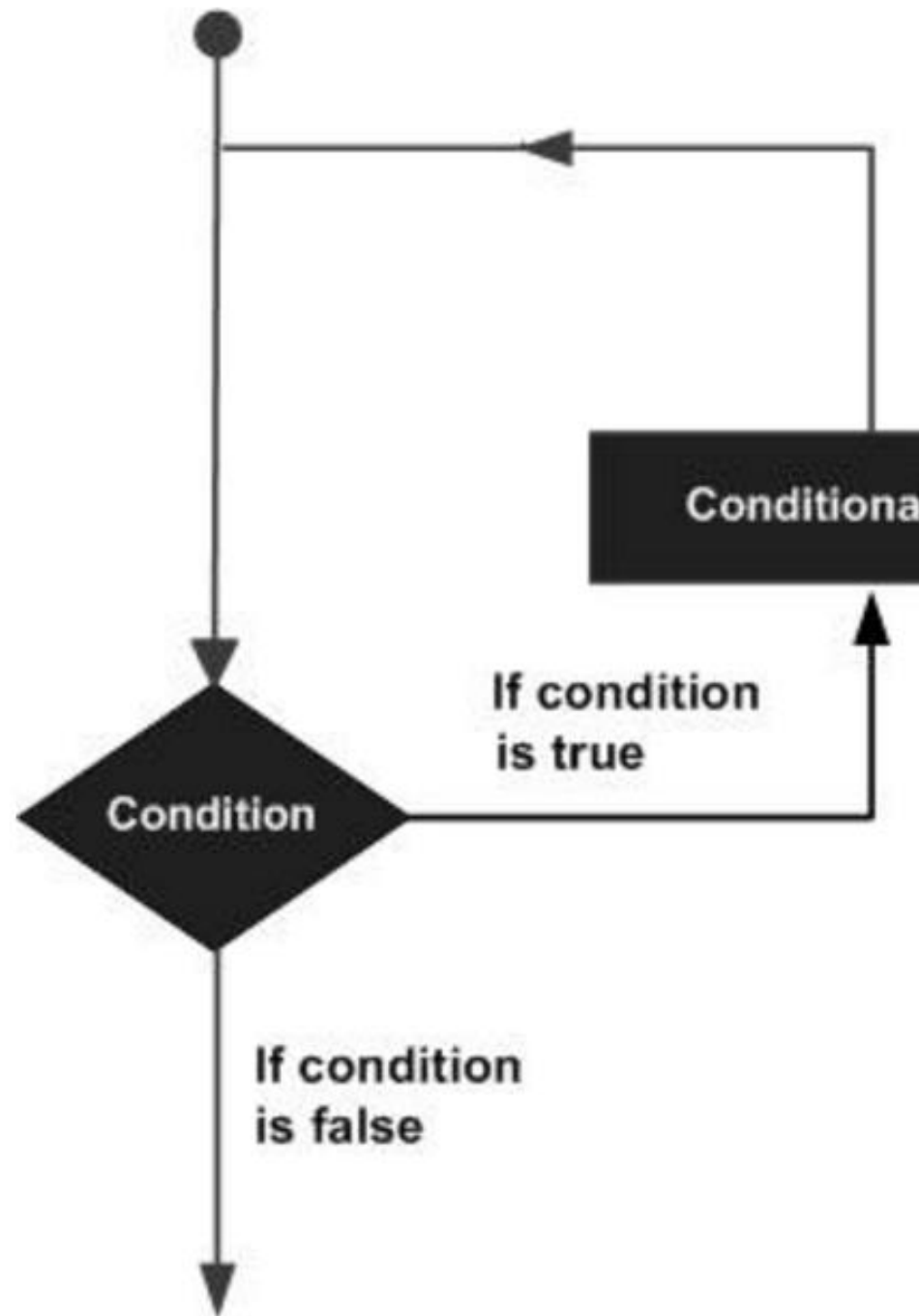


Algoritma ve Programlama II

Fatih Özkaynak

Döngüler

- Yol haritamız
 - Değişkenler
 - Operatörler
 - Kontrol yapıları
 - String sınıfı
- Döngüler



Döngüler

- Belirli bir iş bir çok kez tekrarlanacaksa, programda bu iş bir kez yazılır ve döngü deyimleriyle istenildiği kadar tekrar tekrar çalıştırılabilir.
- Bir döngüde, arka arkaya tekrarlanan deyimler döngü blokunu oluşturur.
- Bu deyimler birden çoksa { } bloku içine alınır.
- Bir döngü blokunda çok sayıda deyim olabileceği gibi, iç-içe döngüler de olabilir.
- Program akısının döngü blokunu bir kez icra etmesine döngünün bir adımı (bir tur) diyeceğiz.

Java'da Döngüler

- Java'da üç tür döngü vardır.
- Bunlar çoğu programlama dilinde olan; “for”, “do-while” ve “while” döngüleridir.
- Bu döngülerden for ve while döngüsünde, koşul sınaması döngüye girmeden önce yapılırken, do-while döngüsünde koşul sınaması bir kod parçası çalıştıktan sonra yapılır.

While Döngüsü

- While döngüsü bir şart sağlanıyor iken sürekli içindeki komutları çalıştırır. Kelime anlamı olarak "olduğu müddetçe" anlamı çıkar. Yapısı şöyledir:

```
1  while (koşul) {  
2      //komut 1  
3      //komut 2  
4      //...  
5      //komutlar  
6  }
```

- While döngüsünde koşulun en başta olması ayırt edici bir özelliktir.
- Program akışı önce buradaki koşulu mantıksal bir süzgeçten geçirir.
- Buna göre eğer koşula yazılan ifade matematiksel olarak **true (doğru)** değer döndürüyorsa, süslü parantezlerle çevrili *bloktaki* komutlar sırasıyla çalıştırılır.
- Ancak **false (yanlış)** değerini döndürüyorsa, bu komutlar hiçbir zaman çalışmayacaktır.

```
1  public class WhileDongusu1 {  
2      public static void main(String[] args) {  
3  
4          int sayac=10;  
5          while (sayac>=1) {  
6              System.out.println(sayac);  
7              sayac--;  
8          }  
9      }  
10 }
```

```
1 public class WhileDongusu2 {
2     public static void main(String[] args) {
3
4         int faktoriyel=1;
5         int carpan=1;
6         while(carpan<=10) {
7             faktoriyel=faktoriyel*carpan; //1*1, 1*2, 2*3, 6*4, 24*5...
8             System.out.println(carpan+" faktoriyel: "+faktoriyel);
9             carpan++;
10        }
11    }
12 }
```


do ... while Döngüsü

- while deyiminin önemli bir özelliği, test koşulunun döngünün başlangıcında olmasıdır.
- Bunun anlamı, ilk anda koşulun yanlış olması halinde, while gövdesinin hiçbir zaman çalıştırılmayacağıdır.
- Ancak gövdeyi hiç olmazsa bir kere çalıştırmamız gereken durumlar vardır.
- Bu durumlar çok yaygın olmasa da gerektiği zaman do...while deyimini kullanmalıyız.

```
do {  
    Deyim1;  
    Deyim2;  
    ...  
    Deyim_n;  
} while(İfade) ;
```

```
int sayac = 1;  
do {  
    System.out.println("Sayaç: " + sayac);  
    sayac++;  
} while (sayac < 11);
```

For Döngüsü

```
1 | for(ifade1;ifade2;ifade3)  Deyim;  
2 |                             Deyim_x;
```

veya

```
1 | for(ifade1;ifade2;ifade3)  
2 |     { Deyim1;  
3 |       Deyim2;  
4 |       ...  
5 |       Deyim_n;  
6 |     }  
7 |     Deyim_x;
```

- İlk olarak ifade1 hesaplanır; bu genellikle bir ya da daha fazla değişkene ilk değeri veren bir atama deyimidir.
- Sonra ifade2 hesaplanır. Bu, deyimin koşulsal kısmıdır.
- ifade2 yanlışsa, programın kontrolü for deyiminden çıkar ve program akışı programdaki bir sonraki deyim'e geçer (deyim_x). Eğer ifade2 doğru ise, deyim veya deyimler grubu uygulanır.
- Deyim ya da blok yapıları şeklinde deyimler grubu çalıştırıldıktan sonra, ifade3 çalıştırılarak hesaplanır. O zaman çevrim yine geriye, ifade2'ye döner.

For – While Karşılaştırması

```
1 | for (ifade1; ifade2; ifade3)  
2 |   deyim;
```

```
1 | ifade1;  
2 | while (ifade2)  
3 | {  
4 |     deyim;  
5 |     ifade3;  
6 | }
```

```
1  for(i=1; i<=n; i=i+1) {  
2      top=top+i;  
3  }  
4  System.out.println("\n1+2+3+...+n+=" +top);  
5  System.out.println("\ni="+i);
```

```
1 public class Alterne {
2     public static void main(String args[]){
3         int i,p;
4         double x,s;
5         s=0;
6         p=1;
7         x=1.0;
8         for (i=1;i<=4;i++)
9         {
10             s=s+(x*(x+1))/((x+2)*(x+3))*p;
11             x=x+4;
12             p=-p;
13         }
14         System.out.println("Toplam= "+s);
15     }
16 }
```

```
3 public class IcIce {
4     public static void main(String args[]){
5         int a,i;
6
7         //5 kez tekrarla
8         for(a=1;a<=5;a++)
9         {
10             System.out.println("a= "+a);
11
12             //3 kez tekrarla
13             for(i=1;i<=3;i++) {
14                 System.out.println("i= "+i);
15             }
16             System.out.println();
17         }
18     }
19 }
```

```
3 public class CarpinTablosu {
4     public static void main(String args[]){
5         int altcizgi, i, j;
6         System.out.println("          1    2    3    4    5    6    7    8    9    10 ");
7         //döngüyle uzunca bir çizgi çizelim
8         for(altcizgi=1; altcizgi<=70; altcizgi++){
9             System.out.print("_");
10        }
11        System.out.println();
12
13        for (i =1; i<=10; i++) {
14            System.out.print(i+" | ");
15
16            for (j=1 ; j<=10 ; j ++){
17                System.out.print(i*j +" ");
18            }
19            System.out.println();
20        }
21    }
22 }
```



```
1 public class FaktoriyelliSeri {
2     public static void main(String args[]){
3         double s,fakt;
4         int i,j;
5         s=0;
6         for (i=2;i<=10;i++) {
7             fakt=1;
8
9             for(j=1;j<=i;j++) {
10                 fakt=fakt*j;
11             }
12
13             s=s+1/fakt;
14             System.out.println(s);
15         }
16         System.out.println("Seri toplami "+s);
17     }
18 }
```

```
3 public class FaktoriyelliSeri2 {
4     public static void main(String args[]){
5         double s,fakt;
6         int i;
7         s=0;fakt=1;
8
9         for (i=2; i<=10; i++)    {
10             fakt=fakt*i;
11             s=s+1/fakt;
12             System.out.println(s);
13         }
14         System.out.println("Seri toplami: "+s);
15     }
16 }
```

break Deyimi

- break deyiminin Java programları içinde iki farklı kullanım alanı vardır:
 - switch yapısındaki case seçeneklerinden birinde switch yapısını terk etmek ve switch'i izleyen deyime geçmek için kullanılır.
 - Bir döngü (loop) yapısı içinden, döngüyü kontrol eden koşul ifadesini beklemeksizin döngü dışına çıkmak için kullanılır.
- break deyimi bir döngü içinde yer almışsa bu durumda break deyimi ile karşılaşır karşılaşmaz döngü dışına çıkılır ve döngüyü izleyen deyime geçilir.

```
public class BreakClass{
    public static void main(String args[]) {
        Scanner giris = new Scanner(System.in);
        double ort;
        int sayac, toplam, sayi;
        toplam = 0;
        sayac = 0;
        System.out.println("Alt alta pozitif sayilar gireceksiniz ve negatif sayi");
        System.out.println("girene kadar onlar toplanacak. Negatif girdiginizde");
        System.out.println("toplama isleminin sonucu size verilecektir.");

        while (true) {
            System.out.print("Sayi giriniz: ");
            sayi = giris.nextInt();

            if (sayi < 0) {
                break;
            }

            toplam += sayi;
            sayac++;
        }
        ort = toplam / sayac;

        System.out.println("Toplami "+toplam+" olan "+sayac+" tane sayi girdiniz.");
        System.out.println("Sayilarin ortalamasi da "+ort);
    }
}
```

continue Deyimi

- continue deyimi, döngü içinde belirli bir koşulun kontrolüyle bir sonraki döngü adımına gidilmesini gerçekleştirir.
- for döngüsünün parametresi olan i değişkeni 3 ile tam bölünemeyen bir değer alınca, bu değer ile işlem yapılmayıp bir sonraki adıma geçilmesi sağlayan programı yazınız. Yani program, kullanıcı tarafından girilen bir üst limite kadar, 3 ile bölünebilen sayıların toplamını ve bu tip sayıların adedini bulsun.

```
public class Cont{
    public static void main(String args[]) {
        Scanner giris = new Scanner(System.in);
        int toplam, i, ustlimit, sayac;

        System.out.println("Gireceginiz ust limite kadar olan ve");
        System.out.println("3'e bolunebilen sayilar bulunacaktır.");
        System.out.println();
        System.out.println("Ust limiti giriniz: ");
        ustlimit = giris.nextInt();
        toplam = 0;
        sayac = 0;

        for (i = 1; i <= ustlimit; i++) {
            if (i % 3 != 0) {
                continue;
            }

            System.out.println("3'e bolunebilen sayi: " + i);
            toplam += i;
            sayac++;
        }

        System.out.println("3-" + ustlimit + " arasindaki 3 ile bolunebilen "+sayac+" tane");
        System.out.println("Bu tur sayilarin toplami " + toplam);
    }
}
```

```
/* javaprogram.java */  
public class javaprogram {  
    public static void main(String[] args) {  
        int id1;  
  
        for (id1=1; id1<6; id1++) System.out.println(id1);  
    }  
}
```

```
/* javaprogram.java */  
public class javaprogram {  
    public static void main(String[] args) {  
        int[] sayi = { 1, 2, 3, 4, 5 };  
  
        for (int deger : sayi) {  
            System.out.println(deger);  
        }  
    }  
}
```

```
import java.util.Random;

import javax.swing.JOptionPane;

public class donguler1
{
    public static void main(String[] args)
    {
        // dongu1();
        //dongu2();
        //dongu3();
        //dongu4();
        //carpimTablosu();
        carpimTablosu2();
        siralama();
    }
}
```



```
static void dongu2()
{
    int sayi = 0;
    int toplam =0;

    for(int i=0; i<50; i++)
    {
        if(i%7==0)
        {
            sayi++;
            toplam+=i;
        }
    }
    System.out.println("1 Den 50 ye kadar 7' ye bolunen sayı: " +sayi);
    System.out.println("Bu sayıların toplamı: " + toplam +" toplamdır");
}
```

```
static void dongu3()
{
    Random rastgele = new Random();

    int sayi[] = new int[20];

    for(int i=0; i<20;i++)
    {
        sayi[i] = rastgele.nextInt(100);
        System.out.print(sayi[i] + " - ");
    }

}
```

```
static void dongu4()
{
    double sayi[][]= new double[2][4];
    sayi[0][0]=1.2;
    sayi[0][1]=1.3;
    sayi[0][2]=1.5;
    sayi[0][3]=1.7;
    sayi[1][0]=2.2;
    sayi[1][1]=2.3;
    sayi[1][2]=2.5;
    sayi[1][3]=2.7;

    //System.out.println("1. boyutun 2 elemanının değeri: " + sayi[1][

    for(int i=0; i<sayi.length; i++)
    {
        for(int j=0; j<sayi[i].length; j++)
        {
            //System.out.println(i +". Satır "+ j +". sütun değeri " +
            System.out.print(sayi[i][j] + "\t");
        }
        System.out.print("\n");
    }
}
```

```
static void siralama()
{
    Random rastgele = new Random();

    int sayi[] = new int[5];

    for(int i=0; i<sayi.length; i++)
    {
        sayi[i] = rastgele.nextInt(1000);
        System.out.print(sayi[i]+"-");
    }

    int gecici = 0;

    for(int i =0; i<sayi.length;i++)
    {
        for(int k =0; k<sayi.length;k++)
        {
            if(sayi[k] < sayi[i])
            {
                gecici = sayi[i];
                sayi[i] = sayi[k];
                sayi[k] = gecici;
            }
        }
    }

    System.out.println("-----\n");

    for(int m : sayi) // bu işlem bir dizinin içeriğini yazdırmada kullanılır
    {
        System.out.print(m + "-");
    }
}
```

```
public class fordonguyildiz {  
    public static void main(String[ ] args){  
        for(int satir = 0; satir < 4; satir++){ // 1. For Döngüsü (Dış Döngü)  
            System.out.print("*");  
            for(int sayac = 0; sayac < satir; sayac++){ // 2. For Döngüsü (İç Döngü)  
                System.out.print("*");  
  
            } // İç döngü sonu  
  
            System.out.print("\n"); // bu kod çıktıda satır atlatmak için  
        } // Dış döngü sonu  
  
    } // main method sonu  
  
} // class fordonguyildiz sonu
```

1'den kullanıcının girdiği sayıya kadar ekrana yazdırmak.

```
1  package hafta2;
2  import java.util.*;
3  public class Donguler {
4
5      public static void main(String[] args) {
6          int sayi;
7          Scanner giris=new Scanner(System.in);
8          System.out.println("Bir sayı giriniz");
9          sayi = giris.nextInt();
10         for (int i = 1; i < sayi; i++) {
11             System.out.println(i);
12         }
13     }
14
15 }
```

0 ile 100 arasındaki 5'e bölünen sayıları ekrana yazdırma

```
1  package hafta2;
2
3  public class Donguler {
4
5      public static void main(String[] args) {
6          for (int i = 0; i < 100; i++) {
7              if (i%5==0) {
8                  System.out.println(i);
9              }
10         }
11     }
12
13 }
```

10'dan geriye 1'er 1'er yazdırma

```
1  package hafta2;
2
3  public class Donguler {
4
5      public static void main(String[] args) {
6          for (int i = 10; i >= 0; i--) {
7              System.out.println(i);
8          }
9      }
10
11 }
```



```
public class Program {  
    public static void main(String[] args) {  
  
        String[] values = new String[3];  
        values[0] = "Dot";  
        values[1] = "Net";  
        values[2] = "Perls";  
  
        for (String value : values) {  
            System.out.println(value);  
        }  
    }  
}
```

Output

Dot
Net
Perls

```
public class Program {  
    public static void main(String[] args) {  
  
        int[] values = { 1, 2, 3, -1 };  
  
        // ... Loop over array indexes, but break on negative one.  
        for (int i = 0; i < values.length; i++) {  
            if (values[i] == -1) {  
                break;  
            }  
            System.out.println(values[i]);  
        }  
    }  
}
```

Output

1
2
3

```
public class Program {  
    public static void main(String[] args) {  
  
        String[] values = { "cat", "bear", "dog", "bird" };  
  
        // Loop over all Strings.  
        for (String value : values) {  
  
            // Skip Strings starting with letter b.  
            if (value.startsWith("b")) {  
                continue;  
            }  
  
            System.out.println(value);  
        }  
    }  
}
```

Output

cat
dog

```
public class Program {  
    public static void main(String[] args) {  
  
        int x = 0;  
        // Parts of the for-loop can be omitted.  
        // ... Here we use no variable declaration in the for-statement.  
        for (; x < 3; x++) {  
            System.out.println(x);  
        }  
        System.out.println("x is still reachable!");  
        System.out.println(x);  
    }  
}
```

Output

```
0  
1  
2  
x is still reachable!  
3
```

```
public class Program {
```

```
    static int count;
```

```
    static int[] getElements() {  
        // Set array elements based on a static field.  
        int[] array = new int[3];  
        array[0] = count++;  
        array[1] = count++;  
        array[2] = count++;  
        return array;  
    }
```

```
    public static void main(String[] args) {
```

```
        // The method is called once and not many times in the for-loop.  
        for (int value : getElements()) {  
            System.out.println(value);  
        }
```

```
    }
```

```
}
```

Output

0

1

2

```
public class Program {  
    public static void main(String[] args) {  
  
        long iterations = 0;  
        // Count iterations from 100 to 200 decrementing.  
        for (int u = 100; u <= 200; u--) {  
            iterations++;  
        }  
        System.out.println("Iterations from 100 to 200: " + iterations);  
    }  
}
```

Output

Iterations from 100 to 200: 2147483749

```
public class Program {  
    public static void main(String[] args) {  
  
        String value = "art";  
  
        // Loop from 0 to length() of the string.  
        for (int i = 0; i < value.length(); i++) {  
            // Get letters with charAt method.  
            char letter = value.charAt(i);  
            System.out.println(letter);  
        }  
    }  
}
```

Output

a
r
t

```

public class Program {
    public static void main(String[] args) {

        int[] array1 = { 10, 20, 30 };
        int[] array2 = { 20, 10, 30 };
        int[] array3 = { 40, 40, 10 };

        long t1 = System.currentTimeMillis();

        // Version 1: loop over each array separately.
        for (int i = 0; i < 10000000; i++) {

            int sum = 0;
            for (int x = 0; x < array1.length; x++) {
                sum += array1[x];
            }
            for (int x = 0; x < array2.length; x++) {
                sum += array2[x];
            }
            for (int x = 0; x < array3.length; x++) {
                sum += array3[x];
            }
            if (sum != 210) {
                System.out.println(false);
            }
        }
    }
}

```

```

        long t2 = System.currentTimeMillis();

        // Version 2: jam loops together.
        for (int i = 0; i < 10000000; i++) {
            int sum = 0;
            for (int x = 0; x < array1.length; x++) {
                sum += array1[x];
                sum += array2[x];
                sum += array3[x];
            }
            if (sum != 210) {
                System.out.println(false);
            }
        }

        long t3 = System.currentTimeMillis();

        // ... Times.
        System.out.println(t2 - t1);
        System.out.println(t3 - t2);
    }
}

```

109 ms, 3 for-loops
48 ms, 1 for-loop (jammed)


```
public class Program {  
    public static void main(String[] args) {  
  
        int[] array1 = new int[5];  
  
        long t1 = System.currentTimeMillis();  
  
        // Version 1: assign elements in a loop.  
        for (int i = 0; i < 10000000; i++) {  
            for (int x = 0; x < array1.length; x++) {  
                array1[x] = x;  
            }  
        }  
  
        long t2 = System.currentTimeMillis();
```

56 ms, for-loop

17 ms, unrolled statements

```
        // Version 2: unroll the loop and use  
        for (int i = 0; i < 10000000; i++) {  
            array1[0] = 0;  
            array1[1] = 1;  
            array1[2] = 2;  
            array1[3] = 3;  
            array1[4] = 4;  
        }
```

```
        long t3 = System.currentTimeMillis();
```

```
        // ... Times.  
        System.out.println(t2 - t1);  
        System.out.println(t3 - t2);
```

```
public class Program {  
    public static void main(String[] args) {  
  
        // Create new ArrayList.  
        ArrayList<Integer> elements = new ArrayList<>();  
  
        // Add three elements.  
        elements.add(10);  
        elements.add(15);  
        elements.add(20);  
  
        // Get size and display.  
        int count = elements.size();  
        System.out.println("Count: " + count);  
  
        // Loop through elements.  
        for (int i = 0; i < elements.size(); i++) {  
            int value = elements.get(i);  
            System.out.println("Element: " + value);  
        }  
    }  
}
```

Output

```
Count: 3  
Element: 10  
Element: 15  
Element: 20
```

```
public class Program {  
    public static void main(String[] args) {  
  
        int i = 0;  
        int z = 10;  
  
        // Loop with two variables.  
        while (i < z) {  
            i++;  
            z--;  
  
            // Display the values.  
            System.out.println(i + "/" + z);  
        }  
    }  
}
```

Output

1/9

2/8

3/7

4/6

5/5

```
import java.lang.Math;

public class Program {
    public static void main(String[] args) {

        // Loop infinitely.
        while (true) {

            // Get random number between 0 and 1.
            double value = Math.random();
            System.out.println(value);

            // Break if greater than 0.8.
            if (value >= 0.8) {
                break;
            }
        }
    }
}
```

Output

```
0.16129889659284657
0.0977987643977064
0.859556475501672
```

```
public class Program {  
    public static void main(String[] args) {  
  
        int i = 0;  
        // Loop while the variable is less than 3.  
        // ... It is not checked on the first iteration.  
        do {  
            System.out.println(i);  
            i++;  
        } while (i < 3);  
    }  
}
```

Output

0
1
2

```
public class Program {  
    public static void main(String[] args) {  
  
        int index = 0;  
        // Use post increment in while-loop expression.  
        while (index++ < 10) {  
  
            // Continue if even number.  
            if ((index % 2) == 0) {  
                continue;  
            }  
            System.out.println("Element: " + index);  
        }  
    }  
}
```

Output

Element: 1
Element: 3
Element: 5
Element: 7
Element: 9

Enter Number:

5

Enter X:

1

0

1 2

3 4 5

6 7 8 9

10 11 12 13 14

*

**

Enter Number:

5

Enter X:

2

0

2 4

6 8 10

12 14 16 18

20 22 24 26 28

**

*

*

**

**

*

Enter Number:

5

Enter X:

3

0

3 6

9 12 15

18 21 24 27

30 33 36 39 42

**

*

*

**
