

laços = repetições = loop
início, fim, passo
teste

// testes de início / fim
contados 20 n = vezes definido
com sentinela: ñ se sabe o
n = de vezes

no início ↓ pode ter nenhuma repetição	{	for — contados	{	com sentinela
		while		
no final — ↓ é realizada pelo menos uma repetição		do-while		

início: condição ou valor de contador

```
while (condição verdadeira “fim”) {  
    instruções que se repetem;  
    passo = atualizar a condição ou o contador  
}
```

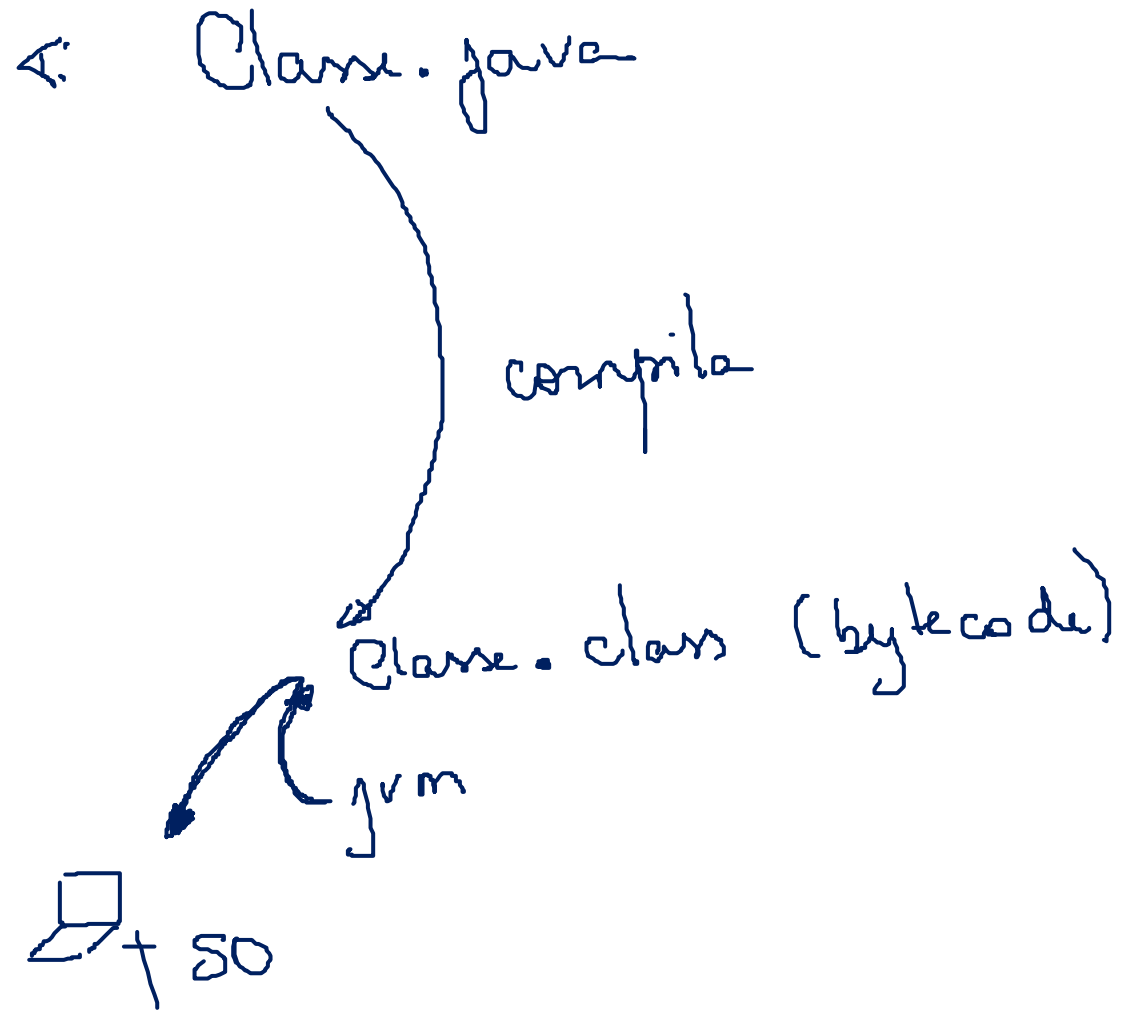
```
for (início: condição ou valor de contador; condição verdadeira “fim”; passo = atualizar a condição ou o contador) {  
    instruções que se repetem;  
}
```

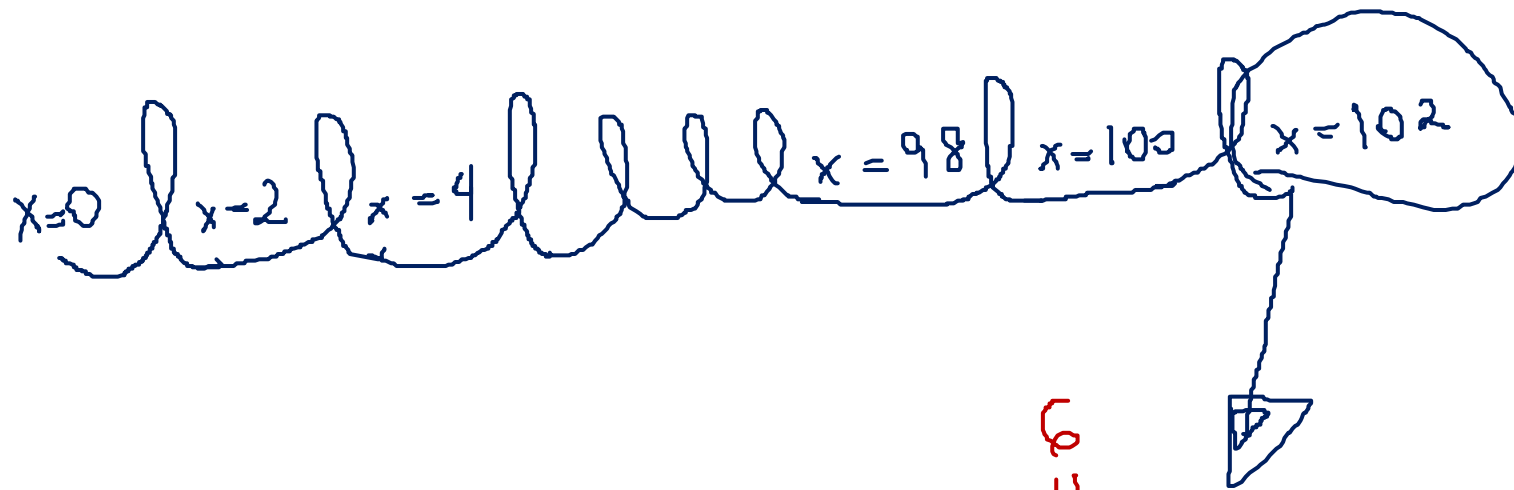
início: condição ou valor de contador // opcional

```
do {  
    instruções que se repetem;  
    passo = atualizar a condição ou o contador; //pode ser o valor inicial  
} while (condição verdadeira “fim”) ;
```

Acumuladores

- `a = a + 5;`
- `b = b - 3;`
- `m = m * 10;`
- `n = n / 2;`
- `String s = "oi"`
- `s = s + "mundo"`
- `x = x + 1;`
- `y = y - 1;`
- `a += 5;`
- `b -= 3;`
- `m *= 10;`
- `n /= 2;`
- `s += "mundo"`
- `x++; //incremento`
- `y--; //decremento`





```

for (int x=0; x <= 100; x = x + 2) {
    msg = msg + x + " ";
}

```

msg = ""
 x = 0
 2
 4
 6
 8
 2, 4
 2, 4, 6
 2, 4, 6, 8

$$5! = 5 \times 4 \times 3 \times 2 \times 1 = 120$$

$$3! = 3 \times 2 \times 1 = 6$$

$$n! = n \times (n-1) \times (n-2) \dots \times 1$$

$$\underline{n \leq 13}$$

$$\text{int} : -2^{31} \text{ --- } 2^{31} - 1$$

```

import java.util.Scanner;
public class Fatorial {
    public static void main(String[] args) {
        Scanner entrada = new Scanner(System.in);
        System.out.print ("digite seu numero: ");
        int n = entrada.nextInt();
        int fatorial = n;
        while (n > 1) {
            n--;
            fatorial = fatorial * n;
        }
        System.out.println ("fatorial = " + fatorial);
        entrada.close();
    }
}

```

n	fat
5	5
4	20
3	60
2	120
1	120