

Operadores

E2.1

```
int multiplicacion(int numero1, int numero2) {  
    return numero1 * numero2;  
}
```

E2.2

```
int suma(int numero1, int numero2) {  
    return numero1 + numero2;  
}
```

E2.3

```
boolean positivoMenorDe20(int numero) {  
    return numero > 0 && numero < 20;  
}
```

E2.4

```
int truncado(int n) {  
    return n / 10;  
}
```

E2.5

```
int ultimoDigito(int n) {  
    return n % 10;  
}
```

E2.6

```
boolean unicoDigito(int n) {  
    return (n >= 0) && (n < 10);  
}
```

E2.7

```
String signo(int n) {  
    return (n >= 0) ? "no-negativo" : "negativo";  
}
```

```
}
```

E2.8

```
boolean esPar(int numero) {  
    return numero % 2 == 0;  
}
```

E2.9

```
boolean esImpar(int numero) {  
    return numero % 2 != 0;  
}
```

E2.10

```
int diferenciaDeCuadrados(int a, int b) {  
    return (a * a) - (b * b);  
}
```

E2.11

```
double tercerAngulo(double a, double b) {  
    return 180.0 - (a + b);  
}
```

E2.12

```
boolean validarMultiploDe7(int a) {  
    return (a > 0) && (a % 7 == 0) && (a < 1000);  
}
```

E2.13

Escreva uma função `validar()` que receba três números inteiros: `inicio`, `meio` e `tamanho` e retorna `true` se `<inicio>` for maior ou igual a 0, `<inicio>` for menor que `<meio>` e `<meio>` for menor que `<tamanho>`.

```
boolean validar(int inicio, int meio, int tamanho) {  
    return (inicio >= 0) && (inicio < meio) && (meio < tamanho);  
}
```

E2.14

Um estacionamento abre às 8:00 horas e fecha às 18:00 horas. Seu sistema de cobrança é o seguinte: O usuário paga \$10,0 por cada minuto de estacionamento, mas tem uma cobrança mínima de \$100,0 e uma cobrança máxima de \$3000,0. Crie uma função `cobro()` que retorne um número `double` sendo a cobrança para um usuário, dado um número inteiro representando o número de minutos que seu veículo ficou no estacionamento.

Para criar essa função, você pode usar esses dois métodos da classe `Math`:

- [https://docs.oracle.com/en/java/javase/17/docs/api/java.base/java/lang/Math.html#min\(double,double\)](https://docs.oracle.com/en/java/javase/17/docs/api/java.base/java/lang/Math.html#min(double,double))
- [https://docs.oracle.com/en/java/javase/17/docs/api/java.base/java/lang/Math.html#max\(double,double\)](https://docs.oracle.com/en/java/javase/17/docs/api/java.base/java/lang/Math.html#max(double,double))

É uma boa ideia escrever o código de forma iterativa. Faça uma primeira versão da função que a torne o mais simples possível. Em seguida, adicione um pouco de funcionalidade e verifique se ela funciona como esperado e assim por diante.

Para este exercício, a recomendação seria:

- Crie uma função que calcule o valor a ser cobrado levando em conta apenas o tempo
- Em seguida, adicione a funcionalidade para que a função nunca retorne um valor menor que US\$ 100,0.
- Por fim, adicione a funcionalidade para que ela também não retorne um valor maior que US\$ 3.000,0.

```
double calculoMinutoXValorPorMinuto(int minutos) {
    return minutos * 10;
}
double cobrar(int minutos) {
    if (calculoMinutoXValorPorMinuto(minutos) <= 100) return 100.0;
    if (calculoMinutoXValorPorMinuto(minutos) >= 3000) return 3000.0;
    return calculoMinutoXValorPorMinuto(minutos);
}
```

E2.15

Para esses problemas, veja os exemplos de uso de uma função e, em seguida, revise o código inicial que não está funcionando como esperado. Execute-o e, em seguida, corrija-o para que corresponda aos exemplos.

```
int inicio(int resultados, int pagina) {
    return (resultados * (pagina - 1)) ;
}
```