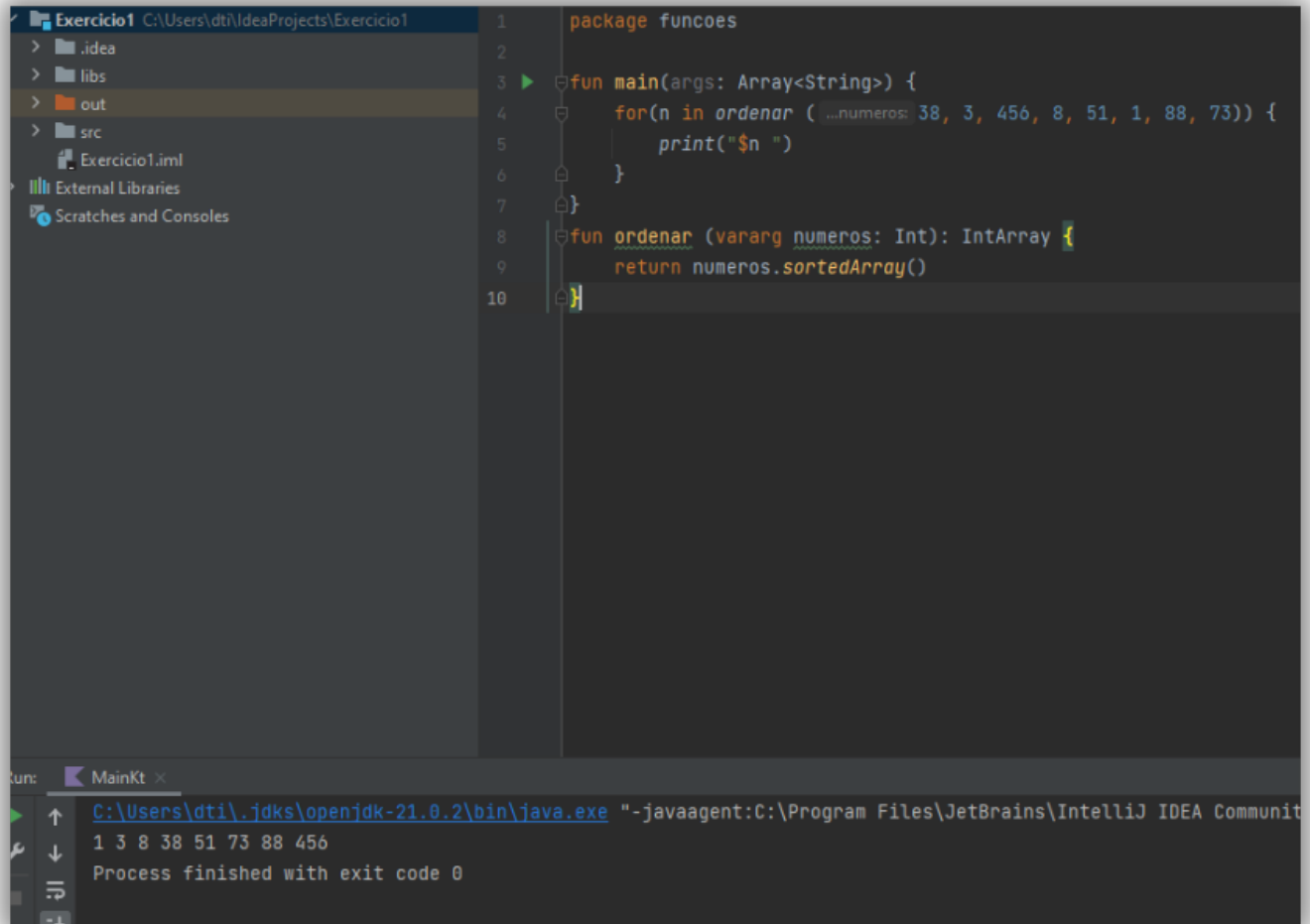


# Exercícios de PAMII

## Exercício 1



The screenshot displays the IntelliJ IDEA IDE interface. On the left, the Project Explorer shows the file structure of 'Exercicio1', including folders like .idea, libs, out, and src, and files like Exercicio1.iml. The main editor window shows the following Kotlin code:

```
1 package funcoes
2
3 fun main(args: Array<String>) {
4     for(n in ordenar ( ...numeros: 38, 3, 456, 8, 51, 1, 88, 73)) {
5         print("$n ")
6     }
7 }
8 fun ordenar (vararg numeros: Int): IntArray {
9     return numeros.sortedArray()
10 }
```

At the bottom, the Run console shows the execution of the program. The output is:

```
1 3 8 38 51 73 88 456
Process finished with exit code 0
```

## Exercício 2

The screenshot shows an IDE with a project named 'Exercicio2'. The code is written in Kotlin and defines a package 'funcoes', a class 'Operacoes', and several functions. The 'main' function calls 'calc' with arguments 2, 3, and the 'somar' function from the 'Operacoes' class. The output window shows the result of the calculation, which is 5, and confirms that the process finished with exit code 0.

```
1 package funcoes
2
3
4 class Operacoes {
5     fun somar(a: Int, b: Int): Int {
6         return a + b
7     }
8 }
9
10 fun somar(a: Int, b: Int): Int {
11     return a + b
12 }
13
14 fun calc(a: Int, b: Int, funcao: (Int, Int) -> Int): Int {
15     return funcao(a, b)
16 }
17
18 fun main(args: Array<String>) {
19     println(calc(a: 2, b: 3, Operacoes().somar))
20     println(calc(a: 2, b: 3, ::somar))
21 }
22
```

Run: MainKt x

C:\Users\dti\.jdk\openjdk-21.0.2\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA\bin\instrumented\idea\_rt.jar=5000:C:\Program Files\JetBrains\IntelliJ IDEA\bin\idea.jar" -Dfile.encoding=UTF-8

5  
5

Process finished with exit code 0

## Exercício 3

The screenshot shows an IDE with a Kotlin file. The code defines a generic function `filtrar` that filters a list based on a predicate, a function `comTresLetras` that checks if a string has three characters, and a `main` function that uses them. A tooltip for `filtrar` is visible, stating "No documentation found." The bottom panel shows the program running successfully, outputting the filtered list `[Ana, Bia, Gui]`.

```
2  
3 fun <E> filtrar(lista: List<E>, filtro: (E) -> Boolean): List<E> {  
4     val listaFiltrada = ArrayList<E>()  
5     for (e in lista) {  
6         if (filtro(e)) {  
7             listaFiltrada.add(e)  
8         }  
9     }  
10    return listaFiltrada  
11 }  
12  
13 fun comTresLetras(nome: String): Boolean {  
14     return nome.length == 3  
15 }  
16  
17 fun main(args: Array<String>) {  
18     val nomes = listOf("Ana", "Pedro", "Bia", "Gui", "Rebeca")  
19  
20     println(filtrar(nomes, comTresLetras))  
21  
22 }
```

Run: MainKt x

C:\Users\dti\.jdk\openjdk-21.0.2\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ ID  
[Ana, Bia, Gui]

Process finished with exit code 0

## Exercício 4

The image shows a screenshot of an IDE with a Kotlin file named `Main.kt`. The code defines a package `funcoes` and an inline function `transacao` that takes a `funcao: () -> Unit` and prints "Abrindo transação...". It then uses a `try-finally` block to call `funcao()` and print "Fechando transação". The `main` function calls `transacao` with three different SQL execution messages.

```
1 package funcoes
2
3 inline fun transacao(funcao: () -> Unit){
4     println("Abrindo transação...")
5
6     try {
7         funcao()
8     } finally {
9         println("Fechando transação")
10    }
11 }
12
13 fun main(args: Array<String>) {
14     transacao {
15         println("Executando SQL 1...")
16         println("Executando SQL 2...")
17         println("Executando SQL 3...")
18     }
19 }
```

The Run console shows the output of the program:

```
Run: MainKt
C:\Users\dti\.jdk\openjdk-21.0.2\bin\java.exe "-javaagent:C:\Program Files\JetBrains\Int
Abrindo transação...
Executando SQL 1...
Executando SQL 2...
Executando SQL 3...
Fechando transação
Process finished with exit code 0
```

## Exercício 5

```
Exercicio5 C:\Users\dti\IdeaProjects 1 package funcoes
2
3 inline fun <T> executarComLog(nomeFuncao: String, funcao: () -> T): T{
4     println("Entrando no Método $nomeFuncao...")
5
6     try {
7         return funcao()
8     } finally {
9         println("Método $nomeFuncao finalizado..")
10    }
11 }
12
13 fun somar(a: Int, b: Int): Int{
14     return a + b
15 }
16
17
18 fun main(args: Array<String>) {
19
20     val resultado = executarComLog( nomeFuncao: "Somar"){
21         somar( a: 4, b: 5)
22     }
23
24     println(resultado)
25 }
```

MainKt x

C:\Users\dti\.jdk\openjdk-21.0.2\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community

Entrando no Método Somar...

Método Somar finalizado..

9

Process finished with exit code 0

## Exercício 6 – Calculadora de Áreas

### I. Retângulo

```
C:\Users\otl\Documents\openjdk-21.0.2\bin\java.exe -javaagent:C:\Program Files\Je
0l seja bem-vindo a CALCULADORA DE ÁREAS:)
0 primeiro passo e escolher qual forma geometrica para calcular a área:
1. Retângulo
2. Triângulo
3. Triângulo Equilátero
4. Triângulo Isósceles
5. Triângulo Escaleno
6. Circunferência
7. Losango
8. Trapézio
1
Digite a base do retângulo:
12
Digite a altura do retângulo:
15
A área do retângulo é: 180.0

Process finished with exit code 0
```

## 2- TRIÂNGULOS

```
Olá seja bem-vindo a CALCULADORA DE ÁREAS:)
0 primeiro passo e escolher qual forma geometrica para calcular a área:
1. Retângulo
2. Triângulo
3. Triângulo Equilátero
4. Triângulo Isósceles
5. Triângulo Escaleno
6. Circunferência
7. Losango
8. Trapézio
2
Digite a base do triângulo:
12
Digite a altura do triângulo:
12
A área do triângulo é: 72.0

Process finished with exit code 0
```

### 3- TRIANGULO EQUILÁTERO

```
Olá seja bem-vindo a CALCULADORA DE ÁREAS:)
0 primeiro passo e escolher qual forma geometrica para calcular a área:
1. Retângulo
2. Triângulo
3. Triângulo Equilátero
4. Triângulo Isósceles
5. Triângulo Escaleno
6. Circunferência
7. Losango
8. Trapézio
2
Digite a base do triângulo:
12
Digite a altura do triângulo:
12
A área do triângulo é: 72.0

Process finished with exit code 0
```

## L-TRIÂNGULO ISÓCELES



```
01. seja bem-vindo a CALCULADORA DE ÁREAS:)
0 primeiro passo e escolher qual forma geometrica para calcular a área:
1. Retângulo
2. Triângulo
3. Triângulo Equilátero
4. Triângulo Isósceles
5. Triângulo Escaleno
6. Circunferência
7. Losango
8. Trapézio
4
Digite a base do triângulo isósceles:
12
Digite o lado do triângulo isósceles:
12
A área do triângulo isósceles é: 62.353829072479584

Process finished with exit code 0
|
```

## 5-Triângulo Escaleno

```

01❖ seja bem-vindo a CALCULADORA DE ❖REAS:)
0 primeiro passo e escolher qual forma geometrica para calcular a ❖rea:
1. Ret❖ngulo
2. Tri❖ngulo
3. Tri❖ngulo Equil❖tero
4. Tri❖ngulo Is❖sceles
5. Tri❖ngulo Escaleno
6. Circunfer❖ncia
7. Losango
8. Trap❖zio
❖
Digite o lado 1 do tri❖ngulo escaleno:
12
Digite o lado 2 do tri❖ngulo escaleno:
12
Digite o lado 3 do tri❖ngulo escaleno:
12
A ❖rea do tri❖ngulo escaleno ❖: 62.353829072479584

Process finished with exit code 0
|

```

## 6-ÁREA DA CIRCUNFERÊNCIA

```

01❖ seja bem-vindo a CALCULADORA DE ❖REAS:)
0 primeiro passo e escolher qual forma geometrica para calcular a ❖rea:
1. Ret❖ngulo
2. Tri❖ngulo
3. Tri❖ngulo Equil❖tero
4. Tri❖ngulo Is❖sceles
5. Tri❖ngulo Escaleno
6. Circunfer❖ncia
7. Losango
8. Trap❖zio
❖
Digite o raio da circunfer❖ncia:
12
A ❖rea da circunfer❖ncia ❖: 452.3893421169302

Process finished with exit code 0
|

```

## 7- ÁREA DO LOSANGO

```
01. seja bem-vindo a CALCULADORA DE ÁREAS:)
0 primeiro passo e escolher qual forma geometrica para calcular a área:
1. Retângulo
2. Triângulo
3. Triângulo Equilátero
4. Triângulo Isósceles
5. Triângulo Escaleno
6. Circunferência
7. Losango
8. Trapézio
7
Digite a diagonal maior do losango:
12
Digite a diagonal menor do losango:
12
A área do losango é: 72.0

Process finished with exit code 0
|
```

## 8-ÁREA DO TRAPÉZIO

```
Olá seja bem-vindo a CALCULADORA DE ÁREAS:)
O primeiro passo é escolher qual forma geometrica para calcular a área:
1. Retângulo
2. Triângulo
3. Triângulo Equilátero
4. Triângulo Isósceles
5. Triângulo Escaleno
6. Circunferência
7. Losango
8. Trapézio
8
Digite a base maior do trapézio:
12
Digite a base menor do trapézio:
12
Digite a altura do trapézio:
12
A área do trapézio é: 144.0

Process finished with exit code 0
|
```