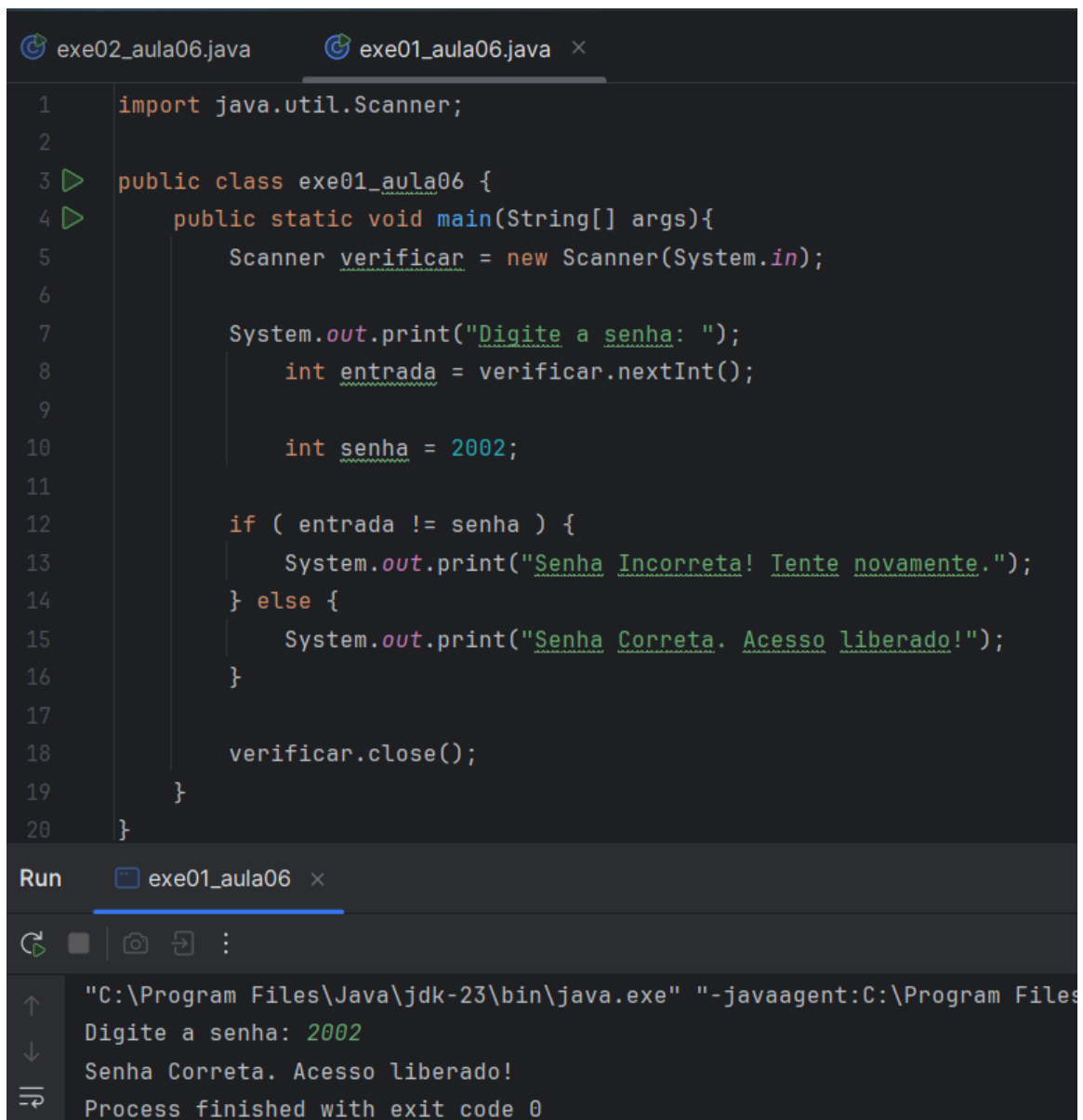


ESTRUTURAS DE REPETIÇÃO - LING. DE PROGRAMAÇÃO

1) Escreva um programa que repita a leitura de uma senha até que seja válida. Para cada leitura de senha incorreta informada, escreva a mensagem “Senha Incorreta”. Quando a senha for informada corretamente, deve-se imprimir a mensagem “Acesso Permitido” e o algoritmo encerrado. Considere que a senha correta seja: “2002”.



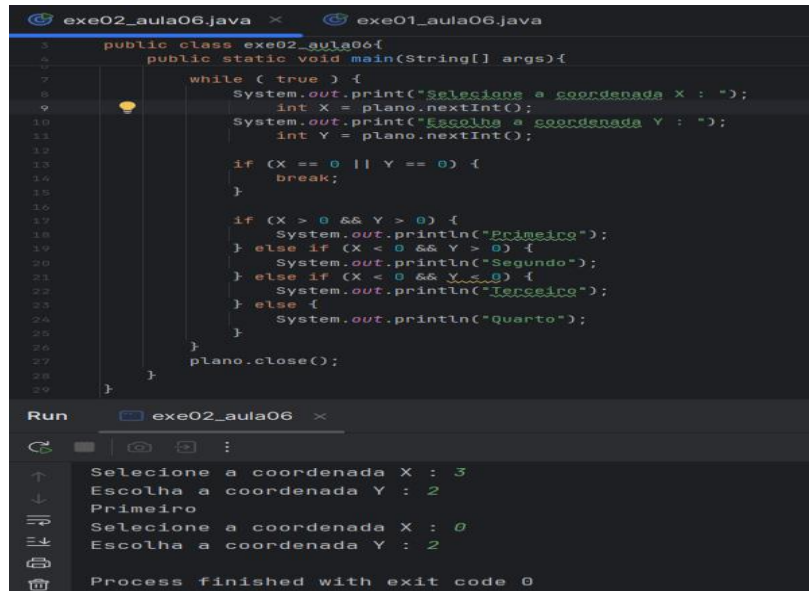
```
exe02_aula06.java  exe01_aula06.java  x
1  import java.util.Scanner;
2
3  public class exe01_aula06 {
4      public static void main(String[] args){
5          Scanner verificar = new Scanner(System.in);
6
7          System.out.print("Digite a senha: ");
8          int entrada = verificar.nextInt();
9
10         int senha = 2002;
11
12         if ( entrada != senha ) {
13             System.out.print("Senha Incorreta! Tente novamente.");
14         } else {
15             System.out.print("Senha Correta. Acesso liberado!");
16         }
17
18         verificar.close();
19     }
20 }
```

Run exe01_aula06 x

↑ "C:\Program Files\Java\jdk-23\bin\java.exe" "-javaagent:C:\Program Files
↓ Digite a senha: 2002
↑ Senha Correta. Acesso liberado!
↑ Process finished with exit code 0

Fonte: Autoria Própria

2) Escreva um programa que leia as coordenadas (X , Y) de uma quantidade indeterminadas de pontos no Sistema Cartesiano, onde cada ponto inserido, exibe o quadrante do qual pertence-o. O algoritmo será encerrado quando, pelo menos, um valor da coordenada for nulo (nesta situação, desconsidere a exibição de mensagem).



```
public class exe02_aula06 {
    public static void main(String[] args) {
        while ( true ) {
            System.out.print("Selecione a coordenada X : ");
            int X = plano.nextInt();
            System.out.print("Escolha a coordenada Y : ");
            int Y = plano.nextInt();

            if (X == 0 || Y == 0) {
                break;
            }

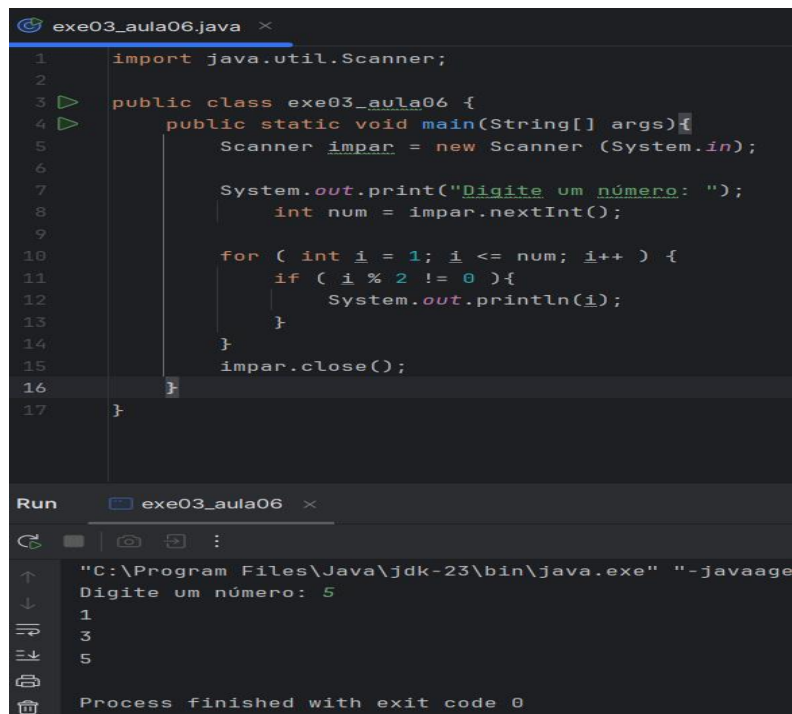
            if (X > 0 && Y > 0) {
                System.out.println("Primeiro");
            } else if (X < 0 && Y > 0) {
                System.out.println("Segundo");
            } else if (X < 0 && Y < 0) {
                System.out.println("Terceiro");
            } else {
                System.out.println("Quarto");
            }
        }
        plano.close();
    }
}
```

Run exe02_aula06

```
Seleção a coordenada X : 3
Escolha a coordenada Y : 2
Primeiro
Seleção a coordenada X : 0
Escolha a coordenada Y : 2
Process finished with exit code 0
```

Fonte: Autoria Própria

3) Leia um valor inteiro X (1 <= X <= 1000). Em seguida, mostre os números ímpares de 1 até X, cada valor por linha, incluindo o próprio X se for o caso.



```
import java.util.Scanner;

public class exe03_aula06 {
    public static void main(String[] args) {
        Scanner impar = new Scanner (System.in);

        System.out.print("Digite um número: ");
        int num = impar.nextInt();

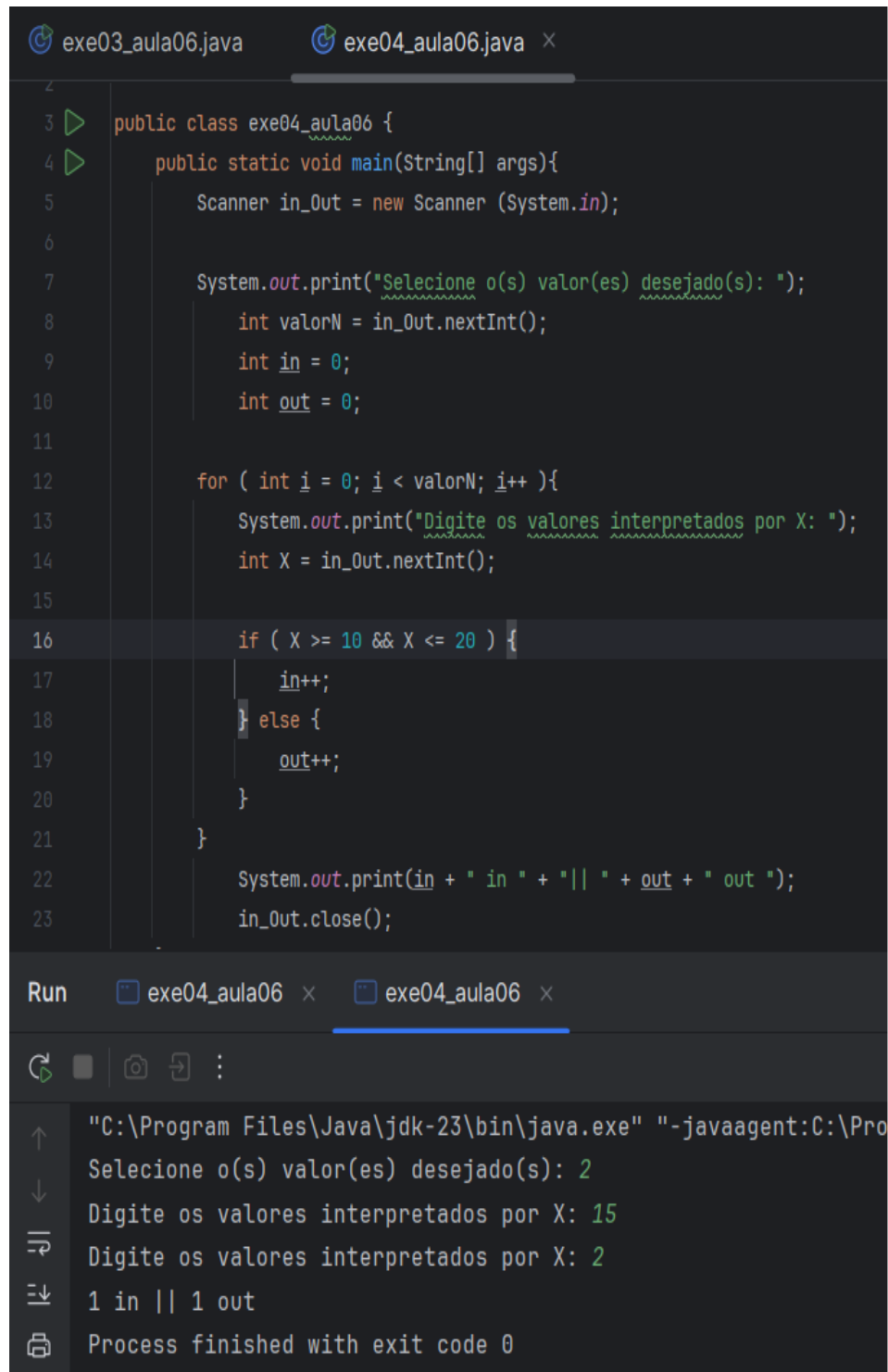
        for ( int i = 1; i <= num; i++ ) {
            if ( i % 2 != 0 ) {
                System.out.println(i);
            }
        }
        impar.close();
    }
}
```

Run exe03_aula06

```
"C:\Program Files\Java\jdk-23\bin\java.exe" "-javaage
Digite um número: 5
1
3
5
Process finished with exit code 0
```

Fonte: Autoria Própria

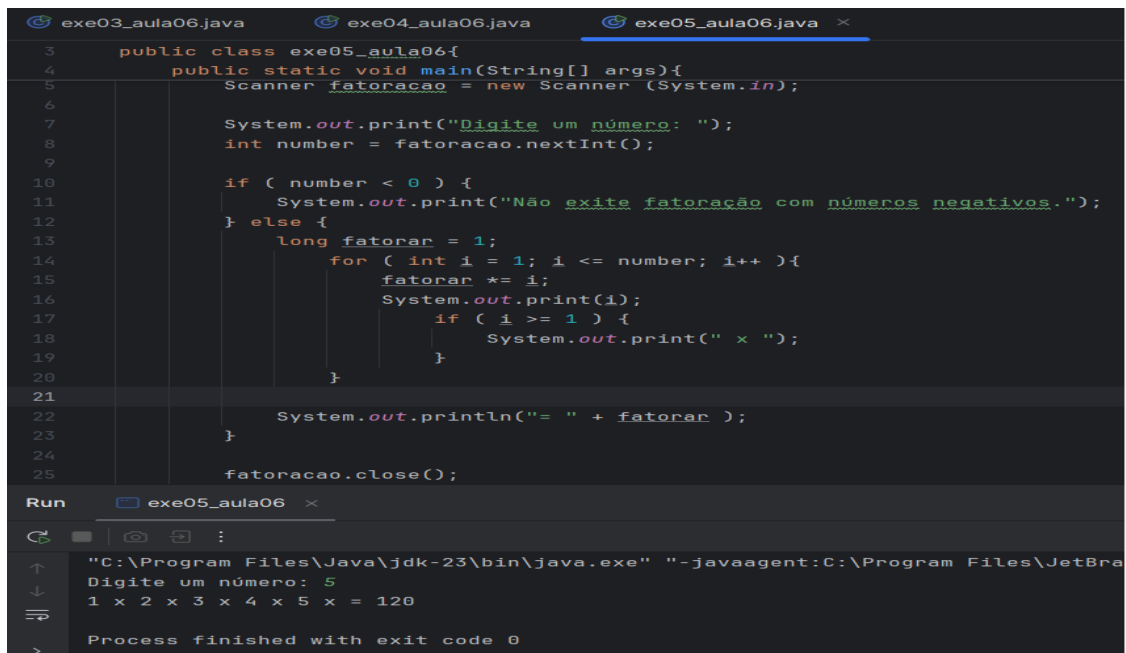
4) Leia um valor inteiro “N”. Este valor será a quantidade de valores inteiros “X” que serão lidos em seguida. Exiba a quantidade de valores X que estão inseridos dentro dos intervalos [10 | 20] e apresente os demais valores fora do intervalo. Use a palavra “ IN ” para destacar os números inseridos no intervalo e a palavra “ OUT ” aos demais números deslocados do intervalo.



```
exe03_aula06.java  exe04_aula06.java x
2
3  public class exe04_aula06 {
4  public static void main(String[] args){
5      Scanner in_Out = new Scanner (System.in);
6
7      System.out.print("Selecione o(s) valor(es) desejado(s): ");
8      int valorN = in_Out.nextInt();
9      int in = 0;
10     int out = 0;
11
12     for ( int i = 0; i < valorN; i++ ){
13         System.out.print("Digite os valores interpretados por X: ");
14         int X = in_Out.nextInt();
15
16         if ( X >= 10 && X <= 20 ) {
17             in++;
18         } else {
19             out++;
20         }
21     }
22     System.out.print(in + " in " + "|| " + out + " out ");
23     in_Out.close();
24 }
25
Run  exe04_aula06 x  exe04_aula06 x
C:\Program Files\Java\jdk-23\bin\java.exe" "-javaagent:C:\Pro
Selecione o(s) valor(es) desejado(s): 2
Digite os valores interpretados por X: 15
Digite os valores interpretados por X: 2
1 in || 1 out
Process finished with exit code 0
```

Fonte: Autoria Própria

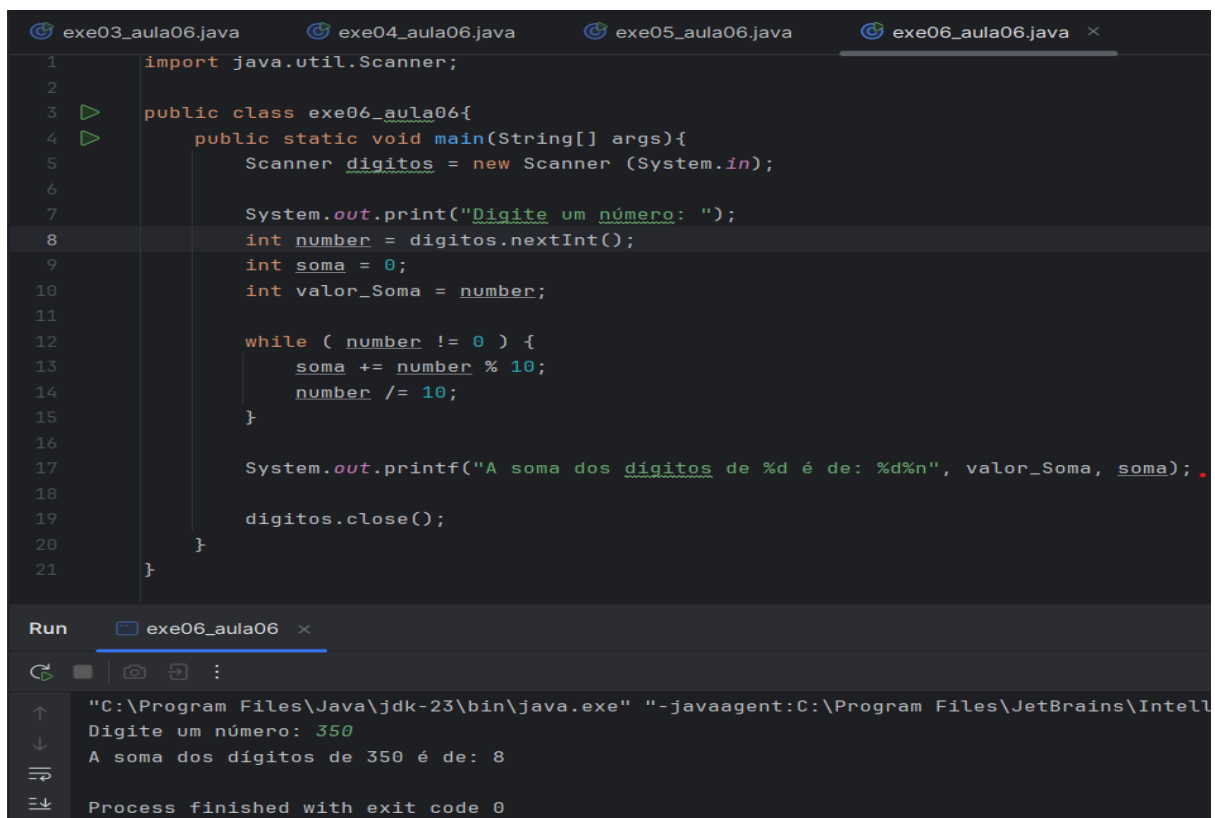
5) Faça um programa que calcule o fatorial de um número fornecido pelo usuário.



```
exe03_aula06.java  exe04_aula06.java  exe05_aula06.java x
3      public class exe05_aula06{
4          public static void main(String[] args){
5              Scanner fatoracao = new Scanner (System.in);
6
7              System.out.print("Digite um número: ");
8              int number = fatoracao.nextInt();
9
10             if ( number < 0 ) {
11                 System.out.print("Não existe fatoração com números negativos.");
12             } else {
13                 long fatorar = 1;
14                 for ( int i = 1; i <= number; i++ ){
15                     fatorar *= i;
16                     System.out.print(i);
17                     if ( i >= 1 ) {
18                         System.out.print(" x ");
19                     }
20                 }
21
22                 System.out.println(" = " + fatorar );
23             }
24
25             fatoracao.close();
26
27         }
28     }
29
30 Run  exe05_aula06 x
31
32 "C:\Program Files\Java\jdk-23\bin\java.exe" "-javaagent:C:\Program Files\JetBra
33 Digite um número: 5
34 1 x 2 x 3 x 4 x 5 x = 120
35
36 Process finished with exit code 0
```

Fonte: Autoria Própria

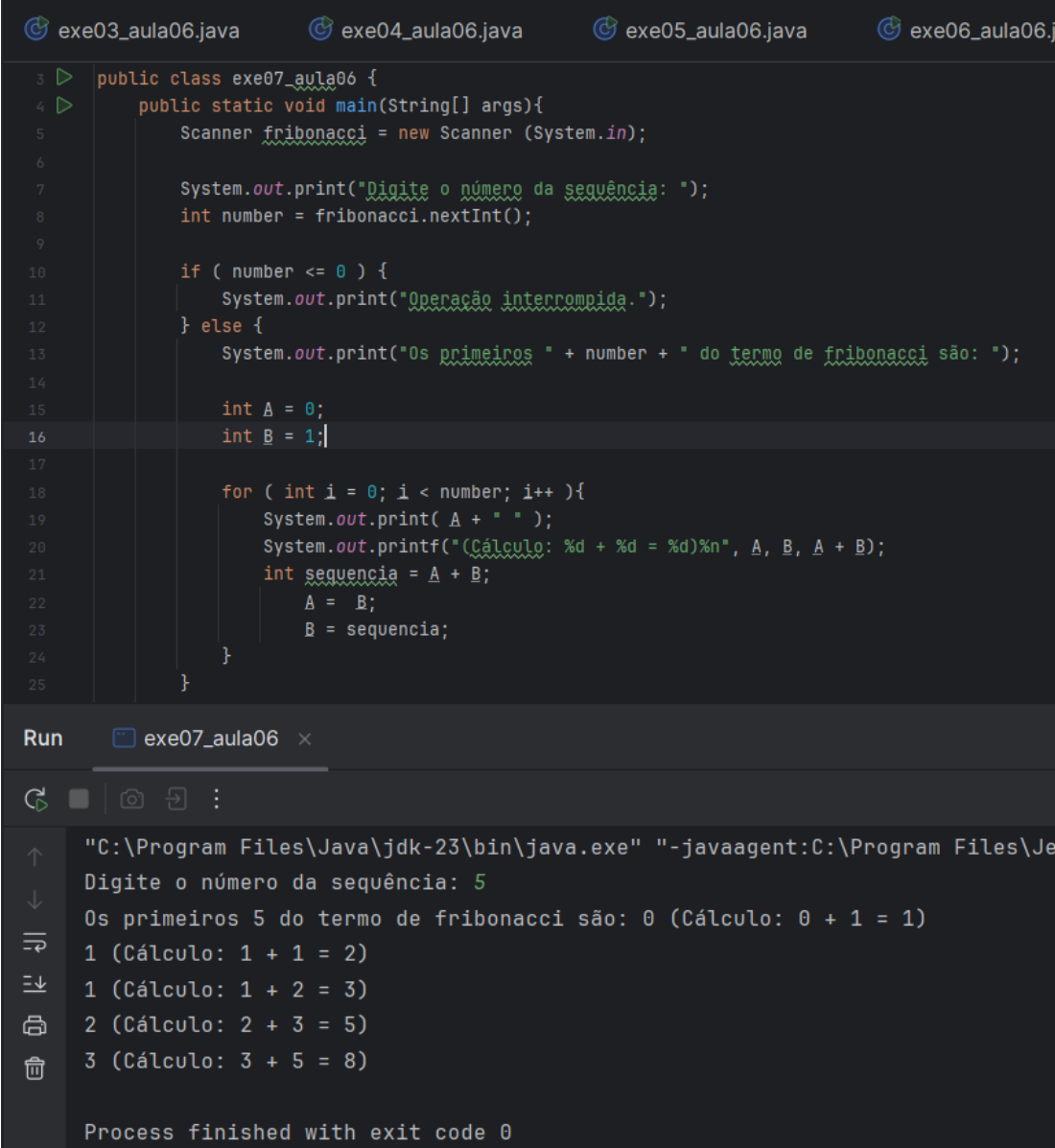
6) Escreva um programa que calcule a soma dos dígitos de um número inteiro, fornecido pelo usuário.



```
exe03_aula06.java  exe04_aula06.java  exe05_aula06.java  exe06_aula06.java x
1      import java.util.Scanner;
2
3      public class exe06_aula06{
4          public static void main(String[] args){
5              Scanner digitos = new Scanner (System.in);
6
7              System.out.print("Digite um número: ");
8              int number = digitos.nextInt();
9              int soma = 0;
10             int valor_Soma = number;
11
12             while ( number != 0 ) {
13                 soma += number % 10;
14                 number /= 10;
15             }
16
17             System.out.printf("A soma dos dígitos de %d é de: %d\n", valor_Soma, soma);
18
19             digitos.close();
20         }
21     }
22
23 Run  exe06_aula06 x
24
25 "C:\Program Files\Java\jdk-23\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\Intell
26 Digite um número: 350
27 A soma dos dígitos de 350 é de: 8
28
29 Process finished with exit code 0
```

Fonte: Autoria Própria

7) Faça um programa que imprima os primeiros valores “N” dos termos de sequência de Fibonacci.



```
public class exe07_aula06 {
    public static void main(String[] args){
        Scanner fribonacci = new Scanner (System.in);

        System.out.print("Digite o número da sequência: ");
        int number = fribonacci.nextInt();

        if ( number <= 0 ) {
            System.out.print("Operação interrompida.");
        } else {
            System.out.print("Os primeiros " + number + " do termo de fribonacci são: ");

            int A = 0;
            int B = 1;

            for ( int i = 0; i < number; i++ ){
                System.out.print( A + " " );
                System.out.printf("(Cálculo: %d + %d = %d)%n", A, B, A + B);
                int sequencia = A + B;
                A = B;
                B = sequencia;
            }
        }
    }
}
```

Run exe07_aula06 x

"C:\Program Files\Java\jdk-23\bin\java.exe" "-javaagent:C:\Program Files\Je
Digite o número da sequência: 5
Os primeiros 5 do termo de fribonacci são: 0 (Cálculo: 0 + 1 = 1)
1 (Cálculo: 1 + 1 = 2)
1 (Cálculo: 1 + 2 = 3)
2 (Cálculo: 2 + 3 = 5)
3 (Cálculo: 3 + 5 = 8)
Process finished with exit code 0

Fonte: Autoria Própria