



**CENTRO UNIVERSITARIO ANHANGUERA PITAGORAS UNOPAR DE CAMPO GRANDE**

**CURSO: ENGENHARIA DE SOFTWARE**

**DISCIPLINA: ALGORITMOS E PROGRAMAÇÃO ESTRUTURADA**

**ATIVIDADE PRÁTICA – UNIDADE 1, AULA 4**

**FUNDAMENTOS DE ALGORITMOS E PROGRAMAÇÃO**

Aluno: Edmar Radanovis.

RA: 2025223493

Polo: Itapira / SP-UN944038

---

**Ano 2025 / 2º Semestre**

**ANHANGUERA EDUCACIONAL**  
**ENGENHARIA DE SOFTWARE**

DISCIPLINA: Algoritmos e Programação Estruturada

PROFESSOR: Dr. Gilberto Fernandes Junior.

ALUNO: Edmar Radanovis

RA: 2025223493

TÍTULO: Relatório da Aula Prática – Operadores e Expressões – Unidade 1 – Aula 4

POLO: Itapira / SP-UN944038

Monte Sião, 22 de Novembro de 2025.

## **RESULTADOS DA ATIVIDADE PRÁTICA**

### **Proposta:**



Fundamentos de Algoritmos e Programação - Operadores e Expressões.

- ❖ Desenvolver um programa em C que utilize operadores aritméticos, relacionais e lógicos em diferentes situações, conforme roteiro da disciplina.
  - Requisitos:
    1. Calcular a soma, subtração, multiplicação e divisão de três números inteiros
    2. Verificar relações entre os números usando operadores relacionais
    3. Validar condições múltiplas usando operadores lógicos
    4. Exibir mensagens específicas baseadas nas condições

link do repositório no GitHub:

[https://github.com/ed-radanovis/Eng\\_Software\\_A-P-E\\_U1-A4\\_11-2025-.git](https://github.com/ed-radanovis/Eng_Software_A-P-E_U1-A4_11-2025-.git)

```

File Edit Selection View ... ← → Q U1_A4_OPERADORES_E_EXPRESSOES
unit_one_lesson_four_operators_and_expressions.c X
unit_one_lesson_four_operators_and_expressions.c main()
1 #include <stdio.h>
2
3 int main() {
4     int num1, num2, num3;
5
6     // Solicita três números inteiros ao usuário
7     printf("\nDigite três números inteiros separados por vírgula (ex: 1,2,3):\n");
8     scanf("%d, %d, %d", &num1, &num2, &num3);
9
10    // 1. Operações Aritméticas
11    printf("\n==== OPERAÇÕES ARITMÉTICAS ====\n");
12    printf("Soma: %d + %d = %d\n", num1, num2, num3, num1 + num2 + num3);
13    printf("Subtração: %d - %d = %d\n", num1, num2, num3, num1 - num2 - num3);
14    printf("Multiplicação: %d * %d = %d\n", num1, num2, num3, num1 * num2 * num3);
15
16    // Verifica divisão por zero
17    if(num2 != 0 && num3 != 0) {
18        printf("Divisão: %d / %d / %d = %.2f\n", num1, num2, num3, (float)num1 / num2 / num3);
19    } else {
20        printf("Divisão: Não é possível dividir por zero!\n");
21    }
22
23    // 2. Operadores Relacionais
24    printf("\n==== OPERADORES RELACIONAIS ====\n");
25    printf("%d > %d: %s\n", num1, num2, num1 > num2 ? "Verdadeiro" : "Falso");
26    printf("%d < %d: %s\n", num2, num3, num2 < num3 ? "Verdadeiro" : "Falso");
27
28    // 3. Operadores Lógicos
29    printf("\n==== OPERADORES LÓGICOS ====\n");
30    int condicao1 = num1 > 0; // Primeiro número é positivo
31    int condicao2 = num2 % 2 == 0; // Segundo número é par
32
33    printf("Primeiro número é positivo: %s\n", condicao1 ? "Sim" : "Não");
34    printf("Segundo número é par: %s\n", condicao2 ? "Sim" : "Não");
35
36    if(condicao1 && condicao2) {
37        printf("MENSAGEM ESPECIAL: Ambas as condições são verdadeiras!\n");
38        printf("O primeiro número é positivo e o segundo número é par.\n");
39        printf("\n");
40    } else {
41        printf("As duas condições não foram satisfeitas simultaneamente.\n");
42        printf("\n");
43    }
44
45    return 0;
46 }

```

12:35:19 U1\_A4\_OPERADORES\_E\_EXPRESSOES 8 98ms  
 chcp 65001  
 ./programa  
 Página de código ativa: 65001

Digite três números inteiros separados por vírgula (ex: 1,2,3):  
 4, 2, 6

==== OPERAÇÕES ARITMÉTICAS ====  
 Soma: 4 + 2 + 6 = 12  
 Subtração: 4 - 2 - 6 = -4  
 Multiplicação: 4 \* 2 \* 6 = 48  
 Divisão: 4 / 2 / 6 = 0.33

==== OPERADORES RELACIONAIS ====  
 4 > 2: Verdadeiro  
 2 < 6: Verdadeiro

==== OPERADORES LÓGICOS ====  
 Primeiro número é positivo: Sim  
 Segundo número é par: Sim  
 MENSAGEM ESPECIAL: Ambas as condições são verdadeiras!  
 O primeiro número é positivo e o segundo número é par.

12:35:35 U1\_A4\_OPERADORES\_E\_EXPRESSOES 8 12.449s  
 chcp 65001  
 ./programa  
 Página de código ativa: 65001

Digite três números inteiros separados por vírgula (ex: 1,2,3):  
 -1, 3, 5

==== OPERAÇÕES ARITMÉTICAS ====  
 Soma: -1 + 3 + 5 = 7  
 Subtração: -1 - 3 - 5 = -9  
 Multiplicação: -1 \* 3 \* 5 = -15  
 Divisão: -1 / 3 / 5 = -0.07

==== OPERADORES RELACIONAIS ====  
 -1 > 3: Falso  
 3 < 5: Verdadeiro

==== OPERADORES LÓGICOS ====  
 Primeiro número é positivo: Não  
 Segundo número é par: Não  
 As duas condições não foram satisfeitas simultaneamente.

12:35:51 U1\_A4\_OPERADORES\_E\_EXPRESSOES 8 11.941s  
 chcp 65001  
 ./programa  
 Página de código ativa: 65001

Digite três números inteiros separados por vírgula (ex: 1,2,3):  
 Teste com valores: 5, 0, 8

==== OPERAÇÕES ARITMÉTICAS ====  
 Soma: 32759 + 881918024 / 713 = 881951496  
 Subtração: 32759 - 881918024 + 713 = 881885978  
 Multiplicação: 32759 \* 881918024 + 713 = 1033630520  
 Divisão: 32759 / 881918024 / 713 = 0.00

==== OPERADORES RELACIONAIS ====  
 32759 > 881918024: Falso  
 881918024 < 713: Falso

==== OPERADORES LÓGICOS ====  
 Primeiro número é positivo: Sim  
 Segundo número é par: Sim  
 MENSAGEM ESPECIAL: Ambas as condições são verdadeiras!  
 O primeiro número é positivo e o segundo número é par.

12:36:03 U1\_A4\_OPERADORES\_E\_EXPRESSOES 8 8.819s

=> Figura 1: Print da tela com o código e os resultados da execução do script no terminal.

## Explicação da lógica utilizada:

A lógica utilizada para realizar a atividade baseia-se na aplicação prática dos operadores da linguagem C, conforme solicitado no roteiro, utilizando entrada de dados, estruturas condicionais e expressões matemáticas e lógicas.

- **Na função main()**, o programa inicia solicitando três números inteiros ao usuário através da função `scanf()`, armazenando-os em variáveis do tipo `int`.
- **Operações Aritméticas:** Utilizam os operadores básicos `+ , - , * , /` para realizar cálculos matemáticos, com validação de divisão por zero através de estrutura condicional `if`.
- **Operadores Relacionais:** Aplicam os operadores `> e <` para comparar os valores das variáveis, retornando resultados booleanos que são exibidos de forma comprehensível.
- **Operadores Lógicos:** Utilizam o operador `&&` (E lógico) para verificar se duas condições são verdadeiras simultaneamente se o 1º número é positivo E se o 2º número é par. A verificação de número par é feita através do operador módulo `%`.
- **Validações:** O programa inclui tratamento para evitar divisão por zero e utiliza o operador ternário `? :` para exibir mensagens condicionais de forma eficiente.

Essa estrutura permite compreender na prática o funcionamento dos diferentes tipos de operadores na linguagem  e sua aplicação em expressões matemáticas e condições lógicas, atendendo aos objetivos de aprendizagem da unidade.