



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

**CURSO: ENGENHARIA DE SOFTWARE**

**DISCIPLINA: ALGORITMOS E PROGRAMAÇÃO ESTRUTURADA**

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

**FUNÇÕES E PASSAGEM DE PARÂMETROS**

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 – Funções e Passagem de Parâmetros – Unidade 4  
– Aula 1

POLO: Itapira / SP-UN944038

Monte Sião, 22 de Novembro de 2025.

## RESULTADOS DA ATIVIDADE PRÁTICA

### Proposta:



#### Funções e Passagem de Parâmetros - Funções.

- ❖ Desenvolver um programa em **C** que calcule o salário de um profissional através de funções modularizadas, conforme roteiro da disciplina.

- Requisitos:

1. Solicitar valor da hora trabalhada e quantidade de horas trabalhadas no mês
2. Calcular salário bruto utilizando função específica
3. Calcular desconto de 9% sobre o salário bruto utilizando função específica
4. Calcular salário líquido utilizando função específica
5. Exibir salário bruto, desconto e salário líquido ao usuário

link do repositório no GitHub:

[https://github.com/ed-radanovis/Eng\\_Software\\_A-P-E\\_U4-A1\\_11-2025.git](https://github.com/ed-radanovis/Eng_Software_A-P-E_U4-A1_11-2025.git)

The image shows a code editor window with a C program and a terminal window below it. The C program, named `unit_four_lesson_one_functions_and_parameter_transfer.c`, defines three functions: `calcular_salario_bruto`, `calcular_desconto`, and `calcular_salario_liquido`. The `main` function prompts the user for the hourly wage and the number of hours worked, then calls the three functions to calculate the gross salary, discount, and net salary. The program uses `printf` and `scanf` for input and output.

```
1 #include <stdio.h>
2
3 // Funcao para calcular o salario bruto
4 float calcular_salario_bruto(float valor_hora, int horas_trabalhadas) {
5     return valor_hora * horas_trabalhadas;
6 }
7
8 // Funcao para calcular o desconto de 9%
9 float calcular_desconto(float salario_bruto) {
10     return salario_bruto * 0.09;
11 }
12
13 // Funcao para calcular o salario liquido
14 float calcular_salario_liquido(float salario_bruto, float desconto) {
15     return salario_bruto - desconto;
16 }
17
18 int main() {
19     float valor_hora, salario_bruto, desconto, salario_liquido;
20     int horas_trabalhadas;
21
22     printf("\n");
23     printf("===  SISTEMA DE CÁLCULO DE SALÁRIO  ===\n\n");
24
25     // Solicita os dados do usuario
26     printf("$ Digite o valor da hora trabalhada: R$ ");
27     scanf("%f", &valor_hora);
28
29     printf("● Digite a quantidade de horas trabalhadas no mês: ");
30     scanf("%d", &horas_trabalhadas);
31
32     // Chama as funcoes para realizar os calculos
33     salario_bruto = calcular_salario_bruto(valor_hora, horas_trabalhadas);
34     desconto = calcular_desconto(salario_bruto);
35     salario_liquido = calcular_salario_liquido(salario_bruto, desconto);
36
37     // Exibe os resultados
38     printf("\n=== RESULTADO DOS CÁLCULOS ===\n");
39     printf("Salário bruto: R$ %.2f\n", salario_bruto);
40     printf("Desconto (9%): R$ %.2f\n", desconto);
41     printf("Salário liquido: R$ %.2f\n", salario_liquido);
42     printf("\n✅ Programa encerrado.\n");
43     printf("\n");
44
45     return 0;
46 }
```

The terminal window shows the execution of the program. It prompts the user for the hourly wage and the number of hours worked, then displays the calculated gross salary, discount, and net salary. The program is run three times with different inputs.

```
20:12:12 U4_A1_FUNCOES_E_PASSAGEM_DE_PARAMETROS 8 92ms
➔ chcp 65001
> ./programa
Página de código ativa: 65001

===  SISTEMA DE CÁLCULO DE SALÁRIO  ===

$ Digite o valor da hora trabalhada: R$ 12
● Digite a quantidade de horas trabalhadas no mês: 200

=== RESULTADO DOS CÁLCULOS ===
Salário bruto: R$ 2400.00
Desconto (9%): R$ 216.00
Salário liquido: R$ 2184.00

✅ Programa encerrado.

20:13:20 U4_A1_FUNCOES_E_PASSAGEM_DE_PARAMETROS 8 10.133s
➔ chcp 65001
> ./programa
Página de código ativa: 65001

===  SISTEMA DE CÁLCULO DE SALÁRIO  ===

$ Digite o valor da hora trabalhada: R$ 85
● Digite a quantidade de horas trabalhadas no mês: 160

=== RESULTADO DOS CÁLCULOS ===
Salário bruto: R$ 13600.00
Desconto (9%): R$ 1224.00
Salário liquido: R$ 12376.00

✅ Programa encerrado.

20:13:33 U4_A1_FUNCOES_E_PASSAGEM_DE_PARAMETROS 8 7.294s
➔ chcp 65001
> ./programa
Página de código ativa: 65001

===  SISTEMA DE CÁLCULO DE SALÁRIO  ===

$ Digite o valor da hora trabalhada: R$ 120
● Digite a quantidade de horas trabalhadas no mês: 80

=== RESULTADO DOS CÁLCULOS ===
Salário bruto: R$ 9600.00
Desconto (9%): R$ 864.00
Salário liquido: R$ 8736.00

✅ Programa encerrado.
```

=> 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 de funções na linguagem **C**, conforme solicitado no roteiro, com foco na modularização de cálculos financeiros através da *divisão do programa* em **funções** especializadas.

- **Função `calcular_salario_bruto()`** : Recebe dois parâmetros - `valor_hora` (float) e `horas_trabalhadas` (int) - e retorna o produto entre esses valores, representando o salário bruto. Esta função encapsula a lógica básica de multiplicação, permitindo reutilização e melhor organização do código.
- **Função `calcular_desconto()`** : Recebe o `salario_bruto` como parâmetro e aplica a taxa de **9% (0,09)** sobre este valor, retornando o montante do desconto. A função isola o cálculo percentual, facilitando possíveis ajustes futuros na taxa de desconto.
- **Função `calcular_salario_liquido()`** : Recebe dois parâmetros - `salario_bruto` e `desconto` - e retorna a diferença entre eles, representando o valor líquido a ser recebido pelo funcionário. Esta função implementa a lógica final do cálculo salarial.
- **Programa Principal:** A função `main()` atua como *orquestradora* do fluxo do programa, sendo responsável pela interação com o usuário (entrada de dados), chamada sequencial das funções especializadas e exibição dos resultados formatados.
- **Modularização e Organização:** A divisão em funções específicas segue os princípios de programação modular, onde cada função tem uma responsabilidade única e bem definida, facilitando a manutenção, teste e compreensão do código.
- **Contextualização Prática:** O cenário de cálculo de salário para profissionais que trabalham por hora fornece um contexto real para a aplicação das funções, demonstrando a utilidade prática da modularização em sistemas de folha de pagamento.

Esta abordagem permite compreender na prática o funcionamento das funções na linguagem



, incluindo declaração, parâmetros, retorno de valores e chamada de funções, atendendo integralmente aos objetivos de aprendizagem da unidade sobre modularização e organização de código.