



CENTRO UNIVERSITARIO ANHANGUERA PITAGORAS UNOPAR DE CAMPO GRANDE

CURSO: ENGENHARIA DE SOFTWARE

DISCIPLINA: ALGORITMOS E PROGRAMAÇÃO ESTRUTURADA

ATIVIDADE PRÁTICA – UNIDADE 3, AULA 1

VETORES, MATRIZES, STRUCTS E PONTEIROS

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 – Vetores, Matrizes, Structs e Ponteiros – Unidade 3
– Aula 1

POLO: Itapira / SP-UN944038

Monte Sião, 22 de Novembro de 2025.

RESULTADOS DA ATIVIDADE PRÁTICA

Proposta:



Vetores, Matrizes, Structs e Ponteiros - Vetores.

- ❖ Desenvolver um programa em **C** que declare e manipule um vetor de inteiros para armazenar vendas diárias, conforme roteiro da disciplina.

➤ Requisitos:

1. Declarar um vetor de tamanho 5 para números inteiros
2. Solicitar ao usuário que insira 5 valores inteiros
3. Armazenar os valores no vetor
4. Calcular a soma de todos os valores do vetor
5. Exibir todos os elementos do vetor, um por linha
6. Exibir a soma total dos valores

link do repositório no GitHub:

https://github.com/ed-radanovis/Eng_Software_A-P-E_U3-A1_11-2025.git

```

File Edit Selection View ... ↶ ↷ 🔍 U3_A1_VETORES_MATRIZES_STRUCTS_PONTEIROS
C unit_three_lesson_one_vectors_matrices_structs_pointers.c x
C unit_three_lesson_one_vectors_matrices_structs_pointers.c > main()
1 #include <stdio.h>
2
3 int main() {
4     // Declaração do vetor para armazenar 5 números inteiros
5     int vendas[5];
6     int soma = 0;
7
8     printf("\n");
9     printf("■ SISTEMA DE CONTROLE DE VENDAS DIÁRIAS ■\n");
10    printf("Informe abaixo as vendas realizadas dia a dia no período de 5 dias:\n\n");
11
12    // Leitura dos valores fornecidos pelo usuário
13    for(int i = 0; i < 5; i++) {
14        printf("■ Dígite as vendas do dia %d: ", i + 1);
15        scanf("%d", &vendas[i]);
16    }
17
18    // Cálculo da soma dos elementos do vetor
19    for(int i = 0; i < 5; i++) {
20        soma += vendas[i];
21    }
22
23    // Exibição dos elementos do vetor e soma total
24    printf("\n■ RELATÓRIO DE VENDAS ■\n");
25    printf("Vendas por dia:\n");
26
27    for(int i = 0; i < 5; i++) {
28        printf("■ Dia %d: %d vendas\n", i + 1, vendas[i]);
29    }
30
31    printf("\n■ TOTAL GERAL ■\n");
32    printf("Soma total de vendas: %d\n", soma);
33    printf("■ Programa encerrado.\n");
34    printf("\n");
35
36    return 0;
37 }

18:50:54 U3_A1_VETORES_MATRIZES_STRUCTS_PONTEIROS 8 168ms
● chcp 65001
> ./programa
Página de código ativa: 65001
■ SISTEMA DE CONTROLE DE VENDAS DIÁRIAS ■
Informe abaixo as vendas realizadas dia a dia no período de 5 dias:
■ Dígite as vendas do dia 1: 10
■ Dígite as vendas do dia 2: 15
■ Dígite as vendas do dia 3: 12
■ Dígite as vendas do dia 4: 8
■ Dígite as vendas do dia 5: 20
■ RELATÓRIO DE VENDAS ■
Vendas por dia:
Dia 1: 10 vendas
Dia 2: 15 vendas
Dia 3: 12 vendas
Dia 4: 8 vendas
Dia 5: 20 vendas
■ TOTAL GERAL ■
Soma total de vendas: 65
■ Programa encerrado.

18:51:31 U3_A1_VETORES_MATRIZES_STRUCTS_PONTEIROS 8 28.46s
● chcp 65001
> ./programa
Página de código ativa: 65001
■ SISTEMA DE CONTROLE DE VENDAS DIÁRIAS ■
Informe abaixo as vendas realizadas dia a dia no período de 5 dias:
■ Dígite as vendas do dia 1: 5
■ Dígite as vendas do dia 2: 10
■ Dígite as vendas do dia 3: 5
■ Dígite as vendas do dia 4: 10
■ Dígite as vendas do dia 5: 5
■ RELATÓRIO DE VENDAS ■
Vendas por dia:
Dia 1: 5 vendas
Dia 2: 10 vendas
Dia 3: 5 vendas
Dia 4: 10 vendas
Dia 5: 5 vendas
■ TOTAL GERAL ■
Soma total de vendas: 35
■ Programa encerrado.

18:52:05 U3_A1_VETORES_MATRIZES_STRUCTS_PONTEIROS 8 7.815s
● chcp 65001
> ./programa
Página de código ativa: 65001
■ SISTEMA DE CONTROLE DE VENDAS DIÁRIAS ■
Informe abaixo as vendas realizadas dia a dia no período de 5 dias:
■ Dígite as vendas do dia 1: -3
■ Dígite as vendas do dia 2: 3
■ Dígite as vendas do dia 3: -2
■ Dígite as vendas do dia 4: 2
■ Dígite as vendas do dia 5: 0
■ RELATÓRIO DE VENDAS ■
Vendas por dia:
Dia 1: -3 vendas
Dia 2: 3 vendas
Dia 3: -2 vendas
Dia 4: 2 vendas
Dia 5: 0 vendas
■ TOTAL GERAL ■
Soma total de vendas: 0
■ 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 vetores na linguagem **C**, conforme solicitado no roteiro, com foco no armazenamento, processamento e exibição de dados em **arrays**.

- **Declaração do Vetor:** O programa inicia declarando um vetor `vendas[5]` capaz de armazenar **5 valores inteiros**, conforme exigido na atividade. Esta declaração aloca espaço na memória para armazenar as vendas de cada dia.
- **Leitura dos Dados:** Utiliza uma estrutura de repetição `for` para solicitar ao usuário os valores de vendas para cada um dos 5 dias. O loop percorre os índices de **0 a 4**, preenchendo cada posição do vetor com os valores fornecidos pelo usuário através da função `scanf()`.
- **Processamento dos Dados:** Um segundo loop `for` é utilizado para calcular a `soma` total de todos os elementos do vetor. A variável `soma` é incrementada com o valor de cada posição do vetor, demonstrando o acesso sequencial aos elementos armazenados.
- **Exibição dos Resultados:** O programa exibe os resultados em formato organizado, mostrando primeiro cada elemento do vetor individualmente (um por linha, conforme solicitado) e depois o resultado da soma total. A interface é projetada para ser clara e informativa, simulando um sistema real de controle de vendas.
- **Contextualização Prática:** O cenário de vendas diárias fornece um contexto real para a aplicação dos vetores, facilitando a compreensão da utilidade prática desta estrutura de dados no armazenamento e processamento de informações sequenciais.

Esta abordagem permite compreender na prática o funcionamento dos vetores na linguagem  , incluindo declaração, inicialização, acesso a elementos e operações matemáticas sobre arrays, atendendo integralmente aos objetivos de aprendizagem da unidade sobre estruturas de dados homogêneas.