

## Sistema de Análise de Vendas - Cinema DNL

Sistema completo para análise de vendas de ingressos desenvolvido em C

### \* Descrição:

\* Este programa oferece um sistema completo para análise de vendas de ingressos de cinema. Permite o cadastro de filmes e respectivas vendas diárias, calcula estatísticas detalhadas e apresenta visualização gráfica dos dados através de histograma ASCII.

=====

=====

### \* Funcionalidades:

- \* - Cadastro de filmes com validação de entrada
- \* - Cálculo de estatísticas (total, média, máximo, mínimo)
- \* - Identificação de filmes mais e menos assistidos
- \* - Visualização gráfica em formato de barras ASCII
- \* - Interface amigável com feedback detalhado

Logo abaixo vou apresentar prints do código fonte completo

Depois vou mostrar o resultado no console com prints !

O CÓDIGO ESTÁ BEM COMENTADO PARA TER UMA BOA COMPREENÇÃO DO ESTÁ SENDO FEITO.

**Antes de começar o código vou deixar no meu repositório do GITHUB para poder ser executado Via terminal do console da IDE.**

**Link do repositório**

**[https://github.com/DaniloAgui/Sistema\\_de\\_Analise\\_Vendas-CinemaDNL](https://github.com/DaniloAgui/Sistema_de_Analise_Vendas-CinemaDNL)**

# Prints do Código fonte em C

## Início do Código - Constantes / estrutura de dados

```
Arquivo  Editar  Seleção  Ver  Acessar  Executar  Terminal  Ajuda  ←  →

C cinema_vendas.c ×
C cinema_vendas.c > ...

1
2  #include <stdio.h>
3  #include <string.h>
4  #include <stdbool.h>
5  #include <ctype.h>
6  #include <stdlib.h>
7
8  // =====
9  //                               CONSTANTES
10 // =====
11 #define DNL_FILMES      100    // Limite máximo de filmes
12 #define DNL_NOME        100    // Tamanho máximo do nome do filme
13 #define BARRA_DNL       60     // Largura máxima das barras no gráfico
14 #define NOME_DISPLAY    25     // Largura para exibição dos nomes
15 #define DIAS_PERIODO    7      // Período de análise em dias
16
17 // =====
18 //                               ESTRUTURA DE DADOS
19 // =====
20 typedef struct {
21     char nome[DNL_NOME];
22     int ingressosPorDia[DIAS_PERIODO]; // Vendas diárias durante 7 dias
23     int totalIngressos;                // Total acumulado no período
24 } Filme;
25
26 typedef struct {
27     long long total;
28     double media;
29     int maximo;
30     int minimo;
31 } Estatisticas;
32
33 // =====
34 //                               PROTÓTIPOS DAS FUNÇÕES
35 // =====
```

# Prints do Código fonte em C

## PROTÓTIPOS DAS FUNÇÕES / FUNÇÃO PRINCIPAL

```
Arquivo  Editar  Seleção  Ver  Acessar  Executar  Terminal  Ajuda  ← →  🔍 ci

C cinema_vendas.c X
C cinema_vendas.c > ...

33 // =====
34 //             PROTÓTIPOS DAS FUNÇÕES
35 // =====
36 void exibirCabecalho(void);
37 void exibirSeparador(char caractere, int tamanho);
38 void limparBuffer(void);
39 int lerQuantidadeFilmes(void);
40 void cadastrarFilme(Filme *filme, int numero);
41 void coletarVendasDiarias(Filme *filme);
42 void calcularTotaisFilmes(Filme filmes[], int quantidade);
43 void calcularEstatisticas(Filme filmes[], int quantidade, Estatisticas *stats);
44 void exibirResumo(Filme filmes[], int quantidade, Estatisticas stats);
45 void exibirFilmesPorCategoria(Filme filmes[], int quantidade, int valor, const char *categoria);
46 void exibirGrafico(Filme filmes[], int quantidade, int maximo);
47 void exibirVendasDiarias(Filme filmes[], int quantidade);
48 void exibirRodape(void);
49 bool validarEntradaNumerica(int *valor);
50 void formatarNome(char *nome);
51
52 // =====
53 //             FUNÇÃO PRINCIPAL
54 // =====
55 int main(void) {
56     Filme filmes[DNL_FILMES];
57     Estatisticas stats;
58     int quantidadeFilmes;
59
60     // Interface inicial
61     exibirCabecalho();
62
63     // Entrada de dados
64     quantidadeFilmes = lerQuantidadeFilmes();
65
66     printf("\n");
67     exibirSeparador('=', 60);
68     printf("          CADASTRO DOS FILMES\n");
69     exibirSeparador('=', 60);
70
71     for (int i = 0; i < quantidadeFilmes; i++) {
72         cadastrarFilme(&filmes[i], i + 1);
73     }
74
75     // Coleta de vendas diárias por 7 dias
76     printf("\n");
77     exibirSeparador('=', 70);
78     printf("          PERÍODO DE COLETA (7 DIAS)\n");
79     exibirSeparador('=', 70);
80     printf("\nSimulando período de vendas de 7 dias...\n");
81 }
```

# Prints do Código fonte em C

## CONTINUAÇÃO DO FUNÇÃO PRINCIPAL E INÍCIO DE IMPLEMENTAÇÃO DAS FUNÇÕES

C cinema\_vendas.c X

C cinema\_vendas.c > main(void)

```
55  int main(void) {
81
82      for (int i = 0; i < quantidadeFilmes; i++) {
83          coletarVendasDiarias(&filmes[i]);
84      }
85
86      // Processamento dos totais e estatísticas
87      calcularTotaisFilmes(filmes, quantidadeFilmes);
88      calcularEstatisticas(filmes, quantidadeFilmes, &stats);
89
90      // Saída de resultados
91      exibirVendasDiarias(filmes, quantidadeFilmes);
92      exibirResumo(filmes, quantidadeFilmes, stats);
93      exibirGrafico(filmes, quantidadeFilmes, stats.maximo);
94      exibirRodape();
95
96      return 0;
97 }
98
99 // =====
100 //             IMPLEMENTAÇÃO DAS FUNÇÕES
101 // =====
102
103 /**
104  * Exibe o cabeçalho inicial do programa
105  */
106 void exibirCabeçalho(void) {
107     system("clear || cls"); // Limpa a tela (funciona no Linux e Windows)
108
109     exibirSeparador('=', 70);
110     printf("                SISTEMA DE ANÁLISE DE VENDAS\n");
111     printf("                CINEMA DNL v2.0\n");
112     exibirSeparador('=', 70);
113     printf("\n📊 ANÁLISE DE VENDAS - PERÍODO DE %d DIAS\n", DIAS_PERIODO);
114     printf("Este sistema permite:\n");
115     printf("• Cadastrar filmes para análise de vendas\n");
116     printf("• Coletar vendas diárias por %d dias consecutivos\n", DIAS_PERIODO);
117     printf("• Calcular estatísticas detalhadas do período\n");
118     printf("• Identificar filmes de maior e menor sucesso\n");
119     printf("• Visualizar dados através de gráfico ASCII\n");
120     printf("• Acompanhar evolução das vendas dia a dia\n");
121     printf("\n");
122     exibirSeparador('-', 70);
123 }
124
125 /**
126  * Exibe uma linha separadora
127  */
128 void exibirSeparador(char caractere, int tamanho) {
```

# Prints do Código fonte em C

## CONTINUAÇÃO : IMPLEMENTAÇÃO DE FUNÇÃO

```
/**
 * Exibe uma linha separadora
 */
void exibirSeparador(char caractere, int tamanho) {
    for (int i = 0; i < tamanho; i++) {
        putchar(caractere);
    }
    putchar('\n');
}

/**
 * Limpa o buffer de entrada
 */
void limparBuffer(void) {
    int c;
    while ((c = getchar()) != '\n' && c != EOF) {}
}

/**
 * Lê e valida a quantidade de filmes
 */
int lerQuantidadeFilmes(void) {
    int quantidade;

    do {
        printf("\n Informe a quantidade de filmes (1 a %d): ", DNL_FILMES);

        if (!validarEntradaNumerica(&quantidade)) {
            printf(" Entrada inválida! Digite apenas números.\n");
            continue;
        }

        if (quantidade < 1 || quantidade > DNL_FILMES) {
            printf(" Quantidade deve estar entre 1 e %d filmes.\n", DNL_FILMES);
        } else {
            printf(" Quantidade confirmada: %d filme(s)\n", quantidade);
            break;
        }
    } while (true);

    return quantidade;
}
```

# Prints do Código fonte em C

## IMPLEMENTAÇÃO DE FUNÇÃO

cinema\_vendas.c

cinema\_vendas.c > main(void)

```
167
168 /**
169  * Valida entrada numérica
170  */
171 bool validarEntradaNumerica(int *valor) {
172     if (scanf("%d", valor) != 1) {
173         limparBuffer();
174         return false;
175     }
176     limparBuffer();
177     return true;
178 }
179
180 /**
181  * Cadastra um filme com validação
182  */
183 void cadastrarFilme(Filme *filme, int numero) {
184     printf("\n FILME %d\n", numero);
185     printf("-----\n");
186
187     // Leitura do nome
188     printf("Nome do filme: ");
189     if (!fgets(filme->nome, DNL_NOME, stdin)) {
190         strcpy(filme->nome, "Filme Sem Título");
191     }
192
193     // Remove quebra de linha e formata
194     filme->nome[strcspn(filme->nome, "\r\n")] = '\0';
195     formatarNome(filme->nome);
196
197     // Inicializa as vendas diárias
198     for (int i = 0; i < DIAS_PERIODO; i++) {
199         filme->ingressosPorDia[i] = 0;
200     }
201     filme->totalIngressos = 0;
202
203     printf("✅ Filme \"%s\" cadastrado! Vendas serão coletadas por %d dias.\n",
204           filme->nome, DIAS_PERIODO);
205 }
206
207 /**
208  * Formata o nome do filme (primeira letra maiúscula)
209  */
210 void formatarNome(char *nome) {
211     if (strlen(nome) == 0) {
212         strcpy(nome, "Filme Sem Nome");
213         return;
214     }
215 }
```



# Prints do Código fonte em C

## IMPLEMENTAÇÃO DE FUNÇÃO

C cinema\_vendas.c X

C cinema\_vendas.c > main(void)

```
210 void formatarNome(char *nome) {
211
212     // Primeira letra maiúscula
213     if (islower(nome[0])) {
214         nome[0] = toupper(nome[0]);
215     }
216 }
217
218 /**
219  * Coleta vendas diárias para um filme durante 15 dias
220  */
221 void coletarVendasDiarias(Filme *filme) {
222     printf("\n??? Coletando vendas para: %s\n", filme->nome);
223     printf("_____ \n");
224
225     for (int dia = 0; dia < DIAS_PERIODO; dia++) {
226         int vendas;
227         do {
228             printf("Dia %2d - Ingressos vendidos: ", dia + 1);
229
230             if (!validarEntradaNumerica(&vendas)) {
231                 printf("X Digite apenas números inteiros!\n");
232                 continue;
233             }
234
235             if (vendas < 0) {
236                 printf("X 0 número de ingressos não pode ser negativo!\n");
237             } else {
238                 filme->ingressosPorDia[dia] = vendas;
239                 break;
240             }
241         } while (true);
242     }
243
244     printf("✅ Coleta concluída para \"%s\" \n", filme->nome);
245 }
246
247 /**
248  * Calcula o total de ingressos para cada filme
249  */
250 void calcularTotaisFilmes(Filme filmes[], int quantidade) {
251     for (int i = 0; i < quantidade; i++) {
252         filmes[i].totalIngressos = 0;
253         for (int dia = 0; dia < DIAS_PERIODO; dia++) {
254             filmes[i].totalIngressos += filmes[i].ingressosPorDia[dia];
255         }
256     }
257 }
258
259 }
```



# Prints do Código fonte em C

## IMPLEMENTAÇÃO DE FUNÇÃO

```
/**
 * Calcula todas as estatísticas
 */
void calcularEstatisticas(Filme filmes[], int quantidade, Estatisticas *stats) {
    stats->total = 0;
    stats->maximo = filmes[0].totalIngressos;
    stats->minimo = filmes[0].totalIngressos;

    for (int i = 0; i < quantidade; i++) {
        stats->total += filmes[i].totalIngressos;

        if (filmes[i].totalIngressos > stats->maximo) {
            stats->maximo = filmes[i].totalIngressos;
        }

        if (filmes[i].totalIngressos < stats->minimo) {
            stats->minimo = filmes[i].totalIngressos;
        }
    }

    stats->media = (quantidade > 0) ? (double)stats->total / quantidade : 0.0;
}

/**
 * Exibe o resumo das estatísticas
 */
void exibirResumo(Filme filmes[], int quantidade, Estatisticas stats) {
    printf("\n\n");
    exibirSeparador('=', 70);
    printf("                                RELATÓRIO FINAL\n");
    exibirSeparador('=', 70);

    // Estatísticas gerais
    printf("\n ESTATÍSTICAS GERAIS:\n");
    printf("-----\n");
    printf("Período de análise: %d dias consecutivos\n", DIAS_PERIODO);
    printf("Total de filmes analisados: %d\n", quantidade);
    printf("Total de ingressos vendidos: %lld\n", stats.total);
    printf("Média de ingressos por filme (período): %.2f\n", stats.media);
    printf("Média diária geral: %.2f ingressos/dia\n", (double)stats.total / (DIAS_PERIODO * quantidade));
    printf("Maior número de ingressos (período): %d\n", stats.maximo);
    printf("Menor número de ingressos (período): %d\n", stats.minimo);

    // Filmes de maior sucesso
    printf("\n FILMES DE MAIOR SUCESSO (%d ingressos):\n", stats.maximo);
    exibirFilmesPorCategoria(filmes, quantidade, stats.maximo, "sucesso");
}
```

# Prints do Código fonte em C

## IMPLEMENTAÇÃO DE FUNÇÃO

```
Arquivo  Editar  Seleção  Ver  Acessar  Executar  Terminal  Ajuda  ← →  🔍 cir

C cinema_vendas.c X

C cinema_vendas.c > main(void)
289 void exibirResumo(Filme filmes[], int quantidade, Estatisticas stats) {
309
310     // Filmes de menor audiência
311     printf("\n FILMES DE MENOR AUDIÊNCIA (%d ingressos):\n", stats.minimo);
312     exibirFilmesPorCategoria(filmes, quantidade, stats.minimo, "menor");
313 }
314
315 /**
316  * Exibe filmes por categoria (maior/menor sucesso)
317  */
318 void exibirFilmesPorCategoria(Filme filmes[], int quantidade, int valor, const char *categoria) {
319     int contador = 0;
320
321     for (int i = 0; i < quantidade; i++) {
322         if (filmes[i].totalIngressos == valor) {
323             contador++;
324             printf("  %d. %s\n", contador, filmes[i].nome);
325         }
326     }
327
328     if (contador == 0) {
329         printf("  (Nenhum filme encontrado)\n");
330     }
331 }
332
333 /**
334  * Exibe o gráfico ASCII
335  */
336 void exibirGráfico(Filme filmes[], int quantidade, int maximo) {
337     printf("\n\n");
338     exibirSeparador('=', 70);
339     printf("                                GRÁFICO DE VENDAS\n");
340     exibirSeparador('=', 70);
341
342     if (maximo == 0) {
343         printf("\n Não há dados para exibir (todos os filmes com 0 ingressos)\n");
344         return;
345     }
346
347     printf("\n");
348     for (int i = 0; i < quantidade; i++) {
349         // Calcula o número de barras proporcionalmente
350         int barras = (int)((double)filmes[i].totalIngressos * BARRA_DNL / maximo + 0.5);
351
352         // Exibe o nome formatado
353         printf("%-*.s | ", NOME_DISPLAY, NOME_DISPLAY, filmes[i].nome);
354     }
}
```

# Prints do Código fonte em C

```
355 // Exibe as barras
356 for (int j = 0; j < barras; j++) {
357     printf("█");
358 }
359
360 // Exibe o valor numérico
361 printf(" %d\n", filmes[i].totalIngressos);
362 }
363
364 printf("\n");
365 exhibirSeparador('-', 70);
366 double valorPorBarra = (double)maximo / BARRA_DNL;
367 printf(" Legenda: cada FILME representa aproximadamente %.2f ingressos\n", valorPorBarra);
368 printf(" Escala máxima: %d ingressos\n", maximo);
369 }
370
371 /**
372  * Exibe as vendas diárias de todos os filmes
373  */
374 void exhibirVendasDiarias(Filme filmes[], int quantidade) {
375     printf("\n\n");
376     exhibirSeparador('=', 70);
377     printf("          VENDAS DIÁRIAS (%d DIAS)\n", DIAS_PERIODO);
378     exhibirSeparador('=', 70);
379
380     for (int i = 0; i < quantidade; i++) {
381         printf("\n█ %s\n", filmes[i].nome);
382         exhibirSeparador('-', 50);
383
384         // Exibe vendas por semana para melhor visualização
385         for (int semana = 0; semana < 3; semana++) {
386             int diaInicio = semana * 5;
387             int diaFim = (semana == 2) ? DIAS_PERIODO : (semana + 1) * 5;
388
389             printf("Semana %d: ", semana + 1);
390             for (int dia = diaInicio; dia < diaFim; dia++) {
391                 if (dia < DIAS_PERIODO) {
392                     printf("D%d:%3d ", dia + 1, filmes[i].ingressosPorDia[dia]);
393                 }
394             }
395             printf("\n");
396         }
397
398         printf("█ Total acumulado: %d ingressos\n", filmes[i].totalIngressos);
399     }
```

# Prints do Código fonte em C

## FINAL DO CODIGO

```
399
400     // Calcula média diária
401     double mediaDiaria = (double)filmes[i].totalIngressos / DIAS_PERIODO;
402     printf("📌 Média diária: %.2f ingressos\n", mediaDiaria);
403 }
404 }
405
406 /**
407  * Exibe o rodapé final
408  */
409 void exibirRodape(void) {
410     printf("\n\n");
411     exibirSeparador('=', 70);
412     printf("                                ANÁLISE CONCLUÍDA COM SUCESSO!\n");
413     printf("                                Sistema de informação v2.0 - 2025\n");
414     exibirSeparador('=', 70);
415     printf("\n Obrigado por usar nosso sistema de análise!\n");
416     printf(" Dica: Use estes dados para tomar decisões estratégicas.\n\n");
417 }
418
```

AS IMAGEM A SEGUIR SÃO ALGUNS RESULTADOS DO CODIGO

## RESULTADO DO CODIGO

O RESULTADO A SEGUIR MOSTRA CADA PASSO  
DIGITADO PELO USUARIO

\* Cadastrar a quantidade de filmes entre 1 a 100  
( escolhi 5 para não ficar  
Longo o programa.

Depois de ter cadastrado os filmes o usuario simula  
as vendas de ingressaos digitando para cada dia a  
quantidade de vendas durante 7 dias.

Para cada dia digitei numeros aleatorios !

No final de digitar todos o usuario vai receber um  
relatório de:

Estatísticas gerais

Grafico de vendas e uma mensagem de fechamento.

*Obs: ao cadastrar o nome do filme cada um recebe o  
cadastro confirmado com mensagem claras ao usuario*



# FILMES CADASTRADOS

RESULTADO DO CODIGO

```
336 void exibirGrafico(Filme filmes[], int quantidade, int maximo) {
```

PROBLEMAS SAÍDA CONSOLE DE DEPURAÇÃO **TERMINAL** ENTRADA E SAÍDA

## ▼ TERMINAL

```
=====
                SISTEMA DE ANÁLISE DE VENDAS
                CINEMA DNL v2.0
=====
```

### ANÁLISE DE VENDAS - PERÍODO DE 7 DIAS

Este sistema permite:

- Cadastrar filmes para análise de vendas
- Coletar vendas diárias por 7 dias consecutivos
- Calcular estatísticas detalhadas do período
- Identificar filmes de maior e menor sucesso
- Visualizar dados através de gráfico ASCII
- Acompanhar evolução das vendas dia a dia

```
-----

Informe a quantidade de filmes (1 a 100): 5
Quantidade confirmada: 5 filme(s)
```

### ===== CADASTRO DOS FILMES =====

#### FILME 1

Nome do filme: Titanic

✅ Filme "Titanic" cadastrado! Vendas serão coletadas por 7 dias.

#### FILME 2

Nome do filme: Romeu e Julieta

✅ Filme "Romeu e Julieta" cadastrado! Vendas serão coletadas por 7 dias.

#### FILME 3

Nome do filme: Paixão de Cristo

✅ Filme "Paixão de Cristo" cadastrado! Vendas serão coletadas por 7 dias.

#### FILME 4

Nome do filme: As Branqueas

✅ Filme "As Branqueas" cadastrado! Vendas serão coletadas por 7 dias.

#### FILME 5

Nome do filme: Catostrofe em Chernobyl

✅ Filme "Catostrofe em Chernobyl" cadastrado! Vendas serão coletadas por 7 dias.



# PERÍODO DE VENDAS

[PROBLEMAS](#)[SAÍDA](#)[CONSOLE DE DEPURAÇÃO](#)[TERMINAL](#)[ENTRADA E SAÍDA](#)

## ✓ TERMINAL

### PERÍODO DE COLETA (7 DIAS)

Simulando período de vendas de 7 dias...

🎬 Coletando vendas para: Titanic

---

Dia 1 - Ingressos vendidos: 2500  
Dia 2 - Ingressos vendidos: 1500  
Dia 3 - Ingressos vendidos: 3500  
Dia 4 - Ingressos vendidos: 4500  
Dia 5 - Ingressos vendidos: 6500  
Dia 6 - Ingressos vendidos: 7500  
Dia 7 - Ingressos vendidos: 5252

✓ Coleta concluída para "Titanic"

🎬 Coletando vendas para: Romeu e Julieta

---

Dia 1 - Ingressos vendidos: 2500  
Dia 2 - Ingressos vendidos: 1500  
Dia 3 - Ingressos vendidos: 1500  
Dia 4 - Ingressos vendidos: 1500  
Dia 5 - Ingressos vendidos: 3500  
Dia 6 - Ingressos vendidos: 4502  
Dia 7 - Ingressos vendidos: 8210

✓ Coleta concluída para "Romeu e Julieta"

🎬 Coletando vendas para: Paixão de Cristo

---

Dia 1 - Ingressos vendidos: 5525  
Dia 2 - Ingressos vendidos: 5954  
Dia 3 - Ingressos vendidos: 5410  
Dia 4 - Ingressos vendidos: 220  
Dia 5 - Ingressos vendidos: 4848  
Dia 6 - Ingressos vendidos: 8789  
Dia 7 - Ingressos vendidos: 2130

✓ Coleta concluída para "Paixão de Cristo"

🎬 Coletando vendas para: As Branqueas

---

Dia 1 - Ingressos vendidos: 4500  
Dia 2 - Ingressos vendidos: 5520  
Dia 3 - Ingressos vendidos: 2556  
Dia 4 - Ingressos vendidos: 4448  
Dia 5 - Ingressos vendidos: 9942  
Dia 6 - Ingressos vendidos: 6599  
Dia 7 - Ingressos vendidos: 9899

✓ Coleta concluída para "As Branqueas"

AS VENDAS DO FILME CHERNOBYL ESTA NA PROXIMA PAGINA

# RELATORIO FINAL

PROBLEMAS   SAÍDA   CONSOLE DE DEPURAÇÃO   TERMINAL   ENTRADA E SAÍDA

## ✓ TERMINAL

🚨 Catostrofe em Chernobyl

-----  
Semana 1: D1:1584   D2:4889   D3:4849   D4:8785   D5:4888

Semana 2: D6:6833   D7:2589

Semana 3:

📊 Total acumulado: 34417 ingressos

📈 Média diária: 4916.71 ingressos

=====

### RELATÓRIO FINAL

=====

#### ESTATÍSTICAS GERAIS:

---

Período de análise: 7 dias consecutivos  
Total de filmes analisados: 5  
Total de ingressos vendidos: 165221  
Média de ingressos por filme (período): 33044.20  
Média diária geral: 4720.60 ingressos/dia  
Maior número de ingressos (período): 43464  
Menor número de ingressos (período): 23212

FILMES DE MAIOR SUCESSO (43464 ingressos):

1. As Branqueas

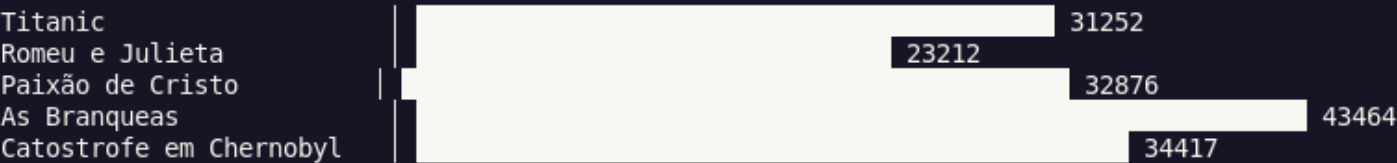
FILMES DE MENOR AUDIÊNCIA (23212 ingressos):

1. Romeu e Julieta

=====

### GRÁFICO DE VENDAS

=====



-----  
Legenda: cada FILME representa aproximadamente 724.40 ingressos  
Escala máxima: 43464 ingressos

=====

ANÁLISE CONCLUÍDA COM SUCESSO!  
Sistema de informação v2.0 - 2025

=====

Obrigado por usar nosso sistema de análise!  
Dica: Use estes dados para tomar decisões estratégicas.

[1] + Done                   "/usr/bin/gdb" --interpreter=mi --tty=\${DbgTerm} 0<"/tmp/Micros  
rwasjd.2ud"

—(kali@kali)-[~/Documentos/faculdade/algorithm e pensamento computacional/cinema\_vendas]  
—\$ █

# VENDAS DIARIAS

PROBLEMAS

SAÍDA

CONSOLE DE DEPURAÇÃO

TERMINAL

ENTRADA E SAÍDA

✓ TERMINAL

✓ Coleta concluída para "As Branqueas"

🎬 Coletando vendas para: Catostrofe em Chernobyl

Dia 1 - Ingressos vendidos: 1584  
Dia 2 - Ingressos vendidos: 4889+  
Dia 3 - Ingressos vendidos: 4849  
Dia 4 - Ingressos vendidos: 8785  
Dia 5 - Ingressos vendidos: 4888  
Dia 6 - Ingressos vendidos: 6833  
Dia 7 - Ingressos vendidos: 2589

✓ Coleta concluída para "Catostrofe em Chernobyl"

## VENDAS DIÁRIAS (7 DIAS)

🎬 Titanic

Semana 1: D1:2500 D2:1500 D3:3500 D4:4500 D5:6500

Semana 2: D6:7500 D7:5252

Semana 3:

📊 Total acumulado: 31252 ingressos

📈 Média diária: 4464.57 ingressos

🎬 Romeu e Julieta

Semana 1: D1:2500 D2:1500 D3:1500 D4:1500 D5:3500

Semana 2: D6:4502 D7:8210

Semana 3:

📊 Total acumulado: 23212 ingressos

📈 Média diária: 3316.00 ingressos

🎬 Paixão de Cristo

Semana 1: D1:5525 D2:5954 D3:5410 D4:220 D5:4848

Semana 2: D6:8789 D7:2130

Semana 3:

📊 Total acumulado: 32876 ingressos

📈 Média diária: 4696.57 ingressos

🎬 As Branqueas

Semana 1: D1:4500 D2:5520 D3:2556 D4:4448 D5:9942

Semana 2: D6:6599 D7:9899

Semana 3:

📊 Total acumulado: 43464 ingressos

📈 Média diária: 6209.14 ingressos

