

## Lista de Exercícios 2

Algoritmo 1) O dono de um cinema que gerenciar a ocupação e lucro de sua única sala de exibição. O cinema é composto por 100 lugares, dispostos como a figura abaixo:

TELA DE EXIBIÇÃO										
	1	2	3	4	5	6	7	8	9	10
A										
B										
C										
D										
E										
F										
G										
H										
I										
J										

Legendas:

	Lugar Popular
	Lugar Comum
	Lugar VIP

Implementar um programa em C que inicialmente peça ao usuário o preço do ingresso para um lugar comum do cinema. Em seguida deve ser apresentado um menu para o usuário, onde sejam possíveis realizar as seguintes tarefas:

a) Vender ou liberar uma cadeira: O usuário deve indicar qual fileira (linha/coluna) da cadeira a ser ocupada, e se o comprador tem direito ou não à meia-entrada. Se o lugar estiver ocupado, essa ação significa, liberar o assento. Caso contrário, significa vender o ingresso

b) Apresentar o resumo de ocupação: O programa deve apresentar a disposição de ocupação da sala, como apresentado na figura abaixo:

	1	2	3	4	5	6	7	8	9	10
A	0	1	0	0	0	0	0	0	0	0
B	0	0	0	0	0	0	0	0	0	0
C	0	0	0	0	0	0	0	0	0	0
D	0	0	1	1	2	2	0	0	0	0
E	0	0	0	0	0	0	0	0	0	0
F	0	0	0	0	0	2	0	0	0	0
G	0	0	0	0	0	0	0	0	0	0
H	0	0	0	2	0	0	0	0	0	0
I	0	0	0	0	0	0	0	0	0	0
J	0	2	0	0	0	0	0	0	0	0

**Ingressos Vendidos: 8 ingressos vendidos**

onde (0) indica uma poltrona desocupada, (1) indica uma poltrona vendida como meia entrada e (2) uma poltrona vendida como inteira.

3) Apresentar o resumo financeiro, indicando quanto foi vendido em ingressos para cada sessão. Para isso, tome como parâmetros que ingressos em lugares populares são 25% mais baratos que ingressos comuns, e ingressos VIP são 30% mais caros que ingressos comuns. Além disso, ocupantes com direito a meia-entrada tem desconto de 50% na compra de seu ingresso. Por exemplo, para um valor de ingresso de R\$ 10,00, a saída de uma sessão com o padrão apresentado no item anterior seria:

**Resumo Financeiro:**

**Preço do Ingresso:** 10 reais (comum), 7.50 (popular), 13.0 (VIP)

**Ingressos Populares:**

0 inteiras, 1 meia. Total: 3.25

**Ingressos VIP:**

2 inteiras, 0 meias: Total: 26.00

**Ingressos Comum:**

3 inteiras, 2 meias: Total: 40.00

**Total Geral: 69.25**

Algoritmo 2) Na turma de programação, o professor precisa organizar a participação dos alunos, para gerar a lista de aprovados e reprovados ao final do curso. Uma possibilidade é um programa que permita organizar esta lista. O programa deve permitir guardar os dados de até 20 alunos de graduação. Estes dados incluem a matrícula, o nome, três notas e o número de faltas de cada aluno. O programa deve oferecer 5 opções para o professor: (1) Inserir os dados de um aluno, (2) Remover os dados de um aluno de acordo com sua matrícula, (3) listar alunos aprovados, (4) listar alunos reprovados por falta e (5) listar alunos reprovados por desempenho.

**Considerações:**

- Casos de exceção como tentativa de se incluir mais de 20 alunos ou remover um aluno cuja a matrícula não exista devem produzir mensagens explicativas ao professor;
- Alunos são considerados aprovados quando não possuem mais de 10 faltas, e a média de suas notas é maior ou igual a 7.0
- Alunos são considerados reprovados por desempenho quando não possuem mais de 10 faltas e a média de suas notas é inferior a 7.0
- Alunos são considerados reprovados por faltas quando possuem mais de 10 faltas na disciplina

Observações e dicas: Utilize apenas vetores de tipos primitivos para guardar os dados dos alunos. Busque utilizar diferentes vetores para guardar as matrículas, o nome e demais dados de cada aluno. No caso das notas, uma sugestão é usar uma matriz 20 x 3 (Em outras palavras, 3 notas de 20 alunos).