

<b>Curso: Jovem programador</b>
<b>Disciplina: Programador de Sistemas</b>
<b>Professor: Ricardo Köhler</b>
<b>Avaliação Final</b>

### **1. Sistema de Controle de Estoque Automatizado**

Desenvolva um sistema que gerencie o estoque de um pequeno armazém. Cada produto deve ser armazenado em um vetor com nome, quantidade e validade. Permita entradas e saídas de produtos por meio de uma **fila**, mantendo o controle de validade para evitar perdas. Crie funções para inserção, remoção e verificação de validade.

### **2. Simulador de Atendimento Hospitalar**

Crie um sistema que simula o atendimento de pacientes em um hospital com **fila de prioridade**. Utilize **estruturas de filas separadas** (urgência e normal), e implemente funções para registrar paciente, chamar o próximo e exibir estatísticas (quantos atendidos, tempo de espera médio etc.).

### **3. Gerador de Horários Escolares**

Crie uma matriz 5x6 representando os dias da semana e horários de aulas. O sistema deve permitir alocar aulas automaticamente respeitando pré-requisitos (por exemplo, uma disciplina não pode estar na 1ª aula da segunda-feira). Use **laços aninhados** e **funções de verificação** para preencher o quadro.

### **4. Simulação de Elevadores**

Simule o funcionamento de três elevadores em um prédio com 20 andares. Use **vetores** para representar os andares de parada solicitados e **funções** para calcular o tempo de espera, o deslocamento e a distribuição mais eficientes dos elevadores. A entrada de andares deve ser dinâmica e contínua (uso de laço while ou do while).

## 5. Editor de Texto com Desfazer/Refazer

Crie um editor simples de texto que simula os comandos "desfazer" e "refazer" usando duas **pilhas**. Cada modificação no texto deve ser registrada. As operações CTRL+Z e CTRL+Y devem manipular essas pilhas. Utilize **funções** para realizar as operações e mostrar o estado atual do texto.

## 6. Jogo de Batalha Naval

Desenvolva uma versão simplificada do jogo Batalha Naval. Use uma **matriz 10x10** para representar o tabuleiro. Implemente funções para posicionar os navios aleatoriamente, realizar jogadas e verificar se houve acerto, erro ou vitória. Use laços para repetição do jogo até o fim.

## 7. Simulador de Ponto de Ônibus Inteligente

Simule um ponto de ônibus onde passageiros entram em uma **fila** e os ônibus têm um limite de capacidade. A cada chegada de um ônibus, passageiros são retirados da fila até a capacidade máxima. Implemente funções para adicionar passageiros, chamar ônibus e calcular tempo médio de espera por passageiro.

## 8. Sistema de Reserva de Assentos em Avião

Crie um sistema de reserva de assentos utilizando uma **matriz 10x6** (10 fileiras, 6 assentos por fileira). Implemente funções para mostrar os assentos livres, reservar assentos (com verificação de disponibilidade) e cancelar reservas. Use **vetores auxiliares** para estatísticas (por ex., fileira mais reservada).

## 9. Gerenciador de Tarefas com Prioridades

Crie um aplicativo de gerenciamento de tarefas que armazene as tarefas em uma **fila de prioridade simulada** com três níveis (alta, média, baixa). Cada tarefa tem descrição, data e prioridade. Implemente funções para adicionar, remover, listar por prioridade e arquivar as concluídas (usando **pilha**).

## 10. Sistema de Monitoramento de Temperaturas

Elabore um sistema para coletar temperaturas em uma matriz 30x24 (30 dias, 24 horas cada). Implemente funções para calcular a média diária, encontrar a maior e menor temperatura do mês e gerar alertas em caso de valores fora do intervalo ideal (ex: > 40°C). Use laços e funções para análise de dados.

