

siga as instruções abaixo. Este código em C simula a alocação de consultas médicas com base na prioridade dos pacientes, disponibilidade de salas e médicos, e gera um relatório das horas trabalhadas pelos médicos.

1. Preparação do ambiente

Antes de executar o código, você deve ter um ambiente de desenvolvimento C configurado. Você pode usar um compilador como o **GCC** ou uma IDE como o **Code::Blocks** ou **DevC++**.

2. Arquivo de dados de entrada (dados.txt)

O código depende de um arquivo de entrada chamado `dados.txt`, que contém informações sobre pacientes, salas e especialidades. Este arquivo deve seguir o formato esperado pela função `lerDados`, que lê o número de pacientes, salas e especialidades, além dos detalhes de cada paciente, médico e especialidade. Um exemplo de como esse arquivo pode ser estruturado:

Exemplo de formato para dados.txt:

```
5          // Número de pacientes
3          // Número de salas
2          // Número de especialidades

Cardiologia // Nome da especialidade
2          // Número de médicos nessa especialidade
101 João   // Médico ID e Nome
102 Maria  // Médico ID e Nome

Pediatria  // Nome da especialidade
1          // Número de médicos nessa especialidade
103 Carlos // Médico ID e Nome

1 João 25 1234567890 70.5 1.75 "Dor no peito" "Medicamento X" 5
2 Maria 30 9876543210 60.0 1.65 "Dor de cabeça" "Medicamento Y" 3
3 José 45 1122334455 80.0 1.80 "Falta de ar" "Medicamento Z" 4
4 Ana 35 2233445566 65.0 1.70 "Febre" "Medicamento A" 2
5 Paula 50 3344556677 75.0 1.68 "Cólica abdominal" "Medicamento B" 1
```

Neste exemplo:

- O primeiro número é o total de pacientes (5).
- O segundo número é o total de salas (3).
- O terceiro número é o total de especialidades (2).
- Para cada especialidade, você tem o nome da especialidade, o número de médicos e a lista de médicos.
- Após as especialidades, são listados os pacientes com seus respectivos dados: ID, nome, idade, telefone, peso, altura, sintomas, medicações e prioridade.

3. Compilação e execução do código

Passo 1: Compilar o código

Abra o terminal ou prompt de comando e navegue até o diretório onde o código-fonte e o arquivo dados.txt estão localizados. Em seguida, execute o comando para compilar o código.

Se estiver usando o **GCC**:

```
gcc -o consulta_medica consulta_medica.c
```

Este comando criará um executável chamado consulta_medica.

Passo 2: Executar o código

Execute o programa gerado:

```
./consulta_medica
```

Certifique-se de que o arquivo dados.txt esteja no mesmo diretório do executável ou forneça o caminho correto para o arquivo.

4. Explicação do Código

- **Estruturas de Dados:**
 - **Paciente:** Armazena informações sobre pacientes, incluindo sintomas e prioridade.
 - **Médico:** Armazena informações sobre médicos, incluindo a especialidade e as horas trabalhadas.
 - **Sala:** Representa as salas de atendimento, com um campo para indicar se a sala está ocupada ou não.

- **Especialidade:** Agrupa médicos por especialidade e armazena o número de médicos para cada especialidade.
- **Funções Principais:**
 - `lerDados`: Lê os dados do arquivo `dados.txt` e os armazena nas estruturas apropriadas.
 - `compararPrioridade`: Função usada pelo `qsort` para ordenar os pacientes pela prioridade.
 - `compararHorasTrabalhadas`: Função usada pelo `qsort` para ordenar os médicos pelas horas trabalhadas.
 - `alocarConsultas`: Atribui consultas aos pacientes com base na prioridade, sala disponível e médico especializado.
 - `gerarRelatorio`: Gera um relatório ordenado dos médicos, mostrando as horas trabalhadas.

5. Saída do Programa

O programa imprimirá no terminal as consultas alocadas para os pacientes, indicando a semana, dia, hora, sala, paciente e médico. Após isso, ele gerará um relatório das horas trabalhadas de cada médico.

Exemplo de saída:

```
Semana 1, Dia 1, Sala 1, Hora 08:00 - Paciente: João, Médico: João
Semana 1, Dia 1, Sala 2, Hora 09:00 - Paciente: Maria, Médico: Maria
Semana 1, Dia 1, Sala 3, Hora 10:00 - Paciente: José, Médico: João
Semana 1, Dia 1, Sala 1, Hora 11:00 - Paciente: Ana, Médico: João
Semana 1, Dia 1, Sala 2, Hora 12:00 - Paciente: Paula, Médico: Maria
```

```
Relatorio de Horas Trabalhadas:
Medico: João, Horas Trabalhadas: 3
Medico: Maria, Horas Trabalhadas: 2
```

6. Observações

- Certifique-se de que o arquivo `dados.txt` está no formato correto para que o programa funcione corretamente.
- O programa usa uma chance de 5% para que um paciente falte à consulta, o que é simulado pela linha `pacientes[i].faltou = rand() % 20 < 1 ? 1 : 0;`.