1. **O Problema do Barbeiro Dorminhoco**

O problema se resume a uma barbearia normal dos dias de hoje, onde se tem uma cadeira em que o barbeiro usa para cortar o cabelo dos clientes e outras cadeiras para os demais que chegarem esperarem a sua vez, caso já tenha alguém sendo atendido. Porém no caso aqui retratado quando não houver ninguém para atendimento o barbeiro que irá ocupar a cadeira destina ao corte para dormir, sendo que de tempo em tempo ele acordará e para verificar se há algum cliente esperando atendimento.

Abstraindo a ideia para a programação, foi trabalhado com duas threads, uma representando o cabeleireiro e outra simulando a chegada de clientes, onde elas compartilham algumas informações entre si, como a fila de espera e se a cadeira de atendimento está ocupada ou não.

O diagrama básico de ação, válido paras as threads, Cabeleireiro e Cliente, poderia ser descrito como na **Imagem 1 e Imagem 2**, respectivamente representadas abaixo.

**O Algoritmo**

O algoritmo desse problema como descrito antes foi criado com duas classes Thread distintas, Cabeleireiro e Cliente, que tem sua execuções exemplificadas pelos seguintes passos:

**- Cabeleireiro:**

1. Verifica se o programa esta executando;
2. Verifica se há algum cliente esperando atendimento nas cadeiras de espera;
3. Se tiver alguém esperando chama o próximo da fila e atende o mesmo;
4. Senão tiver ninguém esperando verifica se já não tem alguém que chegou e ocupou a cadeira para cortar o seu cabelo, se sim atende o mesmo;
5. Caso não tenha ninguém esperando ou já em sua cadeira para atendimento, o cabeleireiro dorme por um tempo;
6. Depois de atender o cliente ou dormir libera a cadeira de uso do cabeleireiro para começar a verificação novamente;

**- Cliente:**

1. Verifica se o programa esta executando;
2. Sorteia um numero entre 0 e 1, se cair 1 “insere” um novo cliente (feito esse sorteio para simular a entrada aleatória de clientes como no mundo real);
3. Se for inserir um novo cliente verifica se a fila de espera esta vazia;
4. Se tiver alguém já esperando na fila ou a cadeira do cabeleireiro estiver ocupada, seja por cliente ou pelo próprio cabeleireiro dormindo, toma o seu lugar, se houver cadeira disponível, na fila de espera, senão “vai embora”;
5. Se estiver vazia a fila de espera e a cadeira do cabeleireiro não estiver ocupada, ocupa a mesma;
6. Depois de simular a chegada de um novo cliente, ficando este na cadeira já para ser atendido, ou na fila de espera, ou até mesmo indo embora, a Thread retoma ao passo 1;

**O Programa explicado em código**

O programa foi feito em Delphi, mais especificemente na versão Xe7, que também possui suporte nativo à threads.