1. **O Problema do Barbeiro Dorminhoco**

O problema se resume a uma barbearia normal dos dias de hoje, onde se tem uma cadeira para sentar-se e cortar o cabelo e outras cadeiras para os demais clientes esperarem a sua vez, caso tenha alguém já sendo atendido. Só que no nosso caso quando não houver ninguém para atendimento o nosso barbeiro ocupa a cadeira destina para o corte e dorme, e de tempo em tempo acorda e verifica se há alguém já esperando.

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 nas **Imagem 1 e Imagem 2**, respectivamente representadas abaixo.

**O Algoritmo**

O algoritmo desse problema foi criado com duas classes Threads distintas, onde cada uma faz determinadas funções, sendo elas Cabeleireiro, que representa as ações praticadas pelo mesmo, e outra classe Thread chamada de Cliente que representa a “criação” ou “chegada” de novos clientes na barbearia, podendo serem representadas nos 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, 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;