Processos e Threads

Barbeiro Dorminhoco

Barbeiro Dorminhoco

Barbeiro Dorminhoco

- Se não há clientes, o barbeiro adormece;
- Se a cadeira do barbeiro estiver livre, um cliente pode ser atendido;
- Um cliente só espera pelo barbeiro se houver uma cadeira de espera vazia.

Clientes só esperam nas cadeiras

- Não pode haver uma "fila na porta", como seria o caso desta implementação:

```
semaforo cadeiras = 5;
wait(cadeiras);
```

Clientes só esperam nas cadeiras

- Não pode haver uma "fila na porta", como seria o caso desta implementação:

```
semaforo cadeiras = 5;
wait(cadeiras);
```

Clientes só esperam nas cadeiras

- Esta abordagem também pode causar "fila na porta":

```
semaforo cadeiras = 5;
if (sem_value(cadeiras) > 0)
  wait(cadeiras);
```

Clientes só esperam nas cadeiras

```
mutex_lock mutex;
int cadeiras = 5;

mutex_lock(mutex);
if (cadeiras > 0)
  cadeiras--;
  mutex_unlock(mutex);
  entra_na_barbearia();
else
   mutex_unlock(mutex);
   desiste_de_cortar_o_cabelo();
```

Clientes só esperam nas cadeiras

```
semaforo cadeiras = 5;

if (trywait(cadeiras) == 0)
  entra_na_barbearia();
else
  desiste_de_cortar_o_cabelo();
```

Disputa pela cadeira do barbeiro

```
semaforo cadeiras = 5;
semaforo cad_barbeiro = 1;

if (trywait(cadeiras) == 0)
   wait(cad_barbeiro);
   signal(cadeiras);
   ...
   signal(cad_barbeiro);
```

- Como saber que o corte acabou?

Disputa pela cadeira do barbeiro

```
semaforo cadeiras = 5;
semaforo cad_barbeiro = 1;
semaforo cabelo_cortado = 0;

if (trywait(cadeiras) == 0)
   wait(cad_barbeiro);
   signal(cadeiras);
   wait(cabelo_cortado);
   signal(cad_barbeiro);
```

Barbeiro

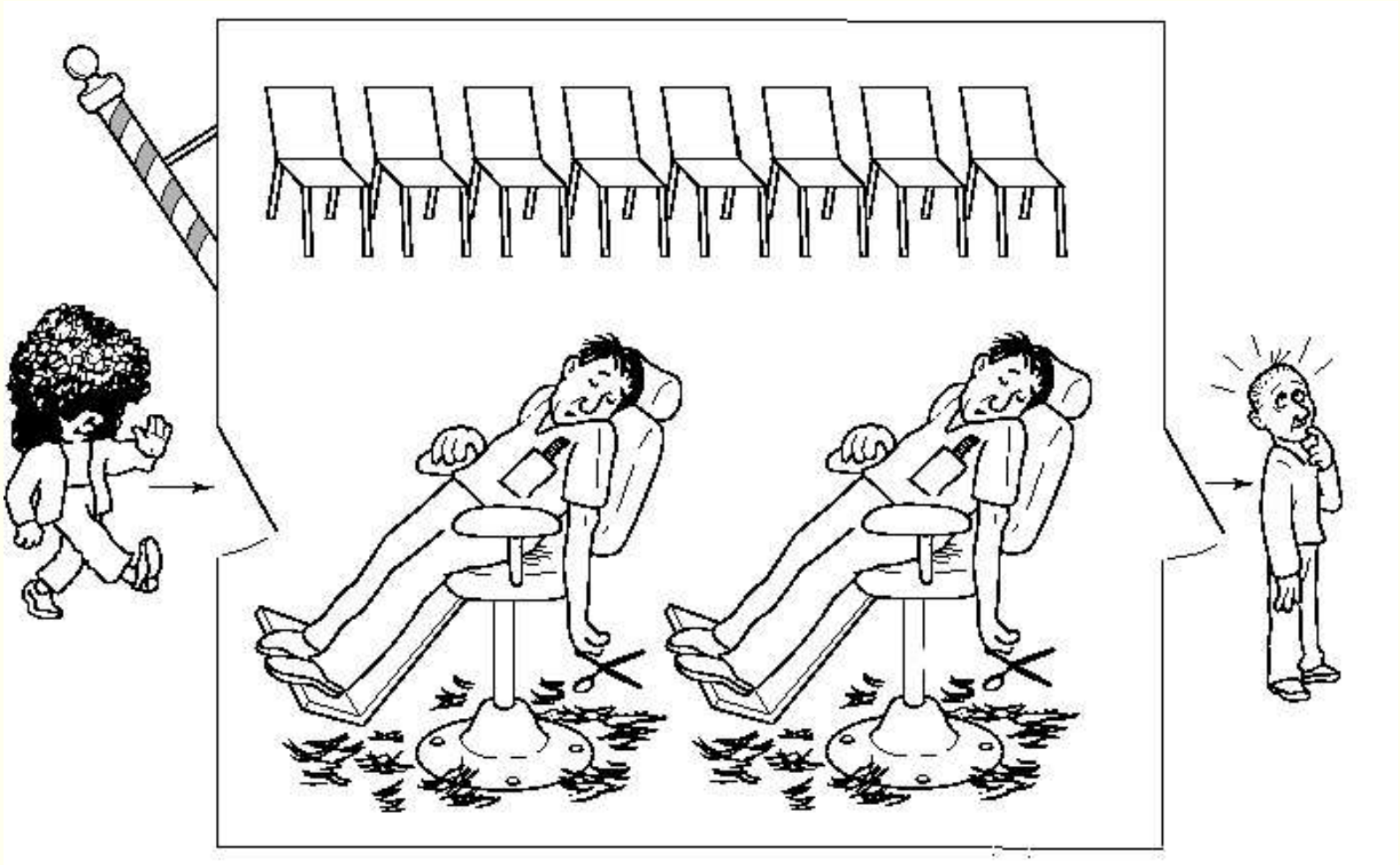
```
semaforo cabelo_cortado = 0;
semaforo cliente_cadeira = 0;

while (true)
  wait(cliente_cadeira); /* ??? */
  corta_cabelo();
  signal(cabelo_cortado);
```

- Como saber que um cliente está na cadeira?

Cliente sinaliza que está na cadeira

```
if (trywait(cadeiras) == 0)
   wait(cad_barbeiro);
   signal(cadeiras);
   signal(cliente_cadeira);
   wait(cabelo_cortado);
   signal(cad_barbeiro);
```

Múltiplos Barbeiros Dorminhocos



Múltiplos Barbeiros Dorminhocos

- Vários semáforos semelhantes ao problema anterior

```
semaforo cadeiras = 8;
semaforo cad_barbeiro[N_BARBEIROS] =
                        {0, 0, 0, ..., 0};
semaforo cabelo_cortado[N_BARBEIROS] =
                        {0, 0, 0, ..., 0};
semaforo cliente_cadeira[N_BARBEIROS] =
                        {0, 0, 0, ..., 0};
```

Múltiplos Barbeiros Dorminhocos

- O cliente precisa saber qual é o identificador do barbeiro disponível.
- Problema análogo a fila única em bancos com visor para chamar os clientes.

```
semaforo escreve_visor = 1;
semaforo le_visor = 0;
int visor;
```

Barbeiro

```
while (true)
  wait(escreve_visor);
  visor = id_barbeiro;
  signal(le_visor);
  wait(cliente_cadeira[id_barbeiro]);
  corta_cabelo();
  signal(cabelo_cortado[id_barbeiro]);
```

Cliente

```
if (trywait(cadeiras) == 0)
   signal(cadeiras);
   wait(le_visor);
   minha_cadeira = visor;
   signal(escreve_visor);
   wait(cad_barbeiro[minha_cadeira]);
   signal(cliente_cadeira[minha_cadeira]);
   wait(cabelo_cortado[minha_cadeira]);
   signal(cad_barbeiro[minha_cadeira]);
```