

**MC514**  
**Sistemas Operacionais:**  
**Teoria e Prática**  
1s2006

**Problema dos Filósofos Famintos II**

# Solução do livro Tanenbaum

```
semaforo lock;  
semaforo filosofo[N] = {0, 0, 0, ..., 0}  
int estado[N] = {T, T, T, ...,T}
```

## Filósofo i:

```
while (true)  
    pensa();  
    pega_garfos();  
    come();  
    solta_garfos();
```

```
testa_garfos(int i)
    if (estado[i] == H && estado[fil_esq] != E &&
        estado[fil_dir] != E)
        estado[i] = E;
        signal(filosofo[i]);
```

```
pega_garfos()
    wait(lock);
    estado[i] = H;
    testa_garfos(i);
    signal(lock);
    wait(filosofo[i]);
```

```
solta_garfos()
    wait(lock);
    estado[i] = T;
    testa_garfos(fil_esq);
    testa_garfos(fil_dir);
    signal(lock);
```

## Como matar os filósofos de fome?

- É preciso ajustar os tempos.
- Veja o código: `tanen-4-2.c` e `tanen-5-1.c`
- Como implementar `tanen-8-2.c`?

# Como implementar este programa utilizando locks e variáveis de condição?

- Veja o código `tanen-mutex-bug.c` que faz substituições simples e descubra porque não funciona.