

# RMI

---

DANIEL ZAMBRANO, GABRIEL COSTA E VITOR AUGUSTO

# Interface Cliente-Servidor

---

Função readFile (número do Arquivo , Cliente)

Função writeFile ( número do Arquivo , Cliente , texto)

# Servidor (sem prioridades)

---

`readFile()` e `writeFile()`

## 3 Semáforos

- 1 semáforo de acesso para cada arquivo
  - 3 permissões (3 clientes podem acessar um arquivo ao mesmo tempo)
- Fairness **TRUE** para garantir FIFO

# Servidor (sem prioridades)

---

`readFile( numArq , idClient)`

- Verifica o semáforo de acesso do arquivo numArq
  - Se estiver com 0 permissão, bloqueia o cliente que quer fazer a leitura até acabar a escrita
- Pega uma permissão no semáforo de acesso (`acquire()`)
- `Sleep(5000)` para simular uma leitura de 5 segundos
- Lê o arquivo
- Libera uma permissão do semáforo de acesso (`release()`)

# Servidor (sem prioridades)

---

**writeFile**( numArq , idClient, string)

- Verifica o semáforo de acesso do arquivo numArq
  - Se houver menos que 3 permissões, o cliente é bloqueado até poder escrever
- Pega as três permissões de acesso (acquire())
- Sleep(10000) para simular uma escrita de 10 segundos
- Escreve no arquivo
- Libera as 3 permissões de acesso (release())

# Servidor (com prioridades)

---

readFile() e writeFile()

## 9 Semáforos

- 3 para cada arquivo
  - 1 de acesso ao arquivo com 3 permissões
  - 1 de escrita ao arquivo com 1 permissão
  - 1 para o readcount
- Fairness **TRUE** para garantir FIFO nas requisições
  - Os semáforos controlam a ordem as ações

# Servidor (com prioridades)

---

**readFile**( numArq , idClient)

- Pede permissão aos semáforos de acesso e readcount do arquivo (acquire())
  - Incrementa readcount e bloqueia a escrita
  - Libera o semáforo de readcount
- Pega uma permissão no semáforo de acesso (acquire())
- Sleep(5000) para simular uma leitura de 5 segundos
- Lê o arquivo
- Chama semáforo do readcount
- Decrementa readcount
  - Se readcount = 0, libera o semáforo de escrita, desbloqueando para escritor (release())
  - Libera readcount
- Libera o semáforo de acesso (release())

# Servidor (com prioridades)

---

**writeFile**( numArq , idClient, string)

- Chama o semáforo de escrita do arquivo numArq (aquire())
  - Se houver 0 permissão, significa que há pelo menos 1 leitor, então bloqueia até poder
- Quando liberado, escreve no arquivo
- Sleep(10000) para simular uma escrita de 10 segundos
- Libera a permissão de (release())