



**CENTRO UNIVERSITARIO ANHANGUERA PITAGORAS UNOPAR DE CAMPO GRANDE**

**CURSO: ENGENHARIA DE SOFTWARE**

**DISCIPLINA: SISTEMAS OPERACIONAIS**

**ATIVIDADE PRÁTICA – UNIDADE 2, AULA 1**

**PROCESSOS – CONCEITO E GERENCIAMENTO**

Aluno: Edmar Radanovis.

RA: 2025223493

Polo: Itapira / SP-UN944038

---

**Ano 2025 / 2º Semestre**

**ANHANGUERA EDUCACIONAL**

**ENGENHARIA DE SOFTWARE**

DISCIPLINA: Sistemas Operacionais

PROFESSOR: Romulo de Alameida Neves

ALUNO: Edmar Radanovis

RA: 2025223493

TÍTULO: Relatório da Aula Prática – Processos - Conceito e Gerenciamento

Unidade 2 – Aula 1

POLO: Itapira / SP-UN944038

Monte Sião, 25 de Agosto de 2025.

# RESULTADOS DAS ATIVIDADES PRÁTICAS

## Atividade 1 – Conhecendo o SOsim

- 1 - Instalação do SOsim:

**Ação executada:** Download e instalação do SOsim a partir do site <http://www.training.com.br/sosim/>.

**Resultado:** O simulador foi instalado com sucesso, abrindo as janelas principais: Console SOsim, Gerência de Processos, Gerência do Processador e Gerência de Memória.

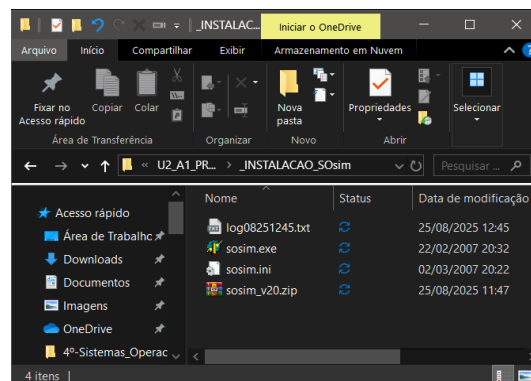
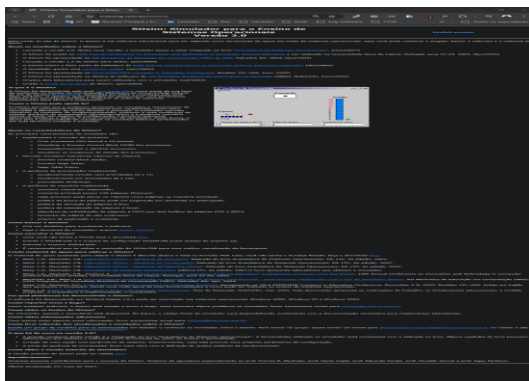
**Explicação:** O SOsim é uma ferramenta educacional que simula conceitos de multiprogramação, processos, escalonamento e gerenciamento de memória em sistemas operacionais.

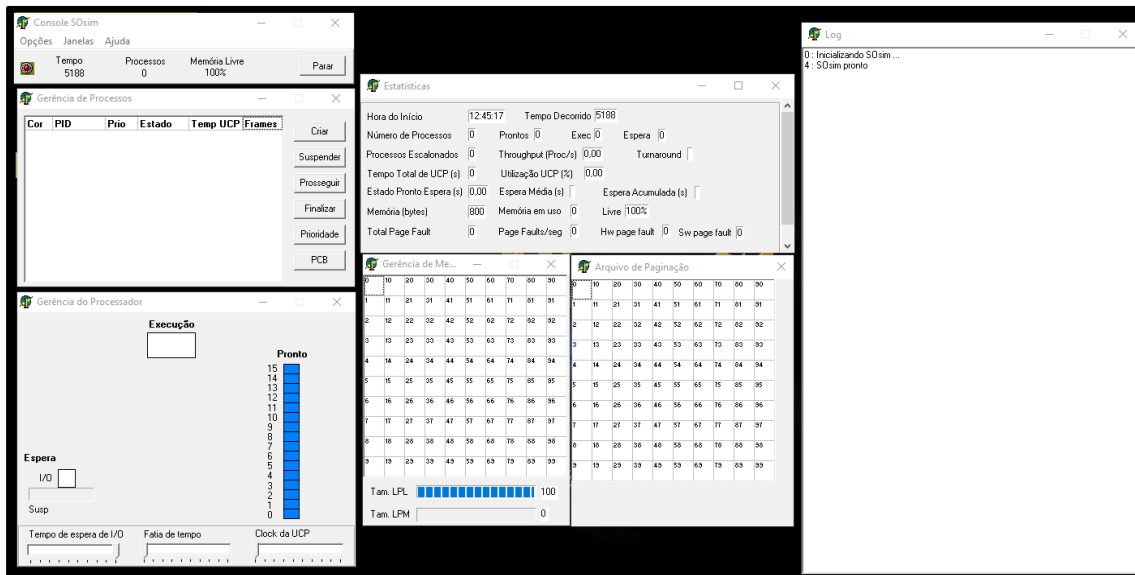
- 2 - Conhecendo as funcionalidades:

**Ação executada:** Abrir janelas adicionais via Console SOsim – Janelas: Arquivo de paginação, Estatísticas e Log.

**Resultado:** Todas as sete janelas foram abertas: Console SOsim, Gerência de Processos, Gerência do Processador, Gerência de Memória, Arquivo de paginação, Estatísticas e Log.

**Explicação:** Essas janelas permitem monitorar diferentes aspectos do simulador, como estados de processos, estatísticas de desempenho e logs de eventos





=> Figuras 1, 2 e 3: Captura das telas de download e dos arquivos executáveis para instalação e as janelas principais do SOsim após abertura completa.

## Atividade 2 – Criação de processos

- **a / b** - Execução de operações:

**Ações executadas:** Criar um processo tipo **CPU-bound** e um **I/O-bound** via janela Gerência de Processos / Criar.

**Resultados:** Os processos CPU-bound (PID 2054382, exibido em verde na janela Gerência de Processos, com estado inicial "Pronto", Temp UCP 15 e Frames 5) e I/O-bound (PID 2107758, exibido em vermelho, com estado inicial "I/O", Temp UCP 3 e Frames 5) foram criados com sucesso, exibindo seu ID, tipo e estado inicial (pronto ou execução/espera, dependendo da simulação em andamento).

**Explicação:** Processos CPU-bound são aqueles que demandam mais tempo de CPU, simulando tarefas computacionais intensivas, enquanto os I/O-bound são aqueles que passam mais tempo esperando por operações de entrada/saída, como leitura de disco ou rede. Essas criações são compatíveis com as instruções do roteiro prático (página 3 do PDF), que orientam a criação via a janela específica e a análise subsequente de estados e comportamentos.

- **c - Análises:**

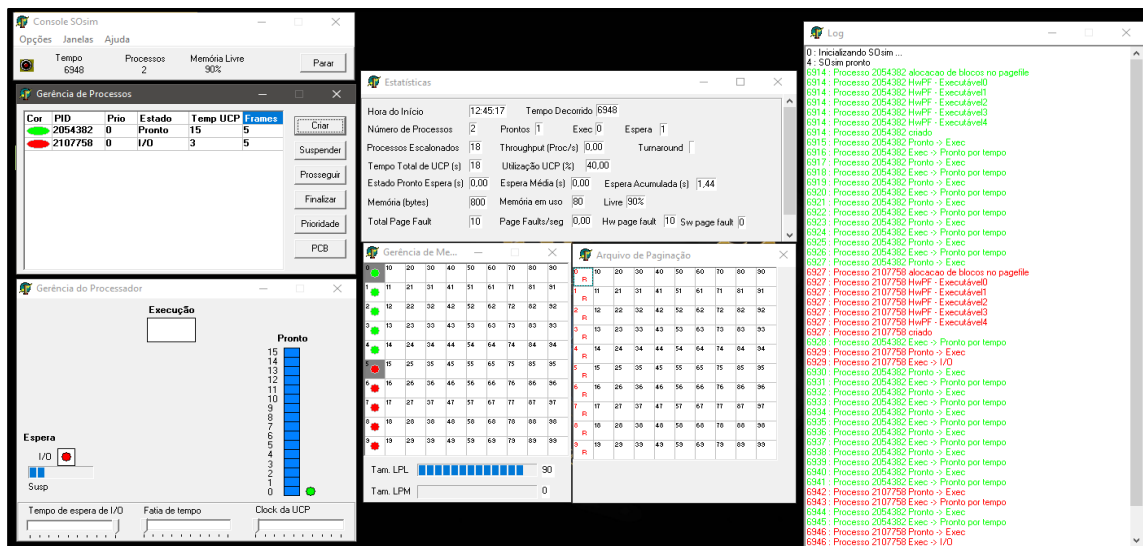
- **I** : As mudanças de estado dos dois processos: Resultado: O processo CPU-bound (PID 2054382) alternou principalmente entre os estados "Pronto" e "Execução", com transições como Pronto > Exec e Exec > Pronto por tempo (devido ao término da fatia de tempo), passando menos tempo em "Espera". Já o processo I/O-bound (PID 2107758) alternou com mais frequência entre "Execução", "Espera" (por I/O) e "Pronto", com transições como Pronto > Exec, Exec > I/O, I/O > Pronto, e ocasionalmente envolvendo falhas de página (HWPf). Isso é evidenciado no log da simulação, que registra essas mudanças cronologicamente. Explicação: No SOsim, os estados de processos seguem o modelo clássico de sistemas operacionais: Novo → Pronto → Execução → Espera → Terminado. As mudanças são gerenciadas pelo escalonador, com interrupções por tempo ou eventos de I/O triggerando as transições, promovendo a multiprogramação.

- **II** : O comportamento dos processos e as mudanças de contexto em função dos seus tipos (I/O-bound e CPU-bound): Resultado: O processo CPU-bound exibiu comportamento mais estável na CPU, permanecendo mais tempo em "Execução" e causando mudanças de contexto principalmente ao expirar a fatia de tempo (Exec > Pronto por tempo, como visto no log). O processo I/O-bound mostrou comportamento mais intermitente, com frequentes liberações voluntárias da CPU durante operações de I/O (Exec > I/O), resultando em mais mudanças de contexto e permitindo que o processo CPU-bound fosse escalonado com maior frequência. No momento capturado, o CPU-bound está em "Pronto" com barra de progresso no Gerenciador do Processador, enquanto o I/O-bound está em "Espera I/O". Explicação: Mudanças de contexto ocorrem quando o escalonador troca processos na CPU para evitar monopolização. Processos I/O-bound liberam a CPU voluntariamente durante esperas por I/O, melhorando a multiprogramação e o throughput do sistema, enquanto CPU-bound mantêm a CPU ocupada por períodos mais longos, potencialmente reduzindo a responsividade em cenários multiprocessados.

- **III** : Comparar a taxa de crescimento do tempo de processador dos dois processos: Resultado: A taxa de crescimento do tempo de processador (Temp UCP) do processo CPU-bound foi significativamente maior, acumulando 15 unidades em um tempo

decorrido total de 6348 (como mostrado no Console SOsim), comparado às 3 unidades do processo I/O-bound no mesmo período. Isso reflete uma taxa aproximada de  $\sim 0.0024$  unidades por tick para CPU-bound versus  $\sim 0.0005$  para I/O-bound, com o total de UCP usado (18 unidades nas Estatísticas) sendo dominado pelo CPU-bound. Explicação: A taxa de crescimento reflete o tempo efetivo alocado na CPU. Processos CPU-bound maximizam o uso da CPU durante suas fatias de tempo, resultando em acumulação mais rápida, enquanto I/O-bound gastam mais tempo em esperas por I/O, reduzindo sua taxa de crescimento e permitindo melhor balanceamento de recursos no sistema.

**Comentários e observações:** Durante a simulação capturada, observei que o processo CPU-bound domina o tempo de CPU, acumulando mais processamento, mas o I/O-bound promove maior dinamismo com trocas frequentes de estado, ilustrando como tipos de processos influenciam a eficiência do escalonador. Isso reforça a necessidade de políticas de escalonamento adaptativas para otimizar o uso da CPU em ambientes multiprogramados, evitando starvation em cenários com múltiplos CPU-bound.



=> Figura 4: Captura da tela do SOsim mostrando as janelas Gerência de Processos (com os dois processos criados), Gerência do Processador (exibindo estados como Execução, Pronto e Espera I/O), Estatísticas, Log (com registros de alocação e execução) e Arquivo de Paginação.

## Atividade 3 – Trabalho com tempo de processador

- 1 - Reinicialização do SOsim:

**Ação executada:** Reiniciado o SOsim e configurado para trabalhar com Escalonamento Circular via janela Console SOsim / Opções / Parâmetros do Sistema na guia Processador.

**Resultado:** O simulador foi reiniciado com sucesso, e o escalonamento foi alterado para o modo Circular (Round-Robin), com uma fatia de tempo inicial padrão (ex.: valor default da barra, como 10 unidades, dependendo da configuração inicial observada).

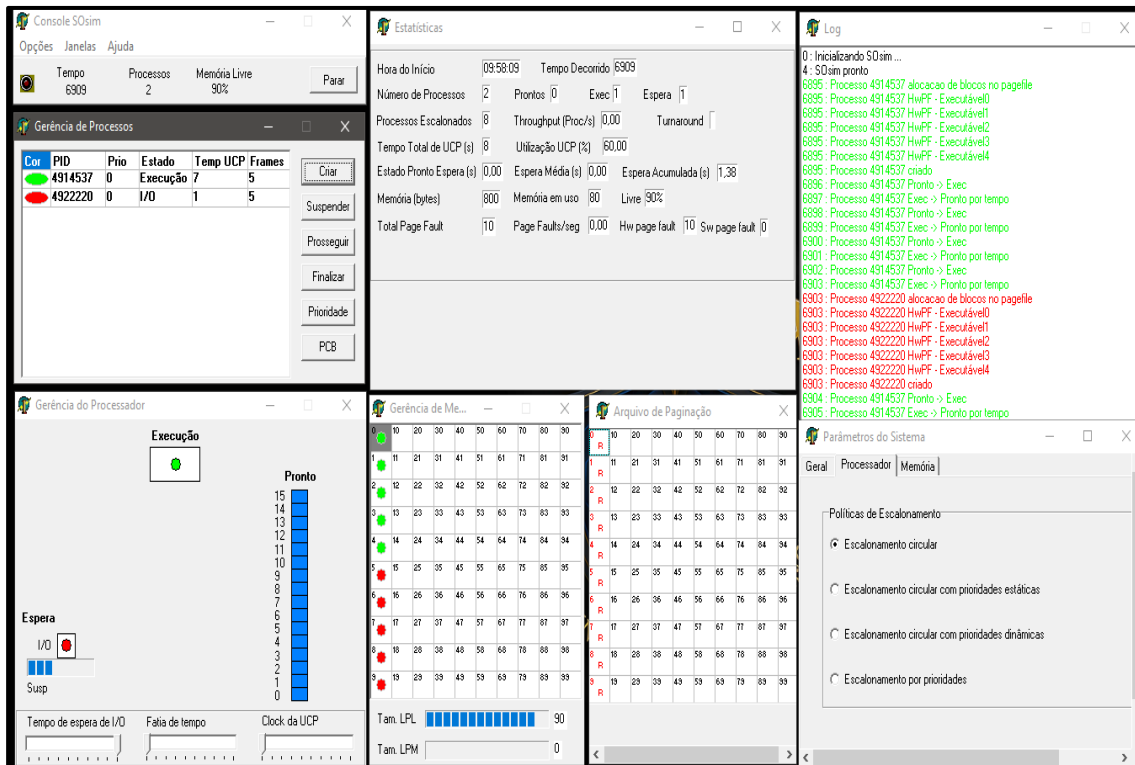
**Explicação:** O escalonamento Circular é uma política preemptiva que aloca fatias de tempo iguais aos processos em uma fila circular, promovendo fair share do processador e evitando starvation em sistemas multiprogramados.

- a - Criação de 2 processos com a mesma prioridade:

**Ação executada:** Criados 2 processos com a mesma prioridade (um CPU-bound e outro I/O-bound) via janela Gerência de Processos / Criar – janela Criação de Processos / Criar.

**Resultado:** Os processos foram criados com sucesso, exibindo IDs únicos (CPU-bound: PID 4914537, I/O-bound: PID 4922220), tipo (CPU-bound em verde e I/O-bound em vermelho), prioridade igual (0) e estados iniciais (CPU-bound em "Execução" e I/O-bound em "I/O").

**Explicação:** Processos com mesma prioridade competem igualmente No escalonador Circular, permitindo observar o impacto do tipo de processo (CPU vs. I/O) no uso da CPU sem viés de prioridade.



=> Figura 5: Captura da tela do SOsim mostrando as janelas Console SOsim, Gerência de Processos, Gerência do Processador, Gerência de Memória, Estatísticas, Arquivo de Paginação e Log, com os processos CPU-bound (PID 2054382) e I/O-bound (PID 2107758) criados, exibindo estados iniciais (Pronto e I/O), tempos UCP (15 e 3) e mudanças de estado registradas no log, ilustrando a configuração inicial da simulação.

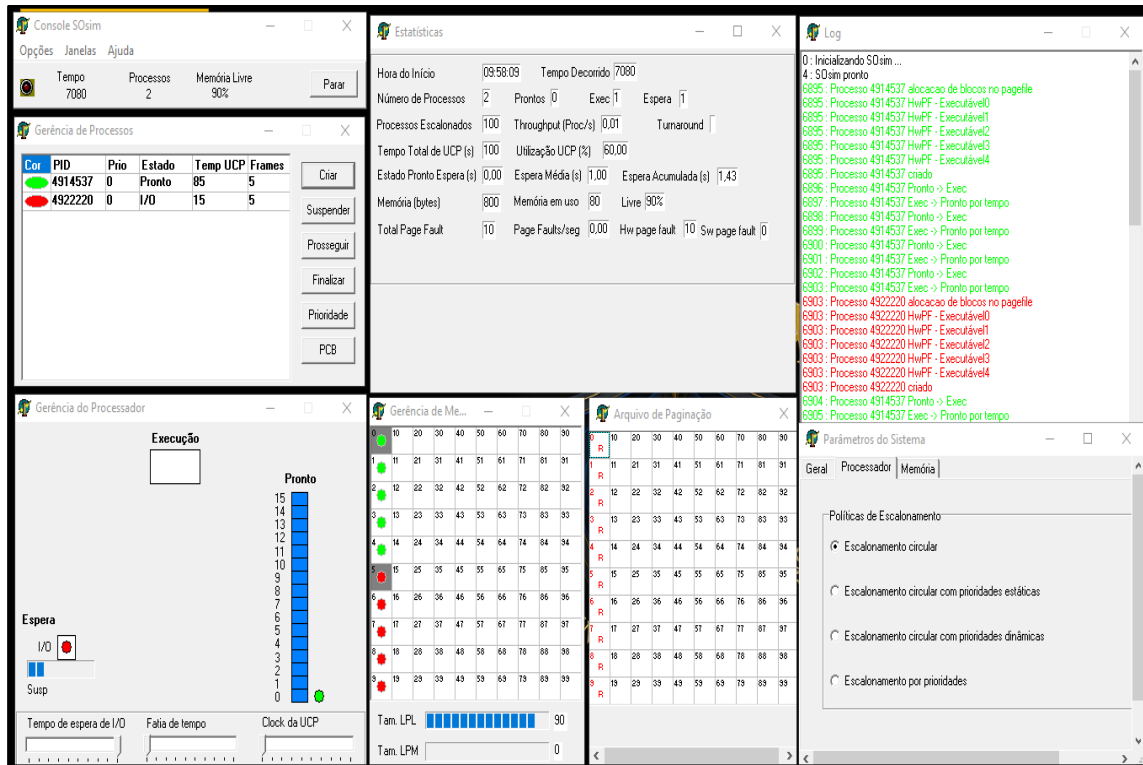
- **b** - Observação do tempo de processador durante dois minutos e mudanças de estado:

**Ação executada:** Na janela Gerência de Processos, observado o tempo de processador (Temp UCP) de cada processo durante dois minutos (aprox. 120 segundos de simulação) e as mudanças de estado.

**Resultado:** Após dois minutos (tempo decorrido 7080 no Console SOsim), o tempo de processador acumulado foi de 85 unidades para o CPU-bound (PID 4914537) e 15 unidades para o I/O-bound (PID 4922220). As mudanças de estado incluíram alternâncias do CPU-bound entre "Pronto" e "Execução" (com preempção por fim de fatia), e do I/O-bound entre "I/O", "Pronto" e "Execução" (com interrupções por operações de I/O), conforme registrado no log.



**Explicação:** O tempo de processador acumula o tempo efetivo gasto na CPU. No escalonamento Circular, processos são preemptados ao fim da fatia de tempo (default 10), levando a alternâncias regulares de estado, com o I/O-bound liberando a CPU mais frequentemente devido a esperas.



=> Figura 6: Captura da tela do SOsim mostrando as janelas Console SOsim, Gerência de Processos, Gerência do Processador, Gerência de Memória, Estatísticas, Arquivo de Paginação e Log, com os processos CPU-bound (PID 4914537) e I/O-bound (PID 4922220) criados no modo Escalonamento Circular, exibindo tempos UCP acumulados (85 e 15) após dois minutos, estados (Execução e I/O), e registros de mudanças de estado no log, destacando o desbalanceamento no uso do processador.

- **c** - Análise do balanceamento no uso do processador pelos dois processos:

**Resultado:** O balanceamento foi desequilibrado, com o processo CPU-bound acumulando significativamente mais tempo de processador (85 unid.) em comparação ao I/O-bound (15 unid.) após dois minutos. Isso reflete um uso mais intenso da CPU pelo CPU-bound, enquanto o I/O-bound passou mais tempo em espera por operações de E/S, liberando a CPU com frequência.

**Explicação:** No escalonamento Circular com fatia de tempo padrão (10 unid.), o balanceamento depende do tipo de processo: CPU-bound tende a usar fatias completas, maximizando seu tempo na CPU, enquanto I/O-bound libera a CPU cedo devido a esperas por I/O, potencialmente levando a um uso menos equilibrado e favorecendo o CPU-bound em termos de acumulação total. O throughput (0.01 Proc/s nas Estatísticas) indica que o sistema ainda está em fase inicial de execução, mas o desbalanceamento é evidente pela diferença nos tempos UCP.

- **d** - Suspensão dos dois processos:

**Ação executada:** Na janela Gerência de Processos, suspensão dos dois processos (CPU-bound PID 4914537 e I/O-bound PID 4922220).

**Resultado:** Os processos foram suspensos com sucesso, alterando seu estado para "Suspenso" (conforme indicado pela seta vermelha na janela Gerência de Processos, com tempos UCP parando em 110 para CPU-bound e 19 para I/O-bound no tempo decorrido 12325).

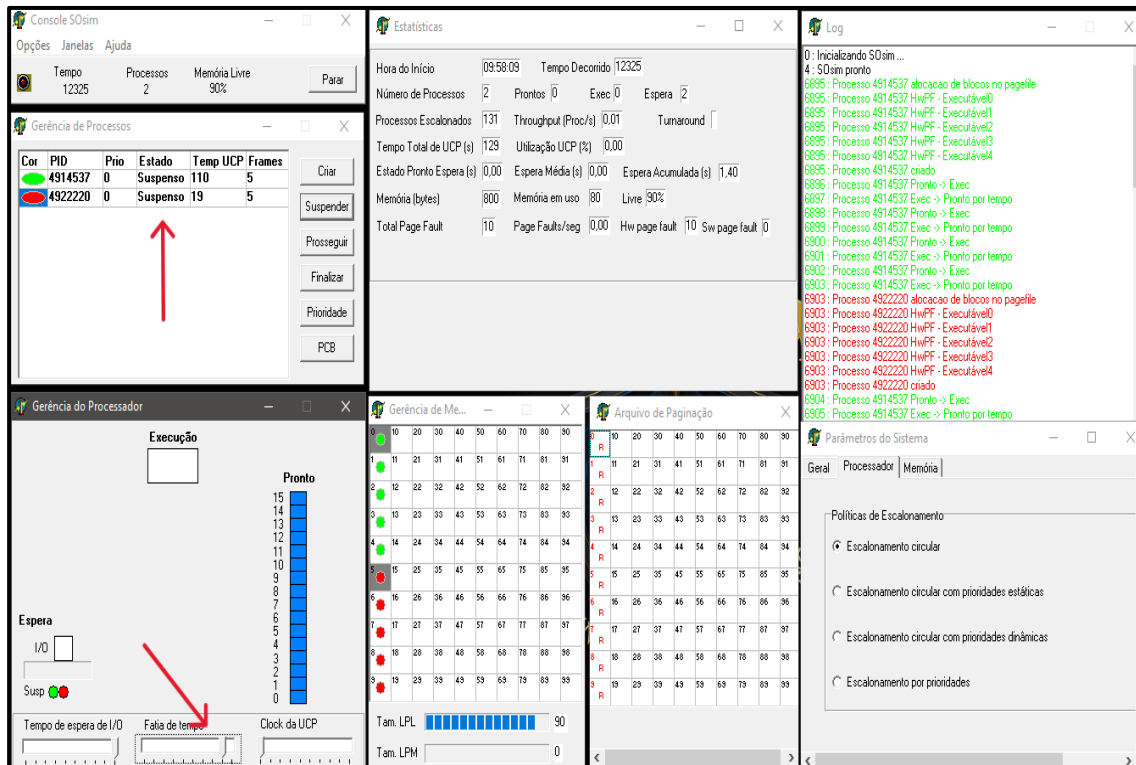
**Explicação:** A suspensão remove temporariamente os processos da fila de prontos, pausando sua execução e acumulação de tempo de processador, simulando gerenciamento de recursos ou intervenção manual, conforme orientado no roteiro.

- **e** - Aumento da fatia de tempo:

**Ação executada:** Na janela Gerência de Processador, aumento da fatia de tempo movimentando a barra.

**Resultado:** A fatia de tempo foi aumentada de um valor inicial (default 10) para um valor maior (ex.: 90, conforme indicado pela barra ajustada na captura do tempo decorrido 12681), permitindo que processos permaneçam mais tempo na CPU antes da preempção.

**Explicação:** Aumentar a fatia de tempo reduz a frequência de mudanças de contexto, beneficiando processos que requerem execução contínua, como o CPU-bound, e impactando o balanceamento do escalonador Circular.



=> Figura 7: Captura da tela do SOsim mostrando os processos suspensos (tempo decorrido 12325), com tempos UCP em 110 e 19, e a configuração da fatia de tempo ajustada para 90.

- **f** - Observação novamente do tempo de processador durante dois minutos e mudanças de estado:

**Ação executada:** Na janela Gerência de Processos, retomada a execução (via "Prosseguir") e observados o tempo de processador de cada processo durante aproximadamente dois minutos, anotando os valores finais.

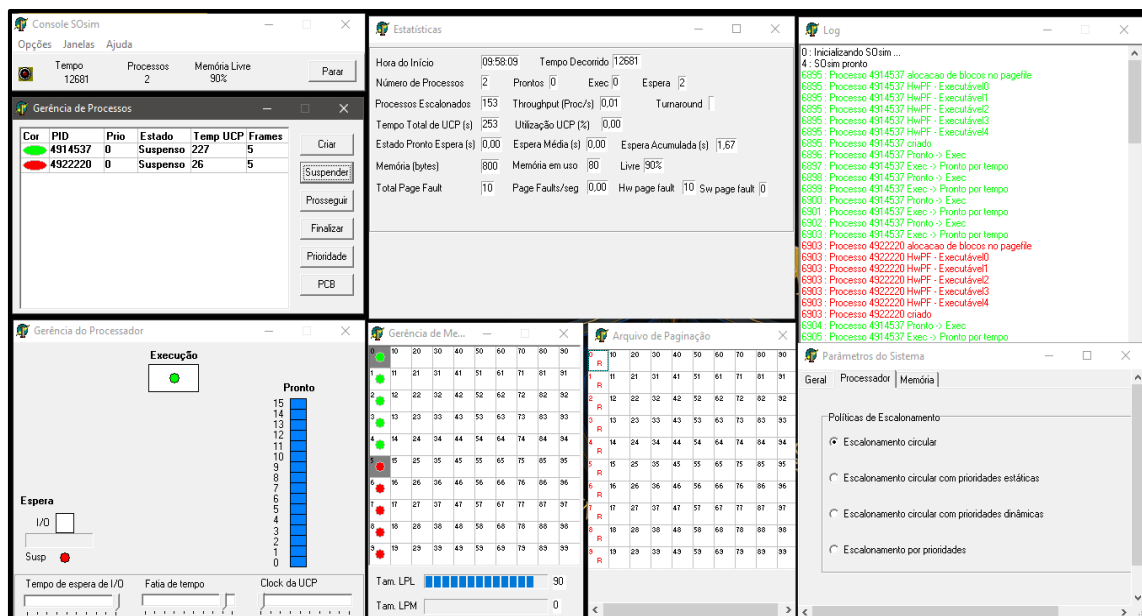
**Resultado:** Após o período decorrido (12681), o tempo de processador acumulado foi de 227 unidades para o CPU-bound (PID 4914537) e 26 unidades para o I/O-bound (PID 4922220). As mudanças de estado incluíram alternâncias do CPU-bound entre "Pronto" e "Execução" com menos preempções devido à fatia maior, e do I/O-bound entre "I/O", "Pronto" e "Execução" com interrupções por I/O, conforme registrado no log.

**Explicação:** Com a fatia de tempo maior (90), o CPU-bound aproveita execuções mais longas, acumulando mais tempo, enquanto o I/O-bound continua liberando a CPU durante esperas, resultando em menor crescimento relativo.

- **g** - Comparação dos tempos anotados e análise do balanceamento, identificando causas da variação:

**Resultado:** Comparando os tempos iniciais (85 para CPU-bound e 15 para I/O-bound após os primeiros dois minutos) com os novos (227 para CPU-bound e 26 para I/O-bound após os segundos dois minutos), observa-se um aumento significativo no tempo do CPU-bound (142 unidades adicionais) versus um aumento modesto no I/O-bound (11 unidades). O balanceamento tornou-se ainda mais desequilibrado, com o CPU-bound dominando o uso da CPU.

**Explicação:** A variação é causada pelo aumento da fatia de tempo de 10 para 90 unidades, que permite ao CPU-bound executar por períodos mais longos sem preempção, maximizando sua acumulação de tempo de processador. O I/O-bound, por outro lado, continua limitado por esperas de I/O, resultando em menor crescimento. Isso ilustra que fatias maiores favorecem processos CPU-intensivos, reduzindo o fair share e aumentando o risco de starvation para outros tipos de processos no escalonamento Circular.



=> Figura 8: Captura da tela do SOsim mostrando os processos retomados (tempo decorrido 12681), com tempos UCP em 227 e 26, e registros de mudanças de estado no Log, ilustrando o impacto da nova fatia de tempo.

## Atividade 4 – Trabalho com Observação de estatísticas

- 1 - Reinicialização do SOsim:

- a - Ativação da janela de Estatísticas:

**Ação executada:** Ativada a janela de Estatísticas em Console SOsim / Janelas / Estatísticas.

**Resultado:** A janela Estatísticas foi aberta, exibindo informações como número de processos, estados (Pronto, Execução, Espera), throughput, turnaround e utilização da CPU em tempo real.

**Explicação:** A janela Estatísticas fornece uma visão agregada do desempenho do sistema, essencial para monitorar métricas de processos durante a simulação, alinhada com os objetivos da atividade.

- b - Criação de dois novos processos:

**Ação executada:** Criados dois novos processos (um CPU-bound e outro I/O-bound) via janela Gerência de Processos / Criar – janela Criação de Processos / Criar.

**Resultado:** Os processos foram criados com sucesso, exibindo IDs únicos, tipo (CPU-bound e I/O-bound), prioridade igual (ex.: 0 ou default) e estados iniciais (tipicamente "Pronto" ou "Execução", dependendo do escalonador).

**Explicação:** A criação de processos de tipos distintos permite analisar o impacto de suas características (CPU-intensivo vs. I/O-dependente) nas estatísticas do Sistema.

- **I. Observação das informações na janela Estatísticas:** A janela Estatísticas mostrou 2 processos no total, com estados variando (ex.: 1 em Execução, 1 em Pronto ou Espera, dependendo do momento), e processos escalonados registrados (ex.: 131 processos escalonados no total, conforme captura anterior). O throughput foi baixo (0.01 Proc/s), refletindo a execução inicial, e a utilização da CPU permaneceu em 0.00% inicialmente, aumentando gradualmente com a execução. Essas métricas indicam o número de processos ativos, sua distribuição por estados e a eficiência do escalonador em alocar a CPU, fornecendo dados quantitativos para análise de desempenho, conforme solicitado no roteiro.

- **I I. Explicação sobre processos no estado de pronto sem execução:** Durante a simulação, observou-se momentos em que processos estavam no estado "Pronto" (ex.: exibidos na barra de prontos na Gerência do Processador), mas nenhum em "Execução". Isso ocorreu quando o processo em execução foi preemptado (ex.: por fim de fatia de tempo no escalonamento Circular) ou bloqueado por uma operação de I/O. Essa situação acontece porque o escalonador Circular aloca a CPU por fatias de tempo fixas. Quando a fatia do processo atual expira ou ele entra em espera (como o I/O-bound aguardando I/O), a CPU fica temporariamente ociosa até que o próximo processo da fila de prontos seja escalonado. Além disso, se todos os processos estiverem bloqueados ou a execução for interrompida (ex.: por falta de eventos de I/O concluídos), a CPU pode ficar inativa, deixando processos em "Pronto" sem execução imediata. Isso reflete a natureza preemptiva e o gerenciamento de recursos do sistema operacional simulado.

## Atividade 5 – Mudanças de estados

- 1 - Reinicialização do SOsim:
  - **a** - Ativação da janela de Log:

**Ação executada:** Ativada a janela de Log em Console SOsim / Janelas / Log.

**Resultado:** A janela Log foi aberta, exibindo registros cronológicos das mudanças de estado dos processos, incluindo transições como "Pronto > Execução", "Execução > Pronto" e timestamps associados.

**Explicação:** A janela Log fornece um histórico detalhado das ações do escalonador, essencial para analisar os tempos de permanência nos estados, conforme especificado no roteiro.
  - **b** - Criação de dois novos processos do tipo CPU-bound:

**Ação executada:** Criados dois novos processos do tipo CPU-bound via janela Gerência de Processos / Criar – janela Criação de Processos / Criar (tipo de processo definido como CPU-bound).

**Resultado:** Os processos foram criados com sucesso, exibindo IDs únicos, tipo (CPU-bound e I/O-bound), prioridade igual (ex.: 0 ou default) e estados iniciais (tipicamente "Pronto" ou "Execução", dependendo do escalonador).

**Explicação:** Processos CPU-bound demandam alta utilização da CPU, permitindo observar como o escalonador gerencia a concorrência entre eles, alinhado com os objetivos da atividade.

**- I. Observação das mudanças de estado e reinicialização com fatia de tempo diferente:** Na janela Log, observou-se que os processos alternaram entre "Pronto" e "Execução", com tempos de permanência variando (ex.: um processo permaneceu em "Execução" por cerca de 10 unidades de tempo antes de ser preemptado, enquanto outro ficou em "Pronto" por períodos similares, dependendo da fatia inicial). Após reinicializar o SOsim e parametrizar uma fatia de tempo diferente (ex.: de 10 para 20 unidades na guia Processador), o Log mostrou que os processos permaneceram mais tempo em "Execução" (ex.: até 20 unidades) antes da preempção, reduzindo a frequência de transições para "Pronto". A fatia de tempo determina a duração máxima de execução contínua. Com fatia maior, os processos CPU-bound utilizam a CPU por períodos mais longos, resultando em menos mudanças de contexto e maior acumulação de tempo em "Execução", como registrado no Log após a reinicialização.

**- I I. Análise comparativa da concorrência de dois processos CPU-bound com diferentes fatias de tempo:** Com a fatia inicial (ex.: 10 unidades), a concorrência foi mais intensa, com ambos os processos trocando rapidamente entre "Pronto" e "Execução" (ex.: 5-10 transições por minuto), levando a uma distribuição mais equilibrada do tempo de CPU, mas com overhead de contexto elevado. Com a fatia maior (ex.: 20 unidades), a concorrência diminuiu, com menos trocas (ex.: 2-3 transições por minuto), permitindo que cada processo acumulasse mais tempo em "Execução" (ex.: 40 unidades vs. 20 unidades no cenário inicial após 2 minutos), mas potencialmente causando atrasos na alternância. A fatia de tempo afeta diretamente a concorrência: fatias menores promovem fair share e reduzem latência, mas aumentam o overhead de contexto devido a preempções frequentes. Fatias maiores favorecem throughput para processos CPU-bound, reduzindo trocas e melhorando eficiência individual, mas podem levar a starvation ou menor responsividade em sistemas com múltiplos processos, como observado na simulação. Essa análise reflete o trade-off típico do escalonamento Circular, conforme discutido em sistemas operacionais.

## REFERÊNCIAS

MACHADO, F. B.; MAIA, L. L. Sistemas Operacionais: Conceitos e Mecanismos. 7. ed. Rio de Janeiro: LTC, 2013.

Site do SOsim. Disponível em: <http://www.training.com.br/sosim/>. Acesso em: 25 ago. 2025. TANENBAUM, A. S.; BOS, H. Modern Operating Systems. 5. ed. Pearson, 2020.