

**UC Sistemas Operacionais  
ICT/UNIFESP**

Prof. Bruno Kimura  
bruno.kimura@unifesp.br  
21/06/2018

**LAB 3: Gerenciamento de Memória**

**Metodologia:** Trabalho individual ou em grupo de no máximo 3 (três) alunos a ser desenvolvido em laboratório de informática através de codificação na linguagem C.

**Data de entrega:** 28/06/18

**Forma de entrega:** Código .c deve ser enviado no SEAD. Insira como comentário no código o nome e matrícula de cada integrante do grupo.

**Observação:** Somente serão aceitos trabalhos **autênticos**. Cópias (entre grupos e/ou de fontes da Internet) serão anuladas.

**Descrição:**

Em gerenciamento de memória virtual, duas tarefas são fundamentais:

1. **Mapeamento de Endereços.** O endereço virtual em espaço de página virtual é mapeado para endereço físico em quadro de página na memória principal. Quando um endereço virtual é referenciado pela CPU, tal endereço é tratado pela MMU. Se o endereço virtual estiver indicado tabela de páginas virtuais como ausente em memória, ocorre uma falta de página (*page miss*), gerando uma interrupção (*trap*), para que o SO carrega a página em memória e a mapeie na entrada na tabela de página virtual. Se não houver espaço para um novo mapeamento dentro do espaço de quadros de páginas físicas em memória, o SO tem que substituir uma página física, movendo-a temporariamente para o disco, de modo a liberar o espaço necessário para a nova página a ser carregada. Posteriormente, quando necessário, i.e. quando houver uma referência à página substituída, gerando uma falta de página, o SO terá de trazer de forma íntegra essa página do disco de volta para a memória. Vale destacar que pode ocorrer uma nova substituição nesse evento, caso não haja espaço novamente.
2. **Substituição de Página.** A falta de página demanda a escolha de qual página física será substituída em memória, tarefa a qual é realizada por algum algoritmo. O algoritmo pode ser baseado, por exemplo, na escolha da página mais ou menos utilizada recentemente, mais ou menos utilizada frequentemente, filas, envelhecimento, relógios lógicos, grupos de trabalho, entre outros. Além da eficiência do algoritmo, que é medida pela frequência de falta de páginas (quanto menor, melhor), o importante também é manter a persistência e integridade da página, de modo que um mesmo conteúdo e suas informações de gerenciamento não possam divergir entre diferentes espaços de endereçamento (página virtual, quadro de página em memória, quadro de página em disco).

Neste exercício, implemente na linguagem C o algoritmo de substituição de página **WSClock**. Defina os parâmetros (e.g., tau, t, intervalos das interrupções de relógio), as entidades (página virtual, quadro de página, memória, disco), e simule o algoritmo com eventos aleatórios (referências às páginas por localidade de referência).