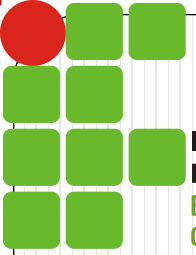


INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
BAHIA
Campus Salvador

Aula 5 : Threads

Instituto Federal da Bahia
INF009 - Sistemas Operacionais
Profª Flávia Maristela

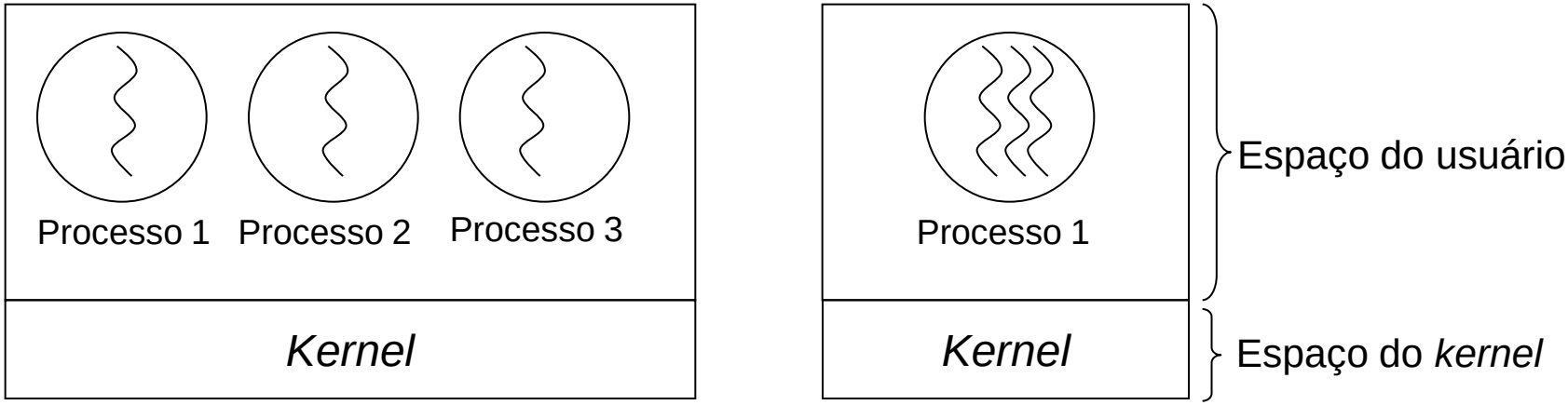

Você não faz
nada certo?
GLASBERGEN
Copyright 2003 Randy Glasbergen. www.glasbergen.com

THREADS

- Motivação:
  - A necessidade de compartilhar diferentes recursos do computador deu origem a *PROGRAMAÇÃO CONCORRENTE*.
  - Neste cenário, um programa que tinha vários processos com um único fluxo de execução passou a ter vários processos.
  - Cada processo possuía um ou mais fluxos de execução.

THREADS

- *Threads* representam uma abstração para uma nova subdivisão necessária para os processos.



THREADS

- Definição:
  - "Entidades escalonadas para execução"

Tanenbaum

  - "Fluxo de execução dentro de um processo"

Rômulo Oliveira

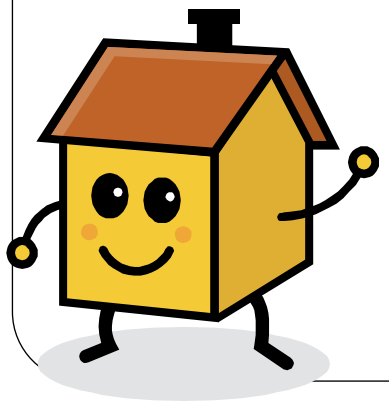
  - "Unidade básica de utilização da CPU"

Silbershatz

- Assim como os processos, as *threads* também possuem estados.

THREADS

- *Multithreading*
  - Termo usado para caracterizar um processo com várias *threads*.
  - Sistema **multithread** executa as *threads* tão rapidamente, que passa ao usuário a impressão de que as mesmas estão sendo executadas em paralelo.
  - O termo também está ligado a dispositivos de hardware que permitem a execução de várias *threads*.



PARA PENSAR! Qual é a diferença entre os seguintes termos:
-Paralelismo
-Pseudo-paralelismo
-Multiprogramação
-Multithreading

THREADS

- *Threads* compartilham os recursos de um processo;
- *Threads* de um mesmo processo não são independentes entre si
- Em sistemas *multithread*, normalmente cada processo inicia com apenas uma *thread*
  - Esta *thread* tem a capacidade de criar novas *threads*

THREADS

- *Threads* não representam a solução para todos os problemas:
  - Se um processo é duplicado, ele deve manter todas as *threads* do processo pai?
  - Se uma *thread* estava bloqueada no momento da cópia de um processo, a *thread* filha também vai estar?
  - Quando um dado é útil para uma *thread*, quem vai receber uma cópia, apenas o processo pai? O processo filho também deve receber?

THREADS

- *Threads* podem ser gerenciadas em dois níveis:

- Nível do usuário
- Nível do *kernel*

THREADS
(-- nível do usuário --)

- *Kernel* do sistema operacional não tem conhecimento sobre tais *threads*.

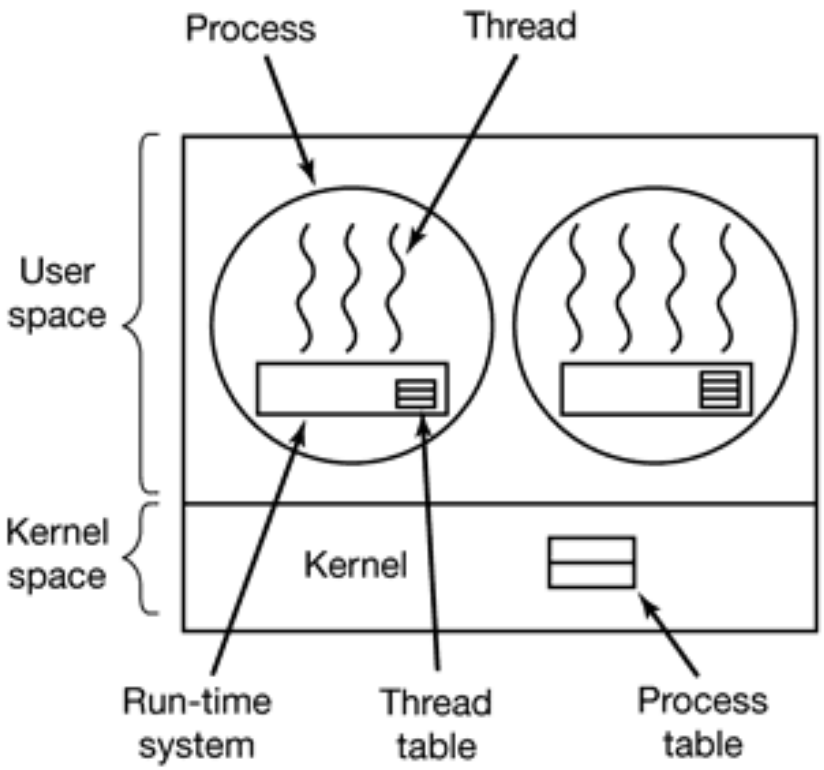
- Sistema operacional enxerga apenas um único processo com uma única *thread*.

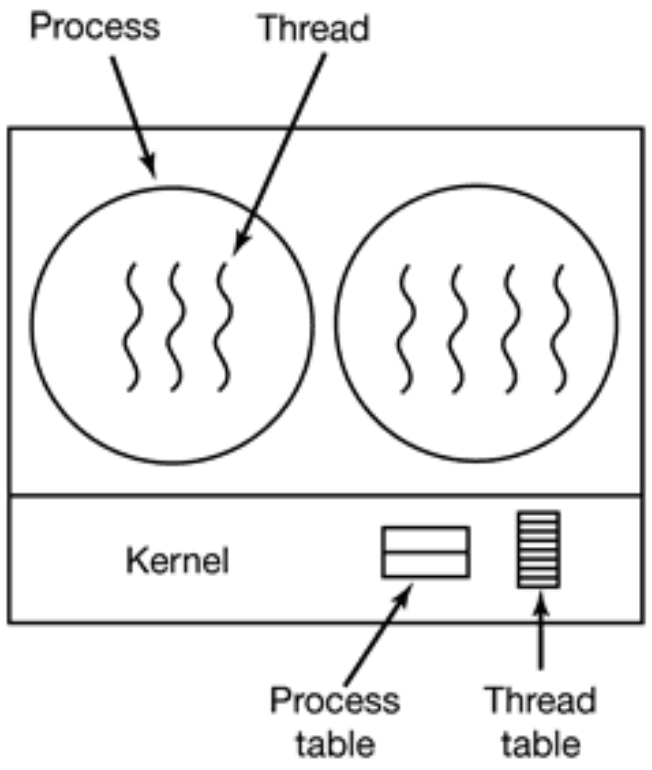
THREADS
(-- nível do kernel --)

- *Kernel* do sistema operacional controla todas as operações entre *threads:*
  - *Create*
  - *Terminate;*
  - *Join*
  - *Yield*
  - *Resource sharing* (compartilhamento de recursos)

THREADS

(-- nível usuário *vs.* nível kernel --)





Dúvidas?

Exercícios I

1. O que são *threads*?
2. Porque as *threads* foram criadas?
3. Explique as operações entre *threads*.
4. Explique como as *threads* são criadas.
5. Explique como as *threads* podem ser executadas.
6. Quais os estados de uma *thread*?