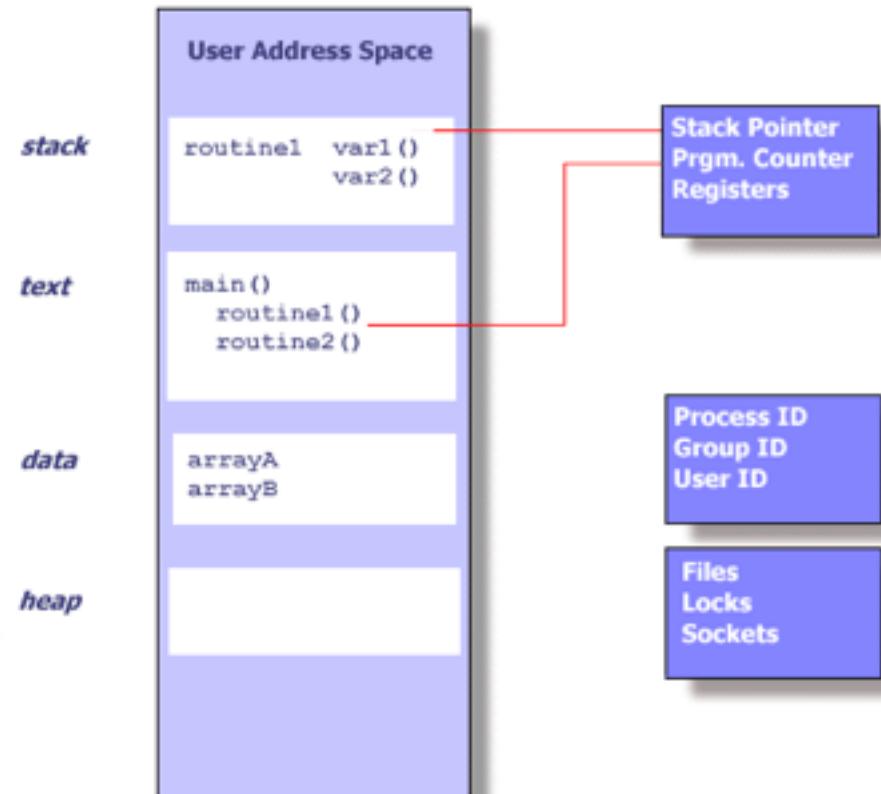


# Programação Pthreads

Dr. Osmar Marchi dos Santos

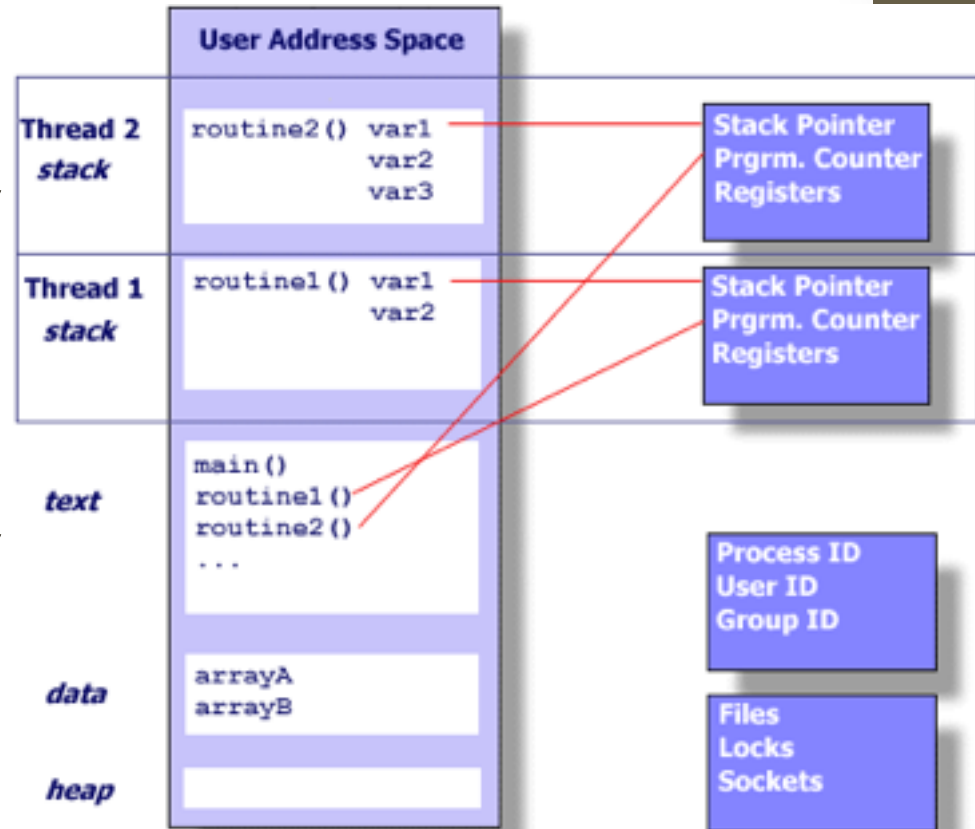
# Conceitos Básicos

- Processo
  - Entidade ativa, uma instância de um programa, que está sendo executada
  - Possibilita a execução de diferentes instruções, sendo escalonado pelo sistema operacional para execução



# Conceitos Básicos

- Thread
  - Um processo pode ser composto por várias threads de execução
  - Cada thread executa de forma independente, mas tem acesso aos recursos do processo como, por exemplo, as variáveis globais do processo
  - A execução independente ocorre porque as threads mantêm suas próprias estruturas de execução



# Conceitos Básicos

- Thread
  - Ao compartilhar informações, todos os recursos modificados por uma thread afetam outras
  - Dessa forma, existe uma “concorrência” entre as threads que precisam de programação do desenvolvedor para sincronizar explicitamente o acesso aos recursos pelas threads

# Pthreads

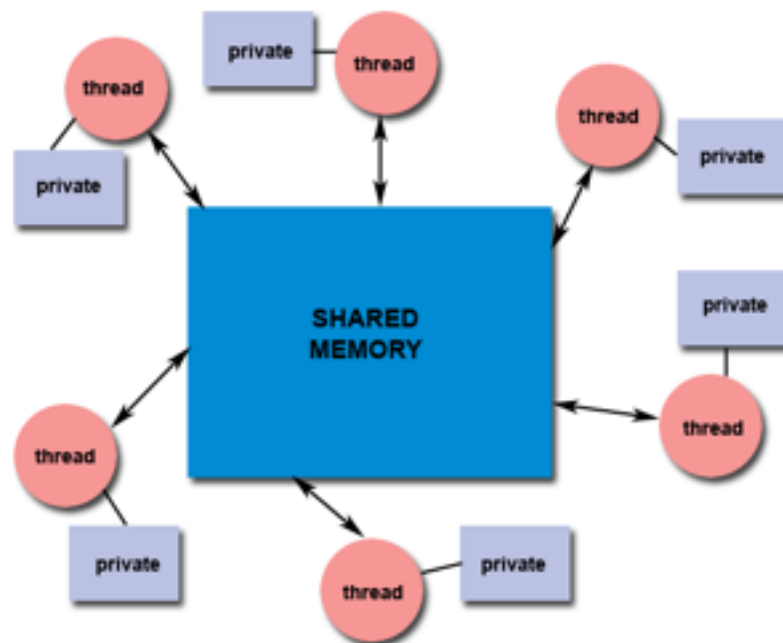
- É um conjunto padronizado de interface de programação
- Está definida no padrão POSIX (Portable Operating System Interface), IEEE 1003.1
- Historicamente, foi criada dentro do padrão IEEE POSIX 1003.1c
- POSIX threads ou, simplesmente, Pthreads
- São definidas na linguagem C, através da implementação do *header **`pthread.h`***

# Pthreads

- O uso de Pthreads está ligado:
  - A velocidade de criação comparado a processos normais (chamadas *fork()*)
  - Ao compartilhamento eficiente de informações entre threads (sem ter a necessidade de criar estruturas específicas de compartilhamento como uso de *shm\_open()* em UNIX)
- Outra motivação, ligada ao uso de concorrência em geral, está relacionada a ganhos expressivos de performance pela aplicação

# Programação Concorrente

- Compartilhamento de informações em conjunto com threads (ou tarefas) concorrente possibilita um aumento considerável na performance das aplicações
- Porém, introduz problemas relacionados a sincronização entre as tarefas



# API Pthreads

- As funções oferecidas pela API (*Application Programming Interface*) são divididas em 4 grandes partes:
  - Gerenciamento de Threads: rotinas para criar, liberar, esperar, definir atributos, entre outros;
  - *Mutexes*: rotinas para criar, destruir, *lock*, e *unlock* de *mutexes*. *Mutex*, abreviação para “*Mutual Exclusion*”;
  - Variáveis de Condição: rotinas para trabalhar com variáveis de condições associadas a *mutexes*;
  - Sincronização: rotinas para *locks* leitura/escrita e *barriers*.
- Por convenção, todos os identificadores para tipos e funções em pthreads iniciam com o prefixo “pthread\_”
- Para compilar programas Pthreads utiliza-se a inclusão da biblioteca Pthread (exemplo com gcc): gcc -lpthread



# Criando e Terminando Threads

- `pthread_create(thread_id, attr, rotina_execução, argumentos)`
- `pthread_exit(status)`
- `pthread_join(thread_id, status)`
- `pthread_cancel(thread)`
- `pthread_attr_init(attr)`
- `pthread_attr_destroy(attr)`

# Exemplo:

- `#include <pthread.h>`
- `#include <stdio.h>`
- `#include <stdlib.h>`
- `#define NUM_THREADS 10`
- `void *PrintHello(void *threadid) {`
- `long tid;`
- `tid = (long)threadid;`
- `printf("Hello World! It's me, thread #%ld!\n", tid);`
- `pthread_exit(NULL);`
- `}`
- `int main (int argc, char *argv[]) {`
- `pthread_t threads[NUM_THREADS];`
- `int rc;`
- `long t;`
- `for(t=0; t<NUM_THREADS; t++){`
- `printf("In main: creating thread %ld\n", t);`
- `rc = pthread_create(&threads[t], NULL, PrintHello, (void *)t);`
- `if (rc){`
- `printf("ERROR; return code from pthread_create() is %d\n", rc);`
- `exit(-1);`
- `}`
- `}`
- `pthread_exit(NULL);`
- `}`

# Exercícios:

- Faça um programa com 10 threads que incrementam de forma concorrente uma variável global (chamada `int var_global`) 1000 vezes. A cada incremento, cada thread imprime o valor incrementado e o identificador da thread que modificou o valor.
- Faça um programa que calcule os números primos existentes entre 0 e 99999 utilizando 9 threads e guardando em um vetor. Ao final, o programa principal deve imprimir os valores calculados.

# Referências

- <https://computing.llnl.gov/tutorials/pthreads/>
- [http://pages.cs.wisc.edu/~travitch/pthreads\\_primer.html](http://pages.cs.wisc.edu/~travitch/pthreads_primer.html)