

Lập trình multithread

Giảng Viên: Lưu An Phú



Agenda

- Ưu điểm nhược điểm
- Tạo và hủy thread
- Đồng bộ giữa các thread

- Tối ưu được hiệu năng xử lý của tất cả các core
- Các thread được lập lịch một cách độc lập
- Tránh được trạng thái block chương trình
- Cần lưu ý đồng bộ dữ liệu giữa các thread

Tạo mới một thread

- `#include <pthread.h>`
- `int pthread_create(pthread_t *restrict tidp,
const pthread_attr_t *restrict attr, void
*(*start_rtn)(void *), void *restrict arg);`
- Returns: 0 if OK, error number on failure

- Thread call `pthread_exit`, return hoặc bị cancel bởi 1 thread khác.
 - `void pthread_exit(void *rval_ptr)`
- 1 thread khác phải clear resource của thread đã exit
- `int pthread_join(pthread_t thread, void **rval_ptr)`
 - Block cho đến khi thread cần đợi kết thúc
 - Return `EINVAL` nếu thread kia đã kết thúc từ trước

- Đếm số lượng số lẻ từ 1 đến 100 tỷ. Code theo 2 trường hợp, trường hợp 1 sử dụng multithread, trường hợp 2 không sử dụng multithread. So sánh thời gian của 2 trường hợp.
- Viết 1 chương trình C tạo 10 file, mỗi file chứa 5 triệu số ngẫu nhiên có 1 chữ số. Sử dụng multithread và không multithread rồi so sánh thời gian.

- Ví dụ minh họa về mất đồng bộ khi 2 thread cùng đọc ghi vào 1 biến
- Mất đồng bộ ghi đọc ghi vào mảng và file

Các phương thức đồng bộ dữ liệu

- Chỉ có một khóa duy nhất.
- Trong một thời điểm chỉ 1 thread có được khóa
- Minh họa
 - `int pthread_mutex_init(pthread_mutex_t *restrict mutex, const pthread_mutexattr_t *restrict attr)`
 - `int pthread_mutex_destroy(pthread_mutex_t *mutex)`
 - `int pthread_mutex_lock(pthread_mutex_t *mutex)`
 - `int pthread_mutex_unlock(pthread_mutex_t *mutex)`

- Viết chương trình C tạo 2 thread, cả 2 thread lần lượt tăng biến đó lên 1 đơn vị, sử dụng mutex để đồng bộ việc ghi vào biến đó.
- Viết 1 chương trình C tạo ra 2 thread, 2 thread đều ghi vào chung 1 file, sử dụng mutex để bảo vệ file, nên sử dụng wrapper function thay cho hàm ghi file thông thường.

- Có 1 hoặc nhiều khóa
- Trong một thời điểm có thể có 1 hoặc nhiều thread có được khóa
- Minh họa
- wrapper function

- Viết 1 chương trình C tạo ra 3 thread, các thread lần lượt được tăng 1 biến chung thêm một đơn vị và ghi giá trị mới vào 1 trong 2 file output. Sử dụng semaphore để đồng bộ việc ghi vào 2 file
- Tạo 10 ma trận kích thước 200x200. Nhân 10 ma trận đó với nhau sử dụng multi thread.

Thank you

