

Hướng dẫn Lab 4.1 – Tạo tiểu trình con

Mục tiêu	Lý thuyết liên quan	Tài nguyên
Tạo thread	Ch4: Threads and Concurrency.	Sử dụng image Ubuntu 16 / 18
Kết thúc thread		

Yêu cầu sinh viên: Nắm rõ lý thuyết, trả lời tốt các câu hỏi. Hiểu và thực thi các đoạn mã đã cung cấp, lời gọi tạo ra tiểu trình và các kết thúc đồng bộ. Áp dụng sinh tiểu trình thực hiện các công việc tại bài tập.

Đánh giá sinh viên: Trả lời các vấn đề lý thuyết. Kỹ năng thực hành. Bài tập.

Yêu cầu nộp bài: các tập tin mã nguồn .c và tập tin khả thực thi .out của các Yêu cầu trong buổi thực hành và Bài tập cuối hướng dẫn trong thời gian cho phép của giảng viên.

Preferences

[1] Abraham Silberschatz, Peter B. Galvin, Greg Gagne, [2018], Operating System Concepts, 10th edition, John Wiley & Sons, New Jersey.

Programming Problems of Chapter 4.

[2] Greg Gagne , [2019], GitHub OS-BOOK OSC10e, Westminster College, United States

Access <https://github.com/greggagne/osc10e> in September 2019.

[3] Linux manual page, [2021], on clock(3)

Access <https://man7.org/linux/man-pages/man3/clock.3.html>

[4] Linux manual page, [2021], on clock(3)

Access <https://man7.org/linux/man-pages/man2/gettimeofday.2.html>

?	- Tiểu trình là gì?	Ch4 p.160	Slide Ch4.5
	- Sự khác nhau của tiểu trình và tiến trình?		Slide Ch4.4
	- Lợi điểm của tiểu trình là gì?	Ch4 p.162	Slide Ch4.7
	- Thư viện cần sử dụng là gì?	Ch4 p.169	Slide Ch4.22

Yêu cầu 1: Hoàn thành chương trình tính tổng các số nguyên từ 1 đến n, với n được truyền vào từ lời gọi thực thi. Việc tính tổng được giao cho một tiểu trình con.

Hướng dẫn: Tải về tập tin *LAB_IntroOS/CHAPTER_4/Fig4_11_MultithreadSum.c*, biên dịch và thực thi. Lưu ý lệnh biên dịch cần thêm chọn lựa “tiểu trình”.

```
$ gcc Fig4_11_MultithreadSum.c -lpthread
```

- Dòng 4 là thư viện pthread của POSIX
- Dòng 10 là biến toàn cục, các tiểu trình có thể chia sẻ biến số này.
- Dòng 11 là nguyên mẫu hàm sẽ giao cho (các) tiểu trình.
- Dòng 13 khai báo biến giữ ID của tiểu trình được tạo ra.
- Dòng 14 định nghĩa thuộc tính của tiểu trình, ở đây dùng thuộc tính mặc định.
- Dòng 15 là lời gọi tạo ra một tiểu trình với thuộc tính khai báo tại dòng 14, giao công việc là hàm khai báo nguyên mẫu ở dòng 11, truyền vào giá trị là chuỗi argv[1] và giao cho tid quản lý tiểu trình.
- Dòng 20 là tiến trình chờ đợi sự kết thúc của tiểu trình được quản lý bởi tid.
- Dòng 24 là hiện thực hàm con.
- Dòng 25 có hàm atoi() là chuyển một chuỗi thành giá trị nguyên.
- Dòng 29 là kết thúc tiểu trình.

4	#include<pthread.h>
10	int sum;
11	void * runner(void * param);
13	pthread_t tid; /* the thread identifier */
14	pthread_attr_t attr; /* set of thread attributes */
16	pthread_attr_init(& attr);
18	pthread_create(& tid, & attr, runner, argv[1]);

20	<code>pthread_join(tid, NULL);</code>
24	<code>void * runner(void * param) {</code>
25	<code>int i, upper = atoi(param);</code>
29	<code>pthread_exit(0);</code>

Yêu cầu 2: (Bài tập 4.19 sách [1]) Minh hoạ khả năng chia sẻ thông tin giữa tiểu trình và tiến trình.

Hướng dẫn: Tải về tập tin *LAB_IntroOS/CHAPTER_4/ Fig4_22_Fork_vs_Thread.c*, biên dịch và thực thi. Lưu ý lệnh biên dịch cần thêm chọn lựa “tiểu trình”.

```
$ gcc Fig4_22_Fork_vs_Thread.c -lpthread
```

- Kết quả in ra tại dòng 34 là bao nhiêu?

- Kết quả in ra tại dòng 38 là bao nhiêu?

- Từ hai kết quả vừa nêu, hãy cho biết hành vi fork() sẽ sao chép hay chia sẻ dữ liệu cho tiến trình con tạo ra; và câu lệnh nào đã thay đổi giá trị value đã in ra tại dòng 34.

Bài tập lập trình

1. (Bài tập 4.22) Viết chương trình đa luồng tính toán các giá trị thống kê khác nhau từ một danh sách các số được truyền vào thông qua đối số của dòng lệnh. Chương trình sau đó sẽ tạo ba tiểu trình tính toán riêng biệt. Một tiểu trình sẽ xác định trung bình cộng của các số, tiểu trình thứ hai sẽ xác định giá trị lớn nhất và tiểu trình thứ ba sẽ xác định giá trị nhỏ nhất. Ví dụ:

```
> ./bai22.out 90 81 78 95 79 72 85
```

```
Gia tri trung binh: 82
```

```
Gia tri lon nhat: 95
```

```
Gia tri nho nhat: 72
```

Các biến số đại diện cho các giá trị trung bình, nhỏ nhất và lớn nhất sẽ được lưu trữ trên toàn cục. Các tiểu trình sẽ tính toán các giá trị này và tiến trình cha sẽ xuất ra các giá trị kết quả khi tiểu trình kết thúc.

Mở rộng: có thể tạo thêm các tiểu trình để tính giá trị trung vị; độ lệch chuẩn; sắp xếp dãy số, ...

Gợi ý cho bài tập này tại ***LAB_IntroOS/LAB_4/ex1_Pseudocode.c***

2. (Bài tập 4.23) Viết chương trình đa luồng để xuất ra số nguyên tố. Người dùng chạy chương trình và nhập vào một số nguyên thông qua đối số tại dòng lệnh. Chương trình sau đó sẽ tạo ra một tiến trình riêng biệt xuất ra tất cả các số nguyên tố nhỏ hơn hoặc bằng số được nhập bởi người dùng.

Gợi ý cho bài tập này tại ***LAB_IntroOS/LAB_4/ex2_Pseudocode.c***