**BÁO CÁO THỰC TẬP**

Vấn đề đặt ra : Tìm hiểu luồng trong C (pthread), xây dụng các luồng để lấy được thời gian hiện tại của hệ thống có độ chính xác nanoseconds , lưu chênh lệch giữa hai lần lấy mẫu

Giải quyết vấn đề: Tạo ra 3 luồng chính là :

+ INPUT : chức năng là đọc chu kì lấy mẫu T

+ SAMPLE dùng để lấy ra thời gian hệ thống

+ LOGGING : tính offset giữa hai lần lấy mẫu và ghi kết quả vào file time\_and\_interval.txt

Chi tiết:

* Sử dụng hàm clock\_gettime() để lấy thời gian hệ thống hiện tại.
* Sử dụng hàm clock\_nanosleep() để thực hiện chu kì T cho mỗi lần lấy mẫu (100ns đến 1000000ns ).

Kết quả đạt được:

* Thu được kết quả tính offset sau mỗi chu kỳ như sau:
  + Với chu kỳ 1000000ns :
    - offset đạt được: 0.001035989(s).
  + Với chu kỳ 100000ns :
    - offset đạt được:: 0.000108783(s).
  + Với chu kỳ 10000ns :
    - offset đạt được:: 0.0000026167(s).
  + Với chu kỳ 1000ns :
    - offset đạt được:: 0. 0000004251(s).
  + Với chu kỳ 100ns :
    - offset đạt được:: 0. 0000005703(s).

Kết luận:

+ Đối với chu kỳ lấy mẫu T trong khoảng 1000000ns đến 100000ns, kết quả offset tương đối gần với chu kì lấy mẫu.

+ Đối với chu kỳ 10000 ns 1000ns và 100ns, kết quả đã không còn được chính xác

Khó khăn:

* Chưa thuần thục cách sử dụng khóa mutex, và biến conditional variable trong threads
* Kết quả thu được chưa đạt được yêu cầu mong muốn

Các lỗi đã gặp phải:

+ Chưa tạo được thứ tự thực hiện luồng: input → sample → logging

+ Sử dụng hàm nanosleep() để tạo chu kì gấy ra tốn kém tài nguyên của CPU, làm delay chương trình.

Khắc phục: Sử dụng khóa Mutex, và conditional variable trong threads để thực hiện các thread một cách hiệu quả.

+ Thay thế hàm nanosleep() bằng hàm clock\_nanosleep().