|  |
| --- |
| **Họ tên: Lê Thị Thu Hiền**  **MSSV: 21522059**  **Lớp: PMCL2021.2**  **Môn: IT007 Giảng viên: Lầu Phi Tường** |

**Đề**

Một biến X được chia sẻ bởi 2 tiến trình cùng thực hiện đoạn code sau :

do

  X = X +1;

  if ( X == 20) X = 0;

while ( TRUE );

1. Bắt đầu với giá trị X = 0, chứng tỏ rằng giá trị X có thể vượt quá 20. Cần chỉnh sửa đoạn chương trình trên như thế nào để đảm bảo X không vượt quá 20? Gợi ý sử dụng giải pháp đồng bộ (busy waiting, semaphore, v.v.) để giải quyết (8đ)

2. Viết chương trình đã chỉnh sửa sử dụng thư viện pthread của Linux để tạo 2 tiểu trình (2đ). Chụp hình kết quả chạy trên Ubuntu

**Bài làm**

**Bài 1:**

Do

{

X = X +1;

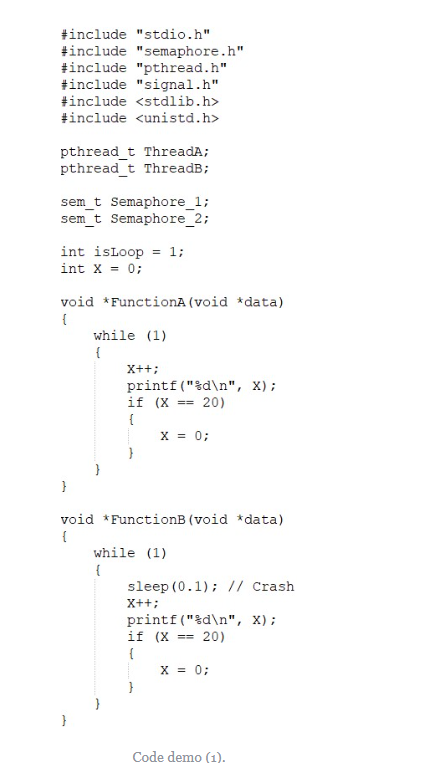
if ( X == 20) X = 0;

}while ( TRUE );

Do X được chia sẻ chung ở cả hai tiến trình và chỉ bị reset khi X == 20 nên X có thể vượt quá 20 khi:

- Giả định tiến trình 2 dừng lại trong khi 2 tiến trình đang hoạt động. Sau đó tại tiến trình 1, khi X = 19 thì tiến trình 2 được release đúng ngay đoạn X = X + 1, và cộng dồn với X = 20 sau lệnh X = X + 1 của tiến trình 1 dẫn tới X vượt quá 20 và còn bị cộng tới vô cùng.

\*Đây là code demo với cách giả định P2 bị crash và release bằng một lệnh sleep nhỏ, kết quả của X sẽ vượt quá 20 trong tích tắc



Text, letter

Description automatically generated

- Để giải quyết vấn đề này ( làm cho X luôn nhỏ hơn 20 ) thì ta có thẻ giải quyết bằng cách đưa 2 tiến trình về làm 1, tức là làm cho cùng một lúc chỉ có một trong hai tiến trình được chạy bằng một biến Semaphore khởi tạo bằng 1.

**Bài 2:**

Text

Description automatically generated