**BÁO CÁO THỰC HÀNH LAB3**

**MÔN: HỆ ĐIỀU HÀNH**

**LỚP: IT007.O212.2**

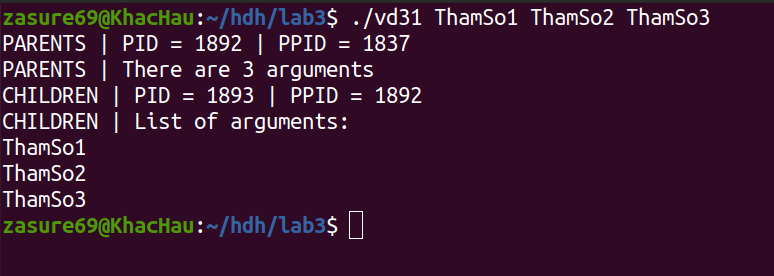
Họ và tên: Nguyễn Khắc Hậu

MSSV: 22520410

**Bài 3.5: Bài tập thực hành**

**1. Thực hiện Ví dụ 3-1, Ví dụ 3-2, Ví dụ 3-3, Ví dụ 3-4 giải thích code và kết quả nhận được?**

**Ví dụ 3-1**:

****

Tiến trình gọi hàm fork() được gọi là tiến trình cha, tiến trình mới được tạo ra là tiến trình con. Tiến trình cha quay lại việc thực thi và tiến trình con bắt đầu thực thi tại cùng một nơi (nơi mà fork() trả về). Và 2 tiến trình thực thi đồng thời với nhau.

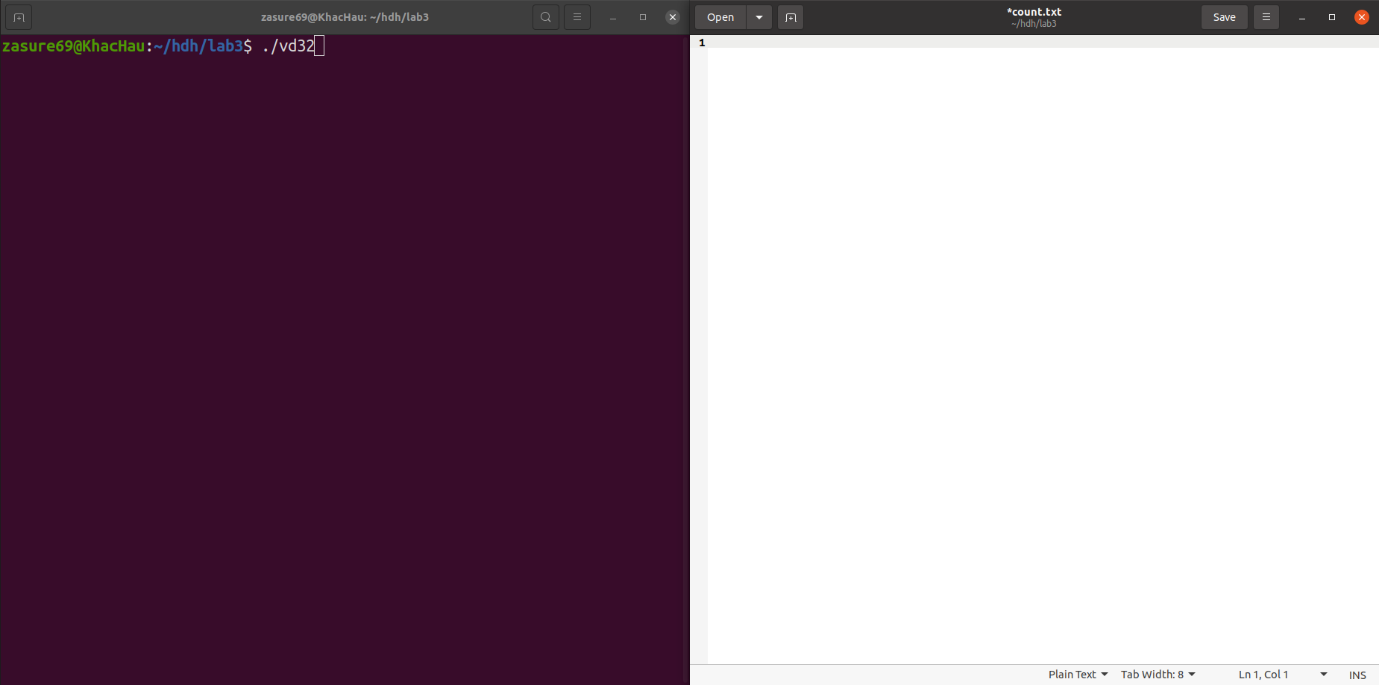
Hàm fork() trả về 0 nếu đang trong tiến trình con và trả về > 0 nếu đang trong tiến trình cha.

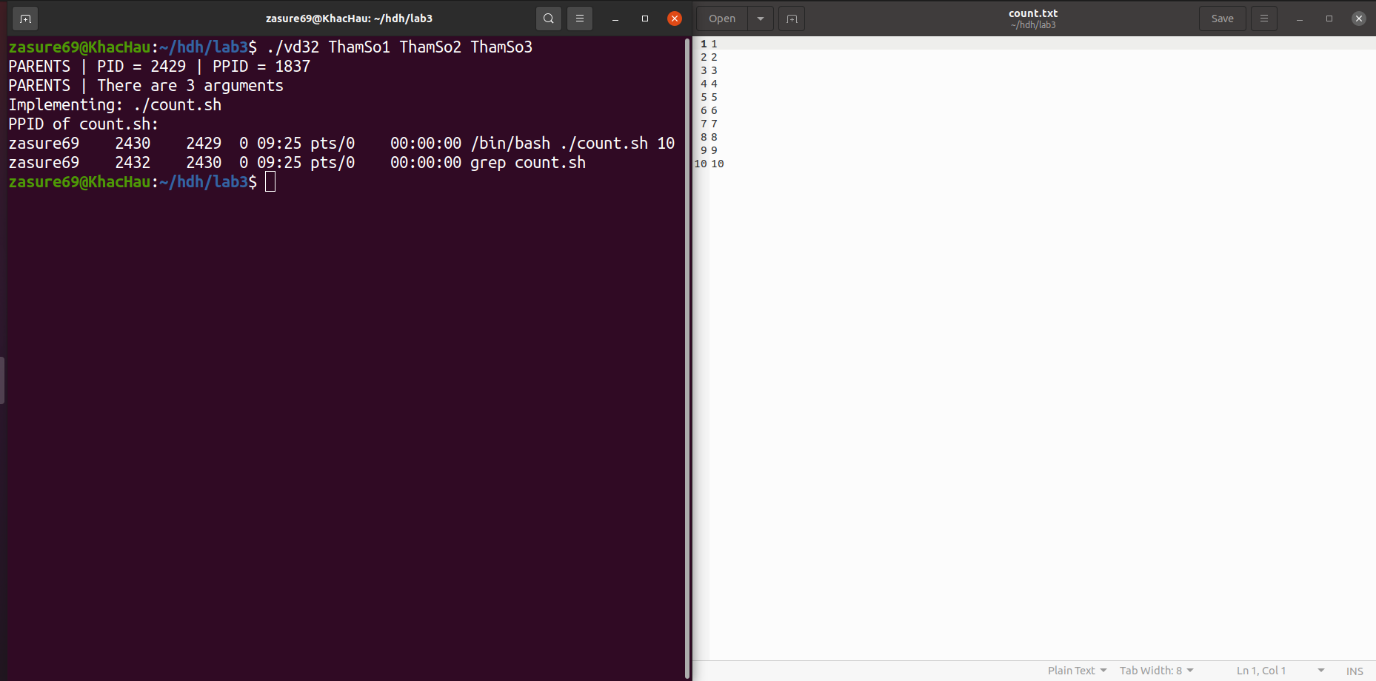
Tiến trình cha sẽ thực thi đoạn lệnh if (pid > 0) 🡪 in ra PID và PPID của tiến trình cha 🡪 in ra số lượng tham số truyền khi thực hiện lệnh chạy chương trình.

Tiến trình con sẽ thực thi đoạn lệnh trong if (pid == 0) 🡪 in PID và PPID của tiến trình con 🡪 in ra danh sách tham số truyền vào khi thực hiện lệnh chạy chương trình.

Và vì 2 tiến trình thực thi đồng thời nên kết quả 2 đoạn lệnh đều được in ra.

**Ví dụ 3-2**:



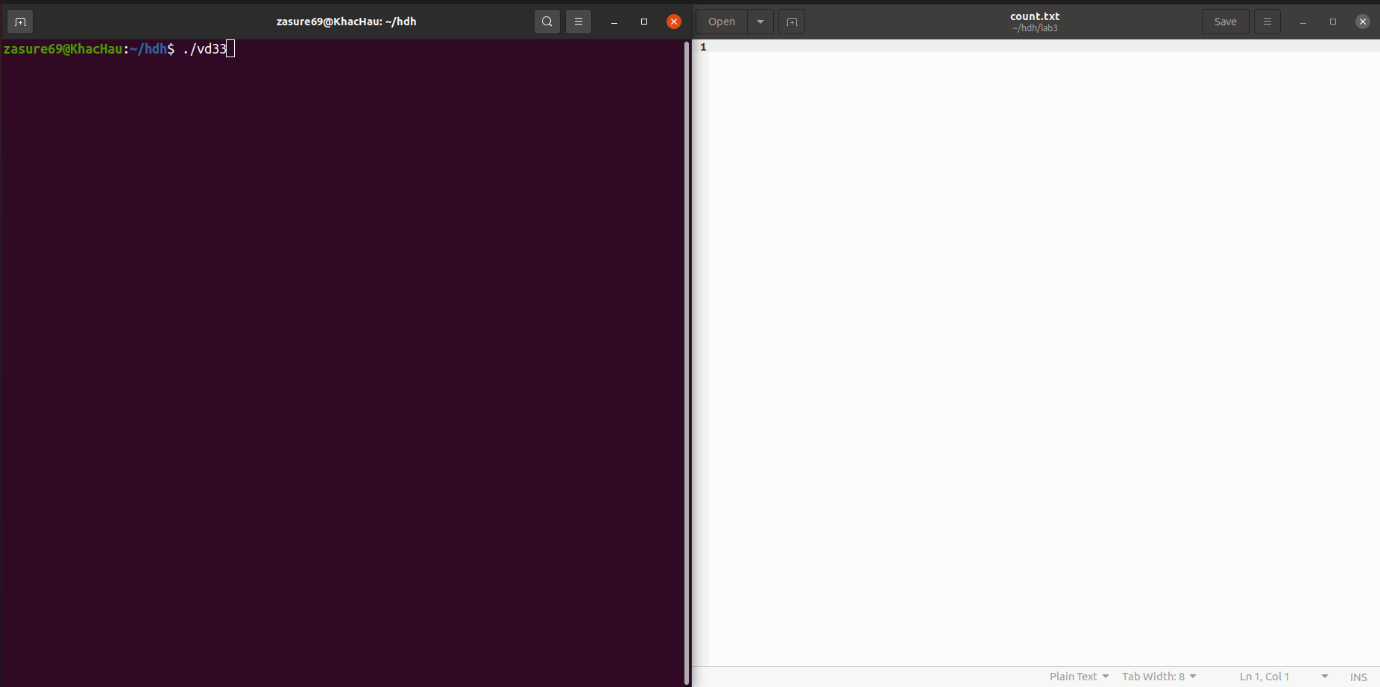


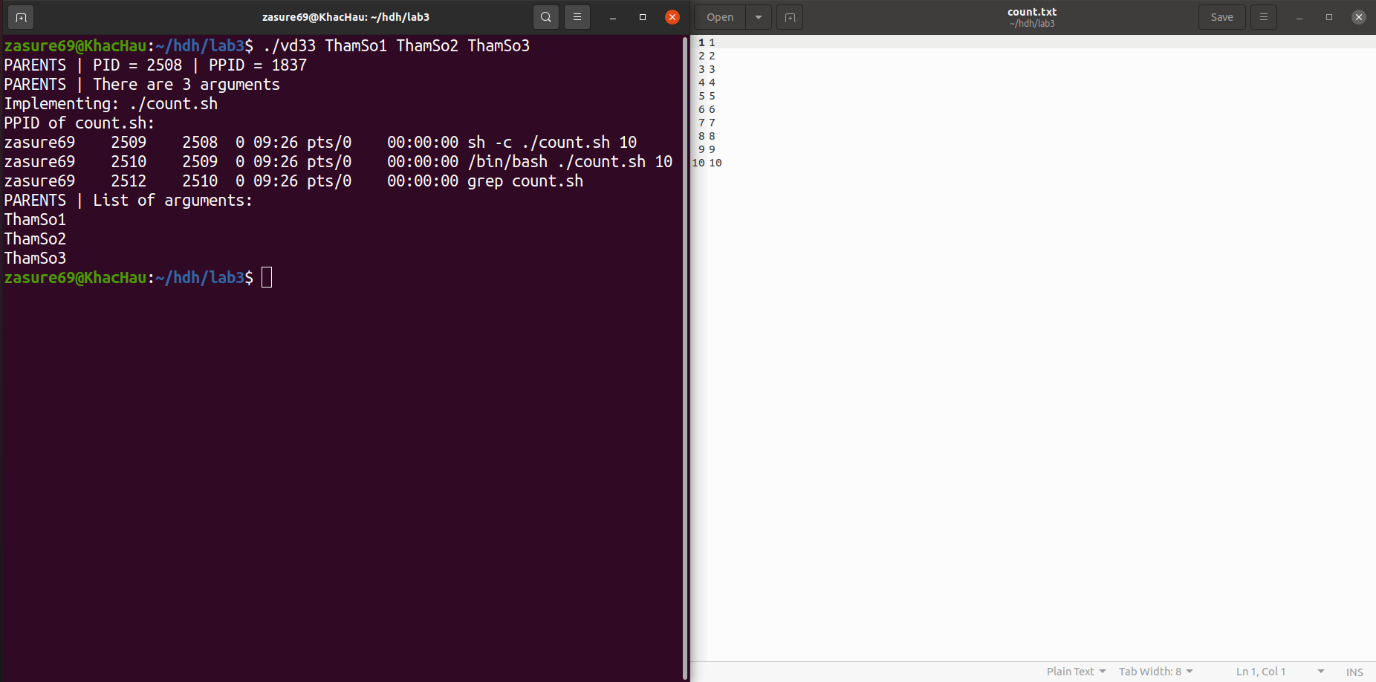
Hàm fork() tạo ra 2 tiến trình cha và con thực thi đồng thời.

Tiến trình cha sẽ thực thi đoạn lệnh if (pid > 0) 🡪 in ra ra PID và PPID của tiến trình cha và in ra số lượng tham số truyền vào khi thực hiện lệnh chạy chương trình.

Tiến trình con sẽ thực thi đoạn lệnh if (pid == 0) 🡪 hàm execl() sẽ thay thế tiến trình hiện tại bằng các nạp chương trình thực thi count.sh (count.sh sẽ đếm từ 1 – 10 và ghi vào file count.txt) tới không gian địa chỉ của nó và sau đó tiến trình con sẽ tự hủy 🡪 sẽ không thực thi đoạn lệnh bên dưới lệnh execl (đoạn lệnh in ra PID và PPID của tiến trình con, và in ra danh sách tham số truyền vào) trong tiến trình con.

**Ví dụ 3-3**:



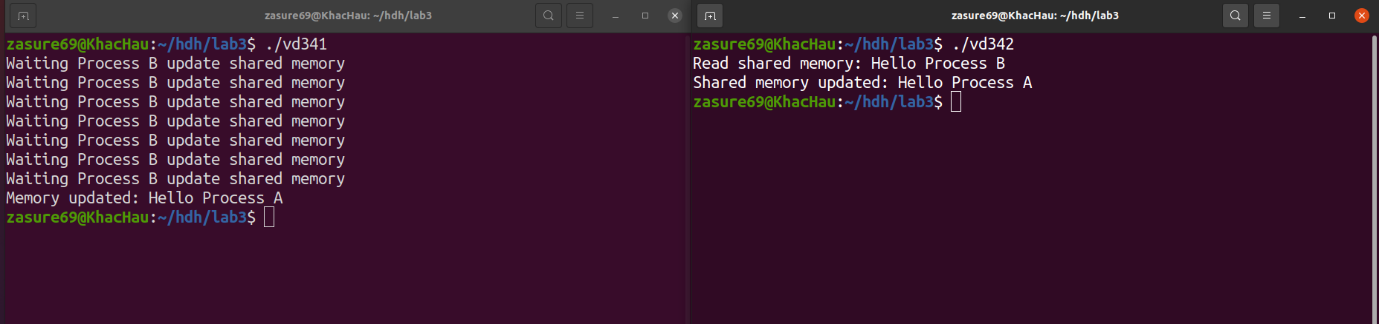


Hàm fork() tạo ra 2 tiến trình cha và con thực thi đồng thời.

Tiến trình cha sẽ thực thi đoạn lệnh if (pid > 0) 🡪 in ra ra PID và PPID của tiến trình cha và in ra số lượng tham số truyền vào khi thực hiện lệnh chạy chương trình.

Tiến trình con sẽ thực thi đoạn lệnh if (pid == 0) 🡪 hàm system sẽ tạo ra một tiến trình thực thi lệnh count.sh độc lập và không thay thế tiến trình con 🡪 lệnh trong hàm system được thực thi trong khi đoạn lệnh dưới hàm system() vẫn đang thực thi 🡪 kết quả 2 tiến trình đều được in ra.

**Ví dụ 3-4**:



Process A sẽ khởi tạo bộ nhớ chia sẻ và ghi vào bộ nhớ “Hello Process B”

🡪 Khi Process B truy cập bộ nhớ chia sẻ, đọc bộ nhớ và in ra “Hello Process B”

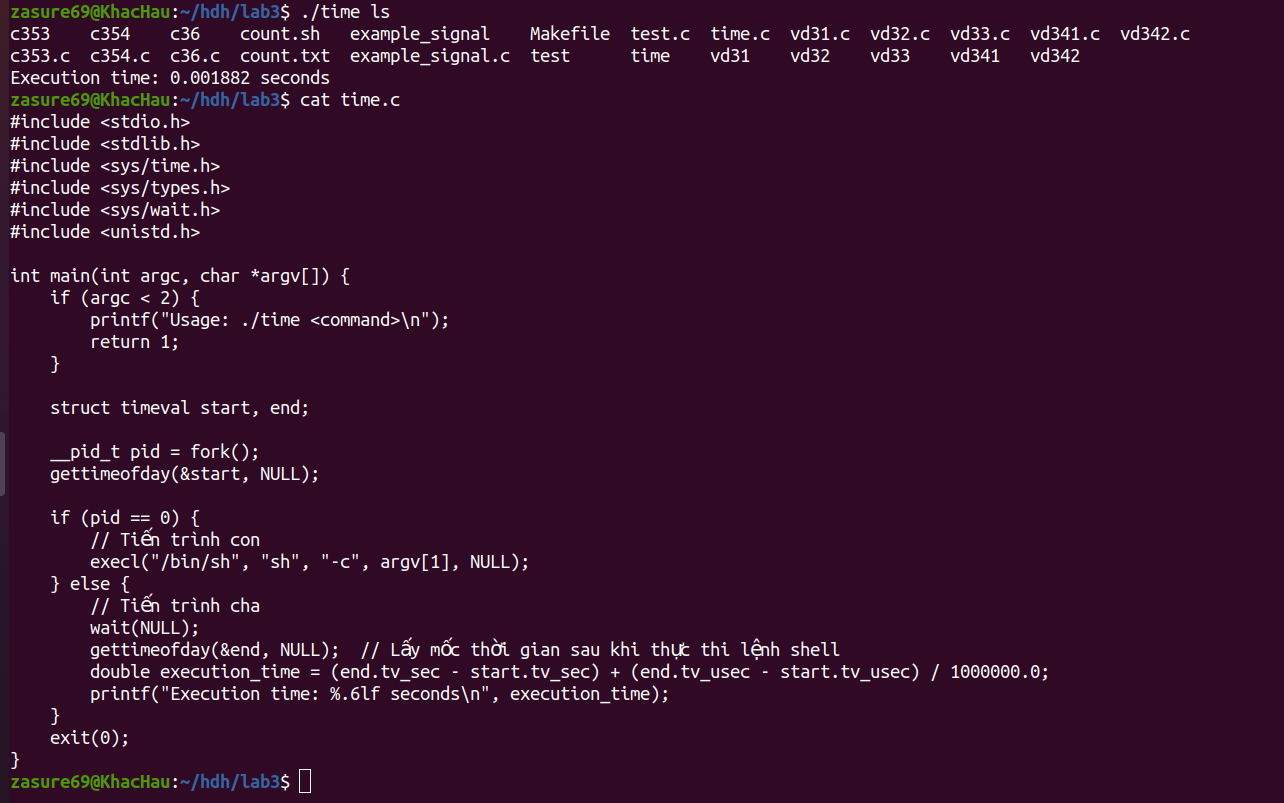
Sau khi ghi vào bộ nhớ “Hello Process B”, Process A sẽ đợi Process B cập nhật bộ nhớ bằng vòng lặp với điều kiện strncmp(ptr, "Hello Process B", 15) == 0

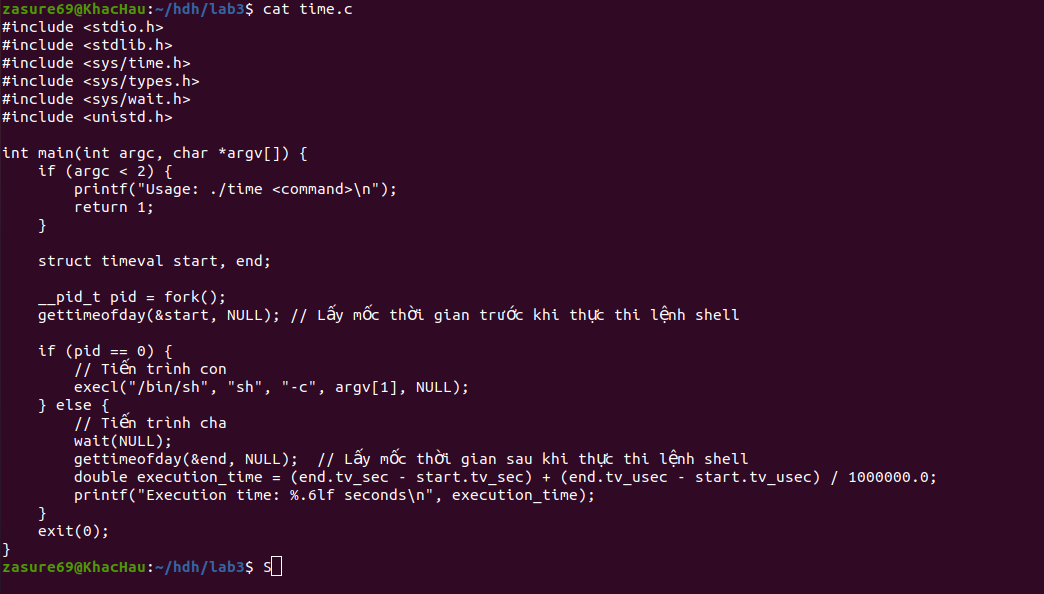
có nghĩa là sẽ so sánh chuỗi trong bộ nhớ chia sẻ nếu không còn bằng “Hello Process B” nữa thì sẽ thoát vòng lặp .

Process B sau khi đọc nội dung trong bộ nhớ chia sẻ sẽ ghi vào bộ nhớ “Hello Process A” và thoát khỏi bộ nhớ chia sẻ.

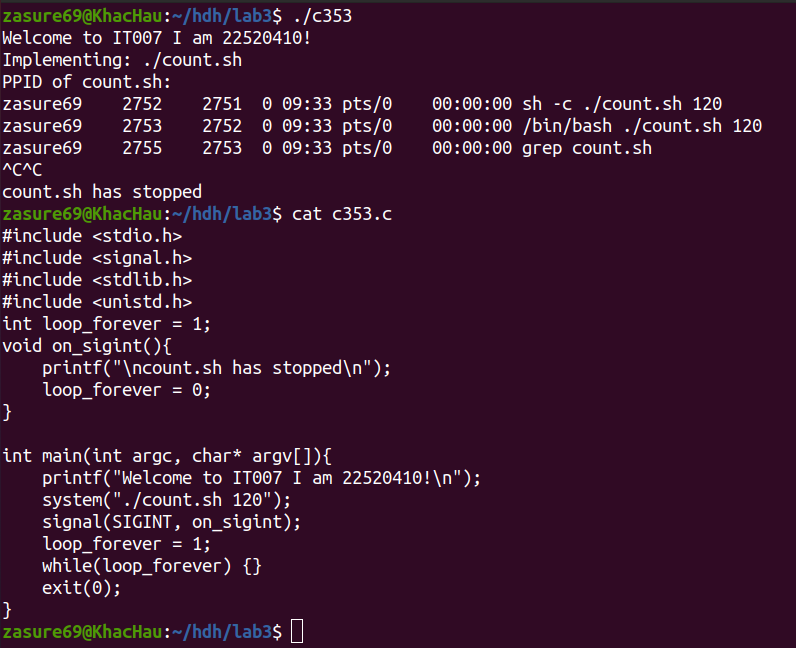
Sau khi Process A nhận thấy nội dung đã được cập nhật 🡪 strncmp(ptr, "Hello Process B", 15) trả về 1 🡪 thoát vòng lặp và in ra nội dung cập nhật “Hello Process A ”🡪 thu hồi bộ nhớ chia sẻ.

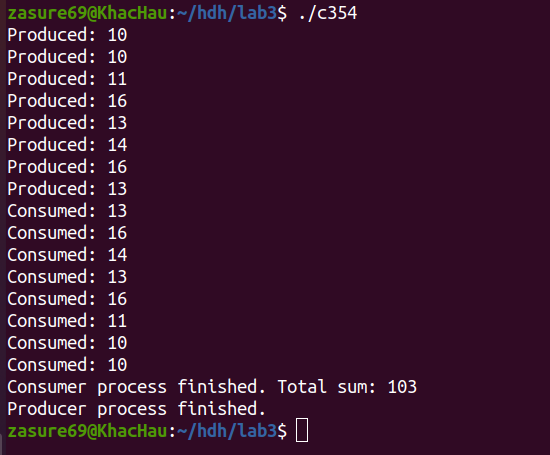
**2. Viết chương trình time.c thực hiện đo thời gian thực thi của một lệnh shell. Chương trình sẽ được chạy với cú pháp "./time <command>" với <command> là lệnh shell muốn đo thời gian thực thi.**

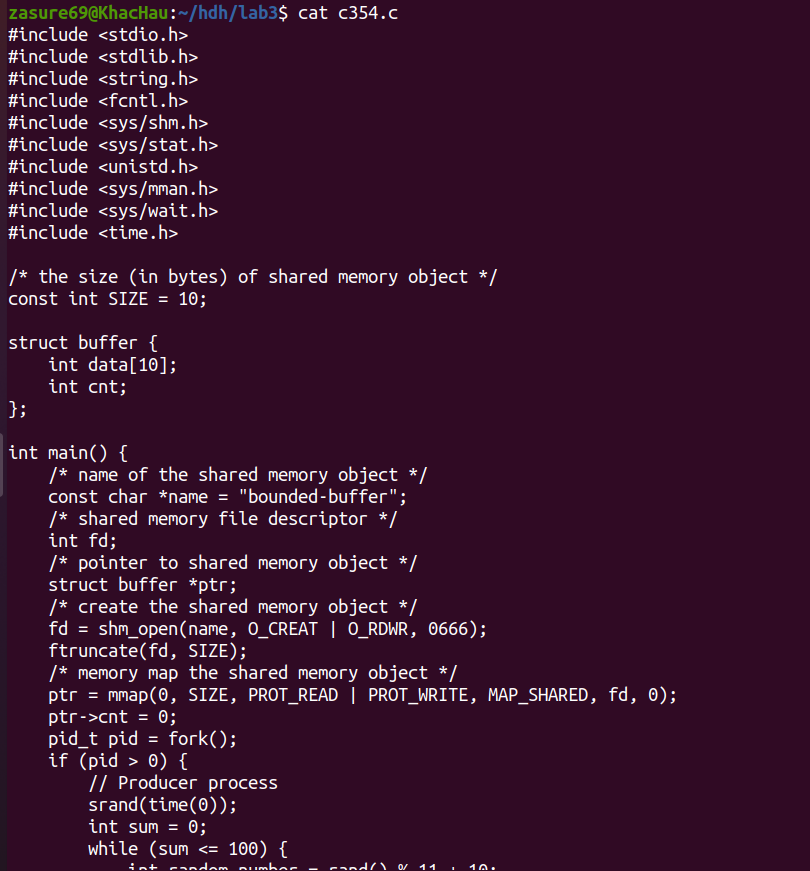


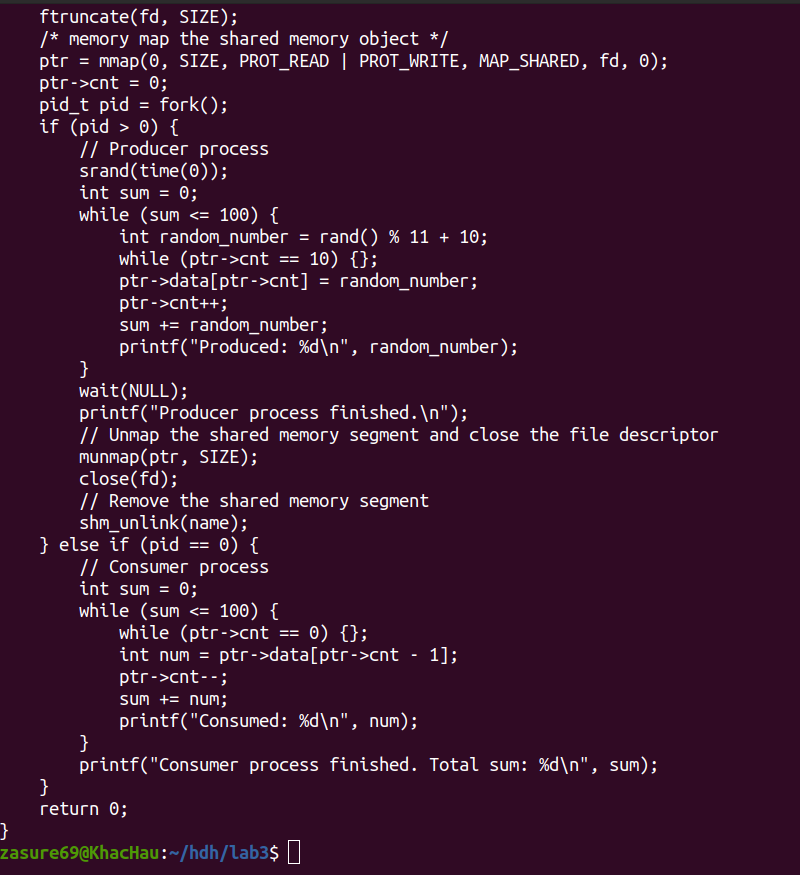
****

**3.**  **Viết một chương trình làm bốn công việc sau theo thứ tự:  
● In ra dòng chữ: “Welcome to IT007, I am <your\_Student\_ID>!”  
● Thực thi file script count.sh với số lần đếm là 120  
● Trước khi count.sh đếm đến 120, bấm CTRL+C để dừng tiến trình này  
● Khi người dùng nhấn CTRL+C thì in ra dòng chữ: “count.sh has stopped”**

**4. Viết chương trình mô phỏng bài toán Producer – Consumer như sau:  
● Sử dụng kỹ thuật shared-memory để tạo một bounded-buffer có độ lớn là 10 bytes.  
● Tiến trình cha đóng vai trò là Producer, tạo một số ngẫu nhiên trong khoảng [10, 20] và ghi dữ liệu vào buffer  
● Tiến trình con đóng vai trò là Consumer đọc dữ liệu từ buffer, in ra màn hình và tính tổng  
● Khi tổng lớn hơn 100 thì cả 2 dừng lại**

****

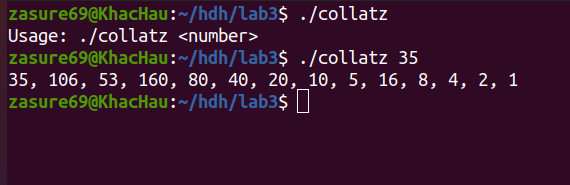
****

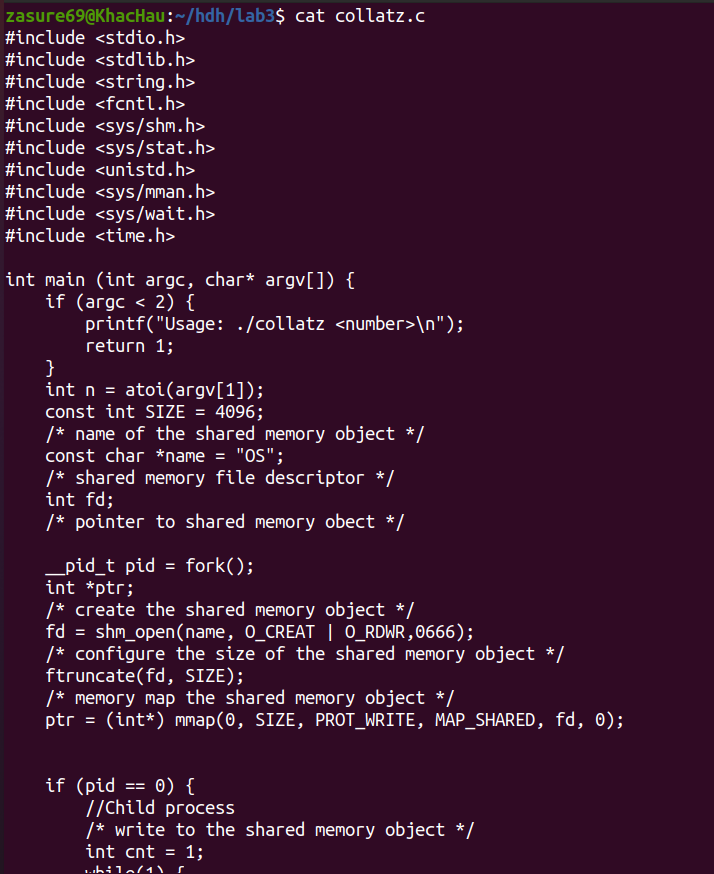
****

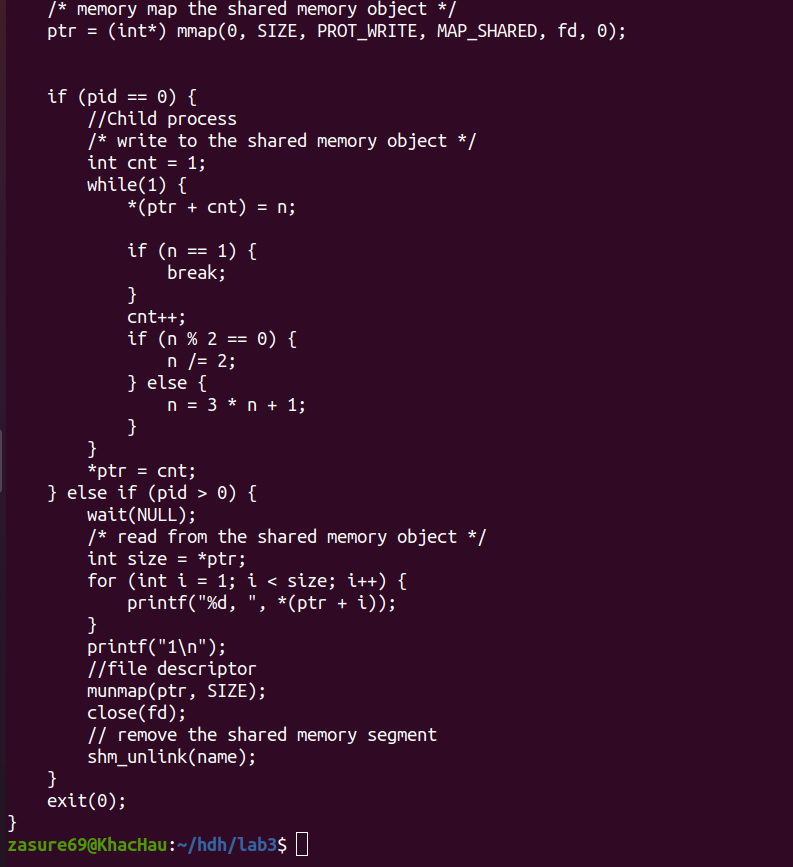
**3.6 Bài tập ôn tập**

Ảnh có chứa văn bản, Phông chữ, ảnh chụp màn hình, số

Mô tả được tạo tự động

****

****

****