|  |
| --- |
| Họ và tên:  Mã số sinh viên:  Lớp: |

HỆ ĐIỀU HÀNH  
BÁO CÁO LAB 5

**CHECKLIST**

**5.5. BÀI TẬP THỰC HÀNH**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **BT 1** | **BT 2** | **BT 3** | **BT 4** |
| **Trình bày cách làm** |  |  |  |  |
| **Chụp hình minh chứng** |  |  |  |  |
| **Giải thích kết quả** |  |  |  |  |

**5.6. BÀI TẬP ÔN TẬP**

|  |  |
| --- | --- |
|  | **BT 1** |
| **Trình bày cách làm** |  |
| **Chụp hình minh chứng** |  |
| **Giải thích kết quả** |  |

**Tự chấm điểm:**

*\*Lưu ý: Xuất báo cáo theo định dạng PDF, đặt tên theo cú pháp:* ***<Tên nhóm>\_LAB5.pdf***

**5.5. BÀI TẬP THỰC HÀNH**

# Hiện thực hóa mô hình trong ví dụ 5.3.1.2, tuy nhiên thay bằng điều kiện sau: sells <= products <= sells + [4 số cuối của MSSV]

Trả lời...

# Cho một mảng a được khai báo như một mảng số nguyên có thể chứa n phần tử, a được khai báo như một biến toàn cục. Viết chương trình bao gồm 2 thread chạy song song:

* Một thread làm nhiệm vụ sinh ra một số nguyên ngẫu nhiên sau đó bỏ vào a. Sau đó đếm và xuất ra số phần tử của a có được ngay sau khi thêm vào.
* Thread còn lại lấy ra một phần tử trong a (phần tử bất kỳ, phụ thuộc vào người lập trình). Sau đó đếm và xuất ra số phần tử của a có được ngay sau khi lấy ra, nếu không có phần tử nào trong a thì xuất ra màn hình “Nothing in array a”.

Chạy thử và tìm ra lỗi khi chạy chương trình trên khi chưa được đồng bộ. Thực hiện đồng bộ hóa với semaphore.

Trả lời...

# Cho 2 process A và B chạy song song như sau:

|  |  |
| --- | --- |
| int x = 0; | |
| **PROCESS A** | **PROCESS B** |
| processA()  {  while(1){  x = x + 1;  if (x == 20)  x = 0;  print(x);  }  } | processB()  {  while(1){  x = x + 1;  if (x == 20)  x = 0;  print(x);  }  } |

Hiện thực mô hình trên C trong hệ điều hành Linux và nhận xét kết quả.

Trả lời…

# Đồng bộ với mutex để sửa lỗi bất hợp lý trong kết quả của mô hình Bài 3.

Trả lời…

**5.6. BÀI TẬP ÔN TẬP**

# Biến ans được tính từ các biến x1, x2, x3, x4, x5, x6 như sau:

w = x1 \* x2; (a)

v = x3 \* x4; (b)

y = v \* x5; (c)

z = v \* x6; (d)

y = w \* y; (e)

z = w \* z; (f)

ans = y + z; (g)

Giả sử các lệnh từ (a) 🡪 (g) nằm trên các thread chạy song song với nhau. Hãy lập trình mô phỏng và đồng bộ trên C trong hệ điều hành Linux theo thứ tự sau:

* (c), (d) chỉ được thực hiện sau khi v được tính
* (e) chỉ được thực hiện sau khi w và y được tính
* (g) chỉ được thực hiện sau khi y và z được tính

Trả lời...