

TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
THÀNH PHỐ HỒ CHÍ MINH
KHOA CÔNG NGHỆ THÔNG TIN



Học phần: HỆ ĐIỀU HÀNH

BÁO CÁO ĐỒ ÁN

Tên đồ án: Đa chương
Lớp: 20CLC11
Giảng viên: Phạm Tuấn Sơn

MỤC LỤC

I. Thông tin nhóm.....	3
II. Nội dung đồ án HĐH NachOS.....	3
1) Cài đặt System Call SpaceId Exec (char* name).....	3
2) Phân trang	4
3) Cài đặt void StartProcess_2(int id).....	4
III. Nguồn tham khảo	4

HỆ ĐIỀU HÀNH

Đại học Khoa học Tự nhiên TpHCM
Khoa Công nghệ thông tin (Chương trình chất lượng cao)

I. Thông tin nhóm

MSSV	Họ và tên	Email	Chức vụ
20127610	Trương Samuel	20127610@student.hcmus.edu.vn	Nhóm trưởng
20127001	Hà Quốc Anh	20127001@student.hcmus.edu.vn	Thành viên
20127298	Nguyễn Trần Minh Quang	20127298@student.hcmus.edu.vn	Thành viên

II. Nội dung đồ án HĐH NachOS

1) Cài đặt System Call SpaceId Exec (char* name):

- Bước 1: Đọc địa chỉ tham số name từ thanh ghi r4.
- Bước 2: Sao chép giá trị r4 từ user space sang system space bằng hàm User2System() và gán vào biến "char* filename".
- Bước 3: Kiểm tra nếu filename = NULL thì báo lỗi cho user và trả về -1 vào thanh ghi r2, rồi break chương trình.
- Bước 4: Nếu bước 3 không lỗi, dò tìm ô còn trống (biến int freeSlot) trong bảng đặc tả file.
- Bước 5: Nếu vị trí i chưa có file, gán biến freeslot = i.
- Bước 6: Mở file đọc và ghi hoặc file chỉ đọc tại vị trí i.
- Bước 7: Kiểm tra nếu file không tồn tại thì báo lỗi cho user, trả về trừ -1 vào thanh ghi r2 và giải phóng bộ nhớ biến filename.
- Bước 8: Nếu tại vị trí i có một file nào đó, kiểm tra xem file đã mở chưa, nếu chưa thì gán freeSlot = i và chuyển đến Bước 10.
- Bước 9: Nếu không tìm được ô trống trong bảng đặc tả file, thông báo lỗi cho user và trả về -1 vào thanh ghi r2.
- Bước 10: Nếu đã tìm được ô trống, tạo thread mới và chạy thread đó.
- Bước 11: Trả biến freeSlot vào thanh ghi r2, giải phóng bộ nhớ biến filename.

HỆ ĐIỀU HÀNH

Đại học Khoa học Tự nhiên TpHCM
Khoa Công nghệ thông tin (Chương trình chất lượng cao)

2) Phân trang:

- Bước 1: Trong phần set up the translation, chỉnh sửa `pageTable[i].physicalPage = i` thành `pageTable[i].physicalPage = pages→Find()`: Tìm trang còn trống bằng cách sử dụng phương thức `Find()` của lớp `Bitmap` thông qua biến toàn cục `pages`.
- Bước 2: Thêm vòng `for` chạy từ 0 đến `numPages` bọc ngoài 2 câu điều kiện `if (noffH.code.size > 0)` và `if (noffH.initData.size > 0)` để duyệt từng trang của tiến trình.
- Bước 3: Đọc `n` bytes từ địa chỉ trang đang giữ vào bộ nhớ chính (`n` là kích thước trang). Chỉnh `executable→ReadAt`. Mỗi trang thứ `i` sẽ được nạp vào trong bộ nhớ tại khung trang thứ `k` được cấp phát tại `pageTable[i].physicalPage`. Vì vậy, để chuyển từ đơn chương sang đa chương, ta phải chỉnh sửa như sau:

```
executable→ReadAt(&(machine→mainMemory[noffH.code.virtualAddr]) +  
pageTable[i].physicalPage * PageSize, PageSize, noffH.code.inFileAddr + i * PageSize);
```

3) Cài đặt void StartProcess 2 (int id):

Cài đặt giống hàm `StartProcess` có sẵn nhưng ta đổi lại tham số đầu vào, từ `char* filename` → `int id`. Dùng `id` để lấy file từ bảng mô tả file.

III. Nguồn tham khảo

[Lập trình Nachos HCMUS - YouTube](#)

<https://github.com/nguyenthanhchungfit/Nachos-Programing-HCMUS>