ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH

ĐẠI HỌC KHOA HỌC TỰ NHIÊN

KHOA CÔNG NGHỆ THÔNG TIN



HỆ ĐIỀU HÀNH

ĐỒ ÁN 1: SYSTEM CALL

*Giảng viên hướng dẫn:* **thầy Phạm Tuấn Sơn**

*Nhóm thực hiện*: **18120078 – Ngô Phù Hữu Đại Sơn**

**18120211 – Võ Thế Minh**

**18120227 – Phạm Văn Minh Phương**

*Lớp*: **HĐH 18\_4**

*Khóa*: **2018**

MỤC LỤC

[A. Thông tin khái quát 2](#_Toc55056361)

[I. Thông tin nhóm 2](#_Toc55056362)

[II. Bảng phân công công việc 2](#_Toc55056363)

[B. Nội dung 3](#_Toc55056364)

[I. Môi trường phát triển 3](#_Toc55056365)

[II. Mục tiêu của đồ án 3](#_Toc55056366)

[III. Ý tưởng thiết kế 3](#_Toc55056367)

[IV. Triển khai 4](#_Toc55056368)

[V. Kiểm thử 5](#_Toc55056369)

[C. TỔNG KẾT 6](#_Toc55056370)

[VI. Đánh giá đồ án 6](#_Toc55056371)

[1. Mức độ hoàn thành của các thành viên 6](#_Toc55056372)

[2. Mức độ hoàn thành đồ án: 6](#_Toc55056373)

[VII. Nguồn tham khảo 6](#_Toc55056374)

# Thông tin khái quát

## I. Thông tin nhóm

|  |  |  |
| --- | --- | --- |
| MSSV | HỌ TÊN | VAI TRÒ |
| 18120211 | Võ Thế Minh | Trưởng nhóm |
| 18120078 | Ngô Phù Hữu Đại Sơn | Thành viên |
| 18120227 | Phạm Văn Minh Phương | Thành viên |

## II. Bảng phân công công việc

|  |  |
| --- | --- |
| MSSV | CÔNG VIỆC PHỤ TRÁCH |
|  | Cài đặt nachOS |
|  | Tái cấu trúc lớp FileSystem |
|  | Tái cấu trúc lớp OpenFile |
|  | Cài đặt + kiểm thử system call Create |
|  | Cài đặt + kiểm thử system call Open |
|  | Cài đặt + kiểm thử system call Read |
|  | Cài đặt + kiểm thử system call Seek |
|  | Cài đặt + kiểm thử system call Write |
|  | Cài đặt + kiểm thử system call Close |
|  | Cài đặt + kiểm thử chương trình Echo |
|  | Cài đặt + kiểm thử chương trình Cat |
|  | Cài đặt + kiểm thử chương trình Copy |

# Nội dung

## Môi trường phát triển

* *Máy ảo*: nachOS 3.4
* *Hệ điều hành:* Ubuntu 14.04-i386
* *Compiler*: gcc - 3.4.6
* *Cross-compiler*: gcc - 2.95.3
* *Binutils*: 2.11.2

## Mục tiêu của đồ án

1. Tìm hiểu cách cài đặt hệ điều hành giả lập nashOS
2. Tìm hiểu cách thức giao tiếp giữa HĐH nashOS và chương trình người dùng
3. Tìm hiểu cách viết các system call cho HĐH nashOS.

## Ý tưởng thiết kế

* 1. **Chuyển đổi dữ liệu liệu giữa System space và User space**
  2. **Từ System space đến User space**

Sử dụng hàm WriteMem (có chức năng ghi 1, 2 hoặc 4 bytes từ System space sang User space) đã được định nghĩa sẵn của lớp Machine, xây dựng phương thức public System2User cho lớp Machine để chép dữ liệu từ 1 vùng nhớ ở System space sang User space

* 1. **Từ User space đến System space**

Sử dụng hàm ReadMem (có chức năng đọc 1, 2 hoặc 4 bytes từ User space sang System space) đã được định nghĩa sẵn của lớp Machine, xây dựng phương thức public User2System cho lớp Machine để chép dữ liệu từ 1 vùng nhớ ở User space sang System space

* 1. **Chén lớp SynchConsole và nashOS để đọc ghi từ thiết bị xuất nhật chuẩn**

HĐH nashOS không cung cấp cho ta các phương thức xuất nhập từ thiết bị xuất/nhập chuẩn nên ta cần phải chèn lớp SynchConsole vào nachos để xây dựng được các system call có sử dụng dữ liệu từ console. Đồng thời cũng dung để xuất các thông báo cho chương trình người dung.

* 1. **Tái cấu trúc lớp FileSystem**

HĐH nashOS ban đầu chỉ cung cấp cho ta lớp FileSystem quản lý việc nhập/xuất của 1 file tại 1 thời điểm. Vì vậy cần tái cấu trúc lại lớp này để có thể quản lý nhiều file hơn tại 1 thời điểm. Sử dụng 1 mảng để quản lý các file đang mở (quản lý tối đa 10 file trong cùng 1 thời điểm). 2 phần tử đầu tiên của mảng này dung để quản lý thiết bị xuất nhập chuẩn.

* 1. **Tái cấu trúc lớp OpenFile**

Lớp OpenFile của nachos mặc định khi mở file thì luôn có thể dùng để đọc và ghi. Ta cần dùng 1 thuộc tính type để phân biệt các file mở chỉ để đọc, có khả năng đọc và ghi, thiết bị nhập chuẩn, thiết bị xuất chuẩn.

* 1. **Xây dựng system call Create**
  2. **Xây dựng system call Open**
  3. **Xây dựng system call Close**
  4. **Xây dựng system call Read**
* Dùng function openFileId đã được định nghĩa trong filesys để mở và kiểm tra tính đúng đắn của file. Trường hợp file không tồn tại, id file không hợp lệ và file có kiểu là stdout thì đọc file thất bại trả về -1.
* Trường hợp đọc file stdin thì dùng phương thức System2User đã được định nghĩa trong lớp machine để chuyển dữ liệu từ System->User, đọc file thành công trả về số byte đọc được.
* Trường hợp đọc file rỗng trả về -2.
  1. **Xây dựng system call Write**
* Tương tự như syscall Read dùng openFileId để mở file và kiểm tra tính đúng đắn của file. Trường hợp file không tồn tại, id file không hợp lệ và file có kiểu là stdin thì đọc thất bại trả về -1
* Trường hợp đọc file stdout thì dùng phương thức User2System để chuyển dữ liệu User->System, ghi file thành công trả về số byte thực sự ghi được
* Trường hợp đọc file read&write thì trả về số byte thực sự của file đó
* Trường hợp ghi file rỗng trả về -2
  1. **Xây dựng system call Seek**
  2. **Xây dựng chương trình echo**
  3. **Xây dựng chương trình cat**
  4. **Xây dựng chương trình copy**

## Triển khai

* 1. **User2System**
  2. **System2User**
  3. **Tái cấu trúc FileSystem**
  4. **Tái cấu trúc OpenFile**
  5. **Xây dựng system call Create**
  6. **Xây dựng system call Open**
  7. **Xây dựng system call Close**
  8. **Xây dựng system call Read**

int Read(char \*buffer, int charcount, OpenFileID id)

If (file rỗng || id file unValid || file stdout):

Return -1

If (Doc file stdin):

Length := chiều dài byte đọc được

Chuyển dữ liệu đọc được cho User

Return length

if (Đọc file thành công):

Chuyển dữ liệu đọc được cho user

Trả về số byte của file

Else:

Return -2

Xóa buffer

* 1. **Xây dựng system call Write**

If (Đọc file rỗng || file stdin ||file read || id file unValid):

Return -1

if(Đọc file stdout): //Ghi lên console

while(file chưa kết thúc):

Đọc từng ký tự

Dùng syncConsole ghi Buffer

Lưu số byte ghi được

if (Đọc file thành công): //Ghi lên file

NextAddress := Vị trí hiện tại của file

Lưu số byte ghi được

Else: (Ghi file rỗng):

Return -2

Xóa Buffer

* 1. **Xây dựng system call Seek**
  2. **Xây dựng chương trình echo**
  3. **Xây dựng chương trình cat**
  4. **Xây dựng chương trình copy**

## Kiểm thử

* 1. **Open, Read, Seek, Write, Close**
  2. **Nhập/xuất từ console**
  3. **Echo**
  4. **Cat**
  5. **Copy**

# TỔNG KẾT

## Đánh giá đồ án

### Mức độ hoàn thành của các thành viên

|  |  |  |
| --- | --- | --- |
| MSSV | Mức độ hoàn thành công việc | Đóng góp |
| 18120078 |  |  |
| 18120211 |  |  |
| 18120227 |  |  |

### Mức độ hoàn thành đồ án:

**Đánh giá**:

## Nguồn tham khảo