**ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN**

**KHOA CÔNG NGHỆ THÔNG TIN**

NGUYỄN MINH KHA

Logo

Description automatically generated

20120622 – Châu Nhật Tuyết

20120635 – Phan Đình Anh Quân

Thành viên trong nhóm:

19120450 – Nguyễn Quốc Anh

20120502 – Nguyễn Minh Kha

20120591 – Lê Đào Thảo Tiên

Giảng viên hướng dẫn: ThS. Lê Viết Long

Môn học: Hệ điều hành

Lớp: 20\_1

**ĐỒ ÁN 2**

**HỆ ĐIỀU HÀNH**

Thành phố Hồ Chí Minh, ngày 9 tháng 11 năm 2022

**MỤC LỤC**

[**A.** **Phân công và ĐÁNH giá** 4](#_Toc119003127)

[**I.** **Phân công** 4](#_Toc119003128)

[**II.** **Đánh giá đồ án** 4](#_Toc119003129)

[**B.** **NGÔN NGỮ Và công cụ hỗ trợ** 5](#_Toc119003130)

[**1.** **Ngôn ngữ lập trình:** 5](#_Toc119003131)

[**2.** **Môi trường** 5](#_Toc119003132)

[**3.** **Công cụ hỗ trợ** 5](#_Toc119003133)

[**C. Cấu hình trước** 5](#_Toc119003134)

[**D.** **Bài làm** 6](#_Toc119003135)

[**1.** **Các exception** 6](#_Toc119003136)

[**2.** **Viết lại cấu trúc điều khiển của chương trình để nhận các Nachos system calls. Kiểm tra cấu trúc mới bằng cách dùng system call Halt để kiểm tra tính đúng đắn của cấu trúc mới**. 7](#_Toc119003137)

[**3.** **Hàm tăng program counter** 8](#_Toc119003138)

[**4.** **System call ReadInt()** 8](#_Toc119003139)

[*a.* *Mô tả* 8](#_Toc119003140)

[*b.* *Kiểm nghiệm kết quả (kết hợp hàm PrintString để ghi các lời dẫn)* 9](#_Toc119003141)

[**5.** **System call PrintInt()** 10](#_Toc119003142)

[*a.* *Mô tả* 10](#_Toc119003143)

[*b.* *Kiểm nghiệm kết quả (kết hợp hàm PrintString để ghi các lời dẫn)* 11](#_Toc119003144)

[**6.** **System call ReadChar()** 11](#_Toc119003145)

[*a.* *Mô tả* 11](#_Toc119003146)

[*b.* *Kiểm nghiệm kết quả (kết hợp hàm PrintString để ghi các lời dẫn)* 12](#_Toc119003147)

[**7.** **System call PrintChar()** 12](#_Toc119003148)

[*a.* *Mô tả* 12](#_Toc119003149)

[*b.* *Kiểm nghiệm kết quả (kết hợp hàm PrintString để ghi các lời dẫn)* 12](#_Toc119003150)

[**8.** **System call ReadString()** 13](#_Toc119003151)

[*a.* *Mô tả* 13](#_Toc119003152)

[*b.* *Kiểm nghiệm kết quả (kết hợp hàm PrintString để ghi các lời dẫn)* 13](#_Toc119003153)

[**9.** **System call PrintString()** 15](#_Toc119003154)

[*a.* *Mô tả* 15](#_Toc119003155)

[*b.* *Kiểm nghiệm kết quả (kết hợp hàm PrintString để ghi các lời dẫn)* 15](#_Toc119003156)

[**10.** **Các chương trình test** 16](#_Toc119003157)

[*a.* *Chương trình help:* 16](#_Toc119003158)

[*b.* *Chương trình ascii:* 17](#_Toc119003159)

[*c.* *Chương trình sort:* 20](#_Toc119003160)

[**E. TÀI LIỆU THAM KHẢO** 22](#_Toc119003161)

1. **Phân công và ĐÁNH giá**
2. **Phân công**

**Bảng phân công:**

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Họ tên - MSSV | Nhiệm vụ chính | Đánh giá |
| 1 | Nguyễn Minh Kha  20120502 | -Câu b, i  -Cấu hình trước  -Viết báo cáo | Hoàn thành 100% |
| 2 | Phan Đình Anh Quân 20120635 | -Câu h  -Viết báo cáo  -Demo đồ án | Hoàn thành 100% |
| 3 | Lê Đào Thảo Tiên  20120591 | -Câu a  -Viết báo cáo | Hoàn thành 100% |
| 4 | Châu Nhật Tuyết  20120622 | -Câu j  -Viết báo cáo | Hoàn thành 100% |
| 5 | Nguyễn Quốc Anh  19120450 | -Câu c, d, e, f, g  -Viết báo cáo | Hoàn thành 100% |

1. **Đánh giá đồ án**

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Công việc | Mô tả | Mức độ hoàn thành |
| 1 | Program counter | Tăng program counter | 100% |
| 2 | Các system call | System call Halt() | 100% |
| 3 | System call ReadInt(), PrintInt() | 100% |
| 4 | System call ReadChar(), PrintChar() | 100% |
| 5 | System call ReadString(), PrintString() | 100% |
| 6 | Chương trình test | Chương trình help | 100% |
| 7 | Chương trình ascii | 100% |
| 8 | Chương trình sort | 100% |
| 9 | Đồ án | | 100% |

1. **NGÔN NGỮ Và công cụ hỗ trợ**
2. **Ngôn ngữ lập trình:**

* C/C++

1. **Môi trường**

* Ubuntu

1. **Công cụ hỗ trợ**

* Phần mềm hỗ trợ: VMWare

# **C. Cấu hình trước**

* Vào thư mục nachos/nachos-3.4/code/Makefile và sửa “gmake” thành “make”.
* Cấp quyền (permission) cho 5 file trong thư mục nachos/nachos-3.4/gnu-decstation-ultrix/decstation-ultrix/2.95.3 và file coff2noff trong thư mục nachos/nachos-3.4/code/bin

Graphical user interface, application

Description automatically generated

* Thêm lớp SynchConsole và khai báo biến toàn cục ptrSynchConsole

+ Thêm 2 file Synchcons.cc và Synchcons.h vào code/threads.

+ Vào thư mục code/threads/system.h khai báo #include "synchcons.h" và khai báo 1 biến toàn cục extern Synchconsole \* ptrSynchConsole, sau khi khai báo thì khởi tạo biến và delete biến.

+ code/Makefile.common khai báo 3 file Synchcons.cc, Synchcons.h, Synchcons.o trong USERPROG\_C, USERPROG\_H, USERPROG\_O.

1. **Bài làm**
2. **Các exception**

Khi xử lý một exception, ta cần quan tâm đến biến which và giá trị của thanh ghi 2.

+ biến which: cho biết exception được throw là exception nào

+ thanh ghi 2: kiểu exception đó

Tất cả các exceptions được liệt kê trong *machine/machine.h*. Hầu hết các exception trong này là run-time errors, khi các exception này xảy ra thì user program không thể được phục hồi. Trường hợp đặc biệt duy nhất là *no exception* sẽ trả quyền điều khiển về HĐH, còn *syscall exceptions* sẽ được xử lý bởi các hàm chúng ta viết cho user system calls. Với tất cả exceptions khác, HĐH hiển thị ra một thông báo lỗi và Halt hệ thống.

Các loại exception:

+ NoException

+SyscallException

+PageFaultException

+ReadOnlyException

+BusErrorException

+AddressErrorException

+OverflowException

+IllegalInstrException

+NumExceptionTypes

1. **Viết lại cấu trúc điều khiển của chương trình để nhận các Nachos system calls. Kiểm tra cấu trúc mới bằng cách dùng system call Halt để kiểm tra tính đúng đắn của cấu trúc mới**.

Text

Description automatically generated

1. **Hàm tăng program counter**

Đoạn mã gốc để tăng program counter nằm trong machine/mipssim.cc cuoi ham "void Machine::OneInstruction(Instruction \*instr), thay đổi lại đoạn mã này cho phù hợp với file exception và tách ra thành một hàm mới lên là Increase\_ProgramCounter.

1. **System call ReadInt()**
2. *Mô tả*

* Dùng con trỏ ptrSynchConsole lớp SynchConsole phương thức Read để đọc từng kí tự.

char c;

ptrSynchConsole->Read(&c, 1);

* Các bước:

+ Bước 0: mảng num\_string chứa các chữ số nhập vào với giá trị khởi tạo num\_string[0] = ‘0’ và số chữ số tối đa cho kiểu số nguyên int là maxlen = 10 (-2^31 + 1 <= int <= 2^31 – 1).

+ Bước 1: Lọc kí tự khoảng trắng và số 0 ở đầu bằng cách đọc đến khi nào kí tự được đọc khác khoảng trắng và ‘0’.

+ Bước 2: Đọc tiếp kí tự tiếp theo, xét dấu kí tự đầu tiên (phải là số hoặc dấu ‘+’ và ‘-‘ nếu không phải thì không thực hiện).

+ Bước 3: Đọc tới khi kí tự được đọc không phải là số và bỏ kí tự được đọc vào mảng num\_string.

+ Bước 4: Tính số nguyên n được nhập vào bằng cách duyệt qua mảng num\_string.

+ Bước 5: Thêm dấu đã xét ở bước 2 vào số nguyên n

+ Bước 6: Ghi số nguyên n vào thanh ghi số 2

* Sau khi thực hiện thì tăng program counter.

1. *Kiểm nghiệm kết quả (kết hợp hàm PrintString để ghi các lời dẫn)*

* Trường hợp nhập đúng

Text

Description automatically generated

* Trường hợp nhập không phải số

Text

Description automatically generated

* Trường hợp nhập số quá lớn:

Text

Description automatically generated

1. **System call PrintInt()**
   1. *Mô tả*

* Bước 1: lấy ra số cần xuất bằng cách đọc giá trị thanh ghi 4, và khai báo một số biến để xử lý việc chuyển dữ liệu kiểu int sang char để xuất ra.
* Bước 2: xử lý nếu giá trị vừa đọc là số âm thì chuyển thành dương và thêm dấu âm trong mảng char của kết quả.
* Bước 3: chuyển từ giá trị kiểu int có n chữ số sang mảng kiểu int có n phần tử để dễ dàng chuyển đổi sáng mảng char.
* Bước 4: lần lượt chuyển từ mảng int sang mảng char có độ dài tương ứng.
* Bước 5: ghi ra màn hình console bằng hàm write trong lớp SynchConsole.
* Sau khi thực hiện thì tăng program counter.
  1. *Kiểm nghiệm kết quả (kết hợp hàm PrintString để ghi các lời dẫn)*

Text

Description automatically generated

1. **System call ReadChar()**
   1. *Mô tả*

* Bước 1: khai báo biến ‘ch’ kiểu char để lưu ký tự.
* Bước 2: đọc ký tự từ màn hình Console bằng hàm read trong lớp SynchConsole.
* Sau đó ghi vào thanh ghi 2 giá trị int(ch).
* Sau khi thực hiện thì tăng program counter.

1. *Kiểm nghiệm kết quả (kết hợp hàm PrintString để ghi các lời dẫn)*

Text

Description automatically generated

1. **System call PrintChar()**
2. *Mô tả*

* Bước 1: đọc giá trị từ thanh ghi 4 bằng hàm readRegister(4), vì hàm này trả về giá trị kiểu int nên phải chuyển giá trị về kiểu char.
* Bước 2: ghi ra màn hình console bằng hàm write trong lớp SynchConsole.
* Sau khi thực hiện thì tăng program counter.

1. *Kiểm nghiệm kết quả (kết hợp hàm PrintString để ghi các lời dẫn)*

Text

Description automatically generated

1. **System call ReadString()**
2. *Mô tả*

* Đọc thanh ghi 4 để lấy địa chỉ xâu được nhập vào (addr)
* Đọc thanh ghi 5 để lấy độ dài tối đa xâu được nhập vào (maxlength)
* Dùng con trỏ ptrSynchConsole lớp SynchConsole để đọc vào buffer

char\* buffer;

ptrSynchConsole->Read(buffer, maxLength);

* Sau đó dùng hàm System2User copy buffer từ System memory space vào User memory space.

System2User(addr, maxLength, buffer);

* Sau khi thực hiện thì tăng program counter.

1. *Kiểm nghiệm kết quả (kết hợp hàm PrintString để ghi các lời dẫn)*

* Trường hợp nhập xâu nhỏ hơn bằng độ dài tối đa.

Text

Description automatically generated

* Trường hợp nhập lớn hơn độ dài tối đa

Text

Description automatically generated

1. **System call PrintString()**
2. *Mô tả*

* Đọc từ thanh ghi số 4 địa chỉ xâu cần in (addr).
* Dùng hàm User2System để copy từ User memory space vào System memory space sau đó gán giá trị cho buffer.

char\* buffer;

buffer = User2System(addr, 255); //255 maxlength

* Dùng con trỏ ptrSynchConsole của lớp SynchConsole với phương thức Write để in kết quả vào console.

ptrSynchConsole->Write(buffer, strlen(buffer));

* Giải phóng buffer (delete buffer)
* Sau khi thực hiện thì tăng program counter.

1. *Kiểm nghiệm kết quả (kết hợp hàm PrintString để ghi các lời dẫn)*

A screenshot of a computer

Description automatically generated

1. **Các chương trình test**
2. *Chương trình help:*
3. Mô tả:

* Dùng PrintString(char[] s) để in ra nội dung.

1. Kiểm nghiệm kết quả:

Graphical user interface, text

Description automatically generated

1. *Chương trình ascii:*
2. Mô tả

* Duyệt i từ 0 đến 255 dùng PrintString để in ra lời dẫn và PrintChar(i) để in ra kí tự tương ứng mã nhị phân

1. Kiểm nghiệm kết quả

Text

Description automatically generated with medium confidence

A screenshot of a computer

Description automatically generated with medium confidence

A screenshot of a computer

Description automatically generated with medium confidence

1. *Chương trình sort:*
2. Mô tả

* Dùng ReadInt() để đọc các số nguyên nhập từ người dùng
* Dùng PrintInt(int n) để in các số nguyên kết quả
* Dùng PrintString để in các lời dẫn

1. Kiểm nghiệm kết quả

Text

Description automatically generated

# **E. TÀI LIỆU THAM KHẢO**

1. Hướng dẫn môn hệ điều hành trên moodle
2. <https://www.youtube.com/watch?v=t0jtY1C129s&list=PLRgTVtca98hUgCN2_2vzsAAXPiTFbvHpO&ab_channel=Th%C3%A0nhChungNguy%E1%BB%85n>