

Phần 1.

Cách thêm một lớp vào Nachos:

Vào FTP của môn học download thêm 2 tập tin: synchcons.h và synchcons.cc hoặc download ở file SynchconsClass.rar trên Moodle. Và sau đó vào chép 2 files này vào thư mục **/code/threads**.

Kế tiếp các bạn mở file: Makefile.common (trong nachos-3.4/code). Và cuối đoạn bắt đầu dòng `USERPROG_H =` thêm vào `../threads/synchcons.h`. Ví dụ như bên dưới:

```
USERPROG_H = ../userprog/addrspace.h\  
../userprog/bitmap.h\  
../filesystem/filesys.h\  
../filesystem/openfile.h\  
../machine/console.h\  
../machine/machine.h\  
../machine/mipsim.h\  
../machine/translate.h\  
../threads/synchcons.h
```

Và cuối đoạn bắt đầu bằng `USERPROG_C = ..` thêm vào `../threads/synchcons.cc`

Và cuối đoạn bắt đầu bằng `USERPROG_O = ..` thêm vào `synchcons.o`

Biên dịch lại Nachos.

Phần 2:

Vào file `machine/machine.h` tìm danh sách các exception và viết lại chương trình bắt tất cả các exception bằng cách sau : Mỗi exception chỉ việc hiện ra thông báo loại exception và gọi hàm `Halt()`.

Sửa file **userprog/exception.cc**

```
switch (which)  
{  
  
    case NoException:  
        return;  
  
    case PageFaultException: ...  
        break;
```

```
case SyscallException:
    switch (type)
    {
        case SC_Halt: ...
        case SC_Open:..
    }
}
```

Với loại exception là SyscallException thì chúng ta vào file **userprog/syscall.h** tìm tất cả các loại syscall để viết hàm xử lý cho tất cả các syscall này.

Nếu phải viết 1 syscall mới từ đầu. Chúng ta sẽ phải vào file **userprog/syscall.h** define 1 syscall mới (vd #define SC_New.. 13). Rồi định nghĩa một prototype để xử lý cho syscall này (ví dụ: int New(int, ...)). Kế tiếp vào tập tin test/start.s viết 1 đoạn assembly để khai báo ánh xạ giữa system call đã định nghĩa và phương thức xử lý nó. Sau đó vào tập tin exception.cc viết hàm xử lý cho prototype int New(int, ...) mà mình đã định nghĩa ở trên.

Chú ý: nếu chúng ta viết hàm xử lý **int New(int, ...)** trong một file là **new.cc** thì lưu file **new.cc** trong thư mục /userprog/. Rồi phải vào file nachos-3.4/code/Makefile.common thêm 1 dòng khai báo file **new.cc** giống như trong ví dụ thêm 1 class mới vào Nachos (lúc chúng ta thêm class synchcons.*)

Ví dụ:

Viết thêm một syscall Sub để trừ 2 số:

Bước 1:

Trong file /code/userprog/syscall.h thêm dòng khai báo syscall mới:

```
#define SC_Sub 43
int Sub (int a, int b);
```

Bước 2:

Trong file /code/test/starts.c và test/start.s thêm dòng:

```
#include syscall.h
.globl Sub
.ent Sub
Sub:
    addiu $2,$0,SC_Sub
    syscall
    j $31
.end Sub
```

Bước 3:

Thêm dòng này trong file **code/userprog/exception.cc**

```
case SC_Sub:
    op1 = machine->ReadRegister (4);
    op2 = machine->ReadRegister (5);
    result = op1 - op2;
    machine->WriteRegister (2, result);
    interrupt->Halt();
```

Bước 4:

Viết chương trình ở mức người dùng để kiểm tra –sub.c

```
#include syscall.h
int main()
{
    int result;
    result = Sub(43, 23);
    Halt();
}
```

Bước 5:

Thêm đoạn vào Makefile trong /code/test/

Thêm sub vào dòng all:

```
all: halt shell matmult sort sub
```

Thêm đoạn sau phía sau matmult:

```
sub.o: sub.c
    $(CC) $(CFLAGS) -c sub.c
sub: sub.o start.o
    $(LD) $(LDFLAGS) start.o sub.o -o sub.coff
    ../bin/coff2noff sub.coff sub
```

Bước 6:

Biên dịch lại Nachos

Phần 3

Viết 2 thủ tục User2System và System2User. Giải thích ?