

```

// Adding an example system call with a pointer.

...
...
...

//We do not give the header files, so you should find them. We assume that
you have add x structure before writing the function codes of the system
call. We assume that you have added the necessary modules. You should add
them in a right way for the proper function of the system call.

asm linkage int sys_xcall(struct x *userspace_pointer){

// x is predefined. x is only an example.

// user_space pointer represents the address of the structure in the user
space.
cli(); // close all interrupts in the kernel.

struct x kernelstructure; // create a kernel space structure of x in the
kernel module.
copy_from_user(&kernelstructure,userspace_pointer,sizeof(struct x));

// this copies the contents of the structure of the user space to structure
in kernel space

kernelstructure.fieldofx = refferedpointerunderkernel->field;

// ...

copy_to_user(userspace_pointer,&kernelstructure,sizeof(struct x));

// this copies the contents of the structure of the kernel space to structure
in user space

sti(); // start all interrupts in the kernel.

return 0;

}

```

```
// The code which calls system call
```

```
...
```

```
...
```

```
...
```

//We do not give the header files, so you should find them. We assume that you have to add x structure before writing the function codes of the system call both in user space and kernel space. You should add them in the right way for the proper function.

```
main(){
```

```
    struct x userdata;
```

```
    ...
```

```
    xcall(&userdata);
```

```
    ...
```

```
    printf("%d\n",userdata.herhangixalan1);
```

```
    return 0;
```

```
}
```