

# 编程作业1

## 作业要求

模仿bash程序做一个多任务控制台，包括以下命令：

1. run: 运行某个程序，可带有参数，也可以指定在哪个CPU上运行。
2. list: 列出当前运行的进程，尽可能的包含一些进程的信息。
3. kill: 杀死某个运行的程序。
4. quit: 退出多任务控制台。

## 运行方法

进入到包含源代码的文件夹下，然后执行 `make` 命令（`-f` 参数用于指定特定文件名的 `makefile`）

```
$ make -f mk1
```

如果源代码之前编译过，并且没有修改，则 `make` 命令会提示可运行程序已是最新。

否则，会按照 `make` 文件对源代码进行编译，得到可运行文件。

可执行文件会生成在当前文件夹下，并且命名为 `myshe11`

接下来，输入以下命令执行程序

```
$ ./myshe11
```

程序运行后，会在终端上输出自定义的命令提示符 `'#'` 来提示输入命令

```
#
```

## 相关命令

### run

通过创建子进程，并调用 `sched_setaffinity()` 和 `execvp()` 函数运行指定程序以及指定运行CPU。

```
命令格式: # run [-c cpu_num] command arg1 arg2 [...]  
例如:  
# run vi test.c  
# run -c 2 vi test.c
```

### list

通过读取 `/proc/` 目录下相关文件来获取运行中进程的相关信息。

```
# list
```

### kill

通过调用 `kill()` 函数杀死指定进程。

```
命令格式: # kill pid  
例如:  
# kill 7619
```

## quit

通过函数 `return` 结束控制台。

```
# quit
```