

● 从本次作业开始，如果未加特殊声明，所有“进程”均指守护进程

【作业要求:】

1. 每个人的目录结构要求如下（假设学号为 1551234，各人按实修改）：首先建立“学号-000106”子目录（可位于任意子目录下），下层不需要再建子目录，示例如下：

1551234-000106

|-- 本次作业的各文件，含 makefile

2. 写一个满足以下各要求的 test 程序（假设学号为 1551234，各人按实修改）

- 2.1. 运行后成为进程，但是不要完全脱离控制台，即在启动控制台上用 ps 仍能查到（如 2.2 图）

```
[root@RHEL71-X64 program]# ./test
[root@RHEL71-X64 program]#
```

- 2.2. 修改进程名，使用 ps 命令（注意：不带任何参数）查看时，在父进程的后面加[main]，随后每隔一秒，分裂出 1 个子进程，共 n 个（ $5 \leq n \leq 100$ ），每个子进程在后面加[sub-xx]

```
[root@RHEL71-X64 program]# ps
  PID TTY          TIME CMD
 3386 pts/0    00:00:01 bash
 4309 pts/0    00:00:00 ./test [main]
 4310 pts/0    00:00:00 ps
[root@RHEL71-X64 program]#
```

10 秒后

```
[root@RHEL71-X64 program]# ps
  PID TTY          TIME CMD
 3386 pts/0    00:00:01 bash
 4309 pts/0    00:00:00 ./test [main]
 4311 pts/0    00:00:00 ./test [sub-01]
 4313 pts/0    00:00:00 ./test [sub-02]
 4314 pts/0    00:00:00 ./test [sub-03]
 4315 pts/0    00:00:00 ./test [sub-04]
 4317 pts/0    00:00:00 ./test [sub-05]
 4318 pts/0    00:00:00 ./test [sub-06]
 4319 pts/0    00:00:00 ./test [sub-07]
 4320 pts/0    00:00:00 ./test [sub-08]
 4321 pts/0    00:00:00 ./test [sub-09]
 4322 pts/0    00:00:00 ./test [sub-10]
 4323 pts/0    00:00:00 ps
[root@RHEL71-X64 program]#
```

- 2.3. 修改进程名，使用 ps -ef 命令（注意：与 ps 不同）查看时，父进程和子进程的名称的后面再加上自己的已运行时间（以秒为单位），即父进程分裂 n 个子进程完成后，只要子进程不退出，只需要每秒更新一次自己的运行时间即可，不需要做其它操作；子进程运行后，只需要每秒更新一次自己的运行时间，不需要做其它操作

```
[root@RHEL71-X64 program]# ps -ef | grep test
root      4309      1  0 02:24 pts/0    00:00:00 ./test [main 00:01:05]
root      4311    4309  0 02:24 pts/0    00:00:00 ./test [Sub-01 00:01:04]
root      4313    4309  0 02:24 pts/0    00:00:00 ./test [Sub-02 00:01:03]
root      4314    4309  0 02:24 pts/0    00:00:00 ./test [Sub-03 00:01:02]
root      4315    4309  0 02:24 pts/0    00:00:00 ./test [Sub-04 00:01:01]
root      4317    4309  0 02:24 pts/0    00:00:00 ./test [Sub-05 00:01:00]
root      4318    4309  0 02:24 pts/0    00:00:00 ./test [Sub-06 00:00:59]
root      4319    4309  0 02:24 pts/0    00:00:00 ./test [Sub-07 00:00:58]
root      4320    4309  0 02:24 pts/0    00:00:00 ./test [Sub-08 00:00:57]
root      4321    4309  0 02:24 pts/0    00:00:00 ./test [Sub-09 00:00:56]
root      4322    4309  0 02:24 pts/0    00:00:00 ./test [Sub-10 00:00:55]
root      4326    3386  0 02:25 pts/0    00:00:00 grep --color=auto test
[root@RHEL71-X64 program]#
```

- 2.4. 用 kill 杀掉若干子进程后，父进程会再次分裂，补齐 n 个，且子进程的编号占用之前被 kill 子进程的编号

```
[root@RHEL71-X64 program]# kill -9 4315 4313 4321
[root@RHEL71-X64 program]#
```

3 秒后

```
[root@RHEL71-X64 program]# ps -ef | grep test
root      4309      1  0 02:24 pts/0    00:00:00 ./test [main 00:02:06]
root      4311    4309  0 02:24 pts/0    00:00:00 ./test [Sub-01 00:02:06]
root      4314    4309  0 02:24 pts/0    00:00:00 ./test [Sub-03 00:02:04]
root      4317    4309  0 02:24 pts/0    00:00:00 ./test [Sub-05 00:02:02]
root      4318    4309  0 02:24 pts/0    00:00:00 ./test [Sub-06 00:02:01]
root      4319    4309  0 02:24 pts/0    00:00:00 ./test [Sub-07 00:02:00]
root      4320    4309  0 02:24 pts/0    00:00:00 ./test [Sub-08 00:01:59]
root      4322    4309  0 02:24 pts/0    00:00:00 ./test [Sub-10 00:01:57]
root      4327    4309  0 02:26 pts/0    00:00:00 ./test [Sub-02 00:00:07]
root      4328    4309  0 02:26 pts/0    00:00:00 ./test [Sub-04 00:00:06]
root      4329    4309  0 02:26 pts/0    00:00:00 ./test [Sub-09 00:00:05]
root      4332    3386  0 02:26 pts/0    00:00:00 grep --color=auto test
[root@RHEL71-X64 program]#
```

- 2.5. Kill 父进程后，所有子进程自动结束（注意：不允许杀子进程）

```
[root@RHEL71-X64 program]# kill -9 4309
[root@RHEL71-X64 program]#
[root@RHEL71-X64 program]# ps -ef | grep test
root      4335    3386  0 02:28 pts/0    00:00:00 grep --color=auto test
```

- 2.6. n 值从 /etc 目录下的 1551234.conf 文件中读取，文件中仅一行，存放运行时需要分裂子进程的数量，合理范围 5-100，如果超过合理范围，缺省为 5

子进程数量=10

- 2.7. 读配置文件的函数编译为动态链接库，命名为 lib1551234.so，放入/usr/lib64 中  
（注意：test 程序打开的是 /usr/lib64/lib1551234.so，不是本目录下的）

- 2.8. 在/usr 下新建 1551234 子目录，里面放一个 1551234.dat 文件，内容为学号+姓名，与可执行程序无关，可执行程序不需要读写此文件

## 2.9. makefile 文件要求如下:

**make** : 生成可执行文件、动态链接库文件

- 本目录下生成 test
- 本目录下生成 lib1551234.so

**make install**: 将可执行文件及附属文件放入指定位置

- 本目录下已生成的 test, 复制一份到 /usr/sbin 下, 改名为 test-1551234
- 本目录下已生成的 lib1551234.so, 复制一份到 /usr/lib64 下
- 将本目录下预先准备好的 1551234.dat 文件复制到 /usr/1551234 下 (如果子目录未建立, 则需要先建立子目录, 不必判断 1551234.dat 在 /usr/1551234 下是否存在, 每次替换即可)
- 将本目录下预先准备好的 1551234.conf 文件复制到 /etc 下 (不必判断 1551234.conf 在 /etc 下是否存在, 每次替换即可)

**make rpm** : 生成 rpm 安装包

- 将本目录下预先准备好的, 与生成 rpm 有关的文件 (包括 service), 复制到特定目录下 (不必判断特定目录下这些文件是否存在, 每次替换即可)
- 生成 rpm 包, 复制到本目录下

**make uninstall**: 清除指定位置的可执行文件及附属文件

- 清除 /usr/sbin/test-1551234
- 清除 /usr/lib64/lib1551234.so
- 清除 /etc/1551234.conf 文件
- 清除 /usr/1551234/1551234.dat 并删除 /usr/1551234 目录 (如果里面有手工放入的其它文件, 则不能删除)
- 清除本目录外的, 与生成 rpm 有关的文件 (包括 service)
- 清除本目录外的 rpm 包
- rpm -e 与 make uninstall 应该一致

**make clean**: 清除本目录下的文件

- 清除本目录下的可执行文件 test 及动态链接库文件 lib1551234.so
- 本目录下的 1551234.dat 及 1551234.conf 不需要删除
- 本目录下的与生成 rpm 有关的文件 (包括 service) 不需要删除
- 清除本目录下的 rpm 包

## 3. 写一个 test-1551234.service, 放在特定目录下, 能完成如下要求:

systemctl enable test-1551234.service : 使开机时自动运行 test-1551234 进程  
systemctl disable test-1551234.service : 取消开机时自动运行 test-1551234 进程  
systemctl start test-1551234.service : 运行 test-1551234 进程  
systemctl stop test-1551234.service : 停止运行 test-1551234 进程  
systemctl restart test-1551234.service : 先停止, 再次运行 test-1551234 进程

## 4. 将 test 程序及附件共同打包成一个 rpm 安装包 (以学号 1551234 为例, 各人按实修改):

### 4.1. 安装包名: test-1551234.rpm

### 4.2. 用 rpm -ivh 1551234-test.rpm 安装后, 目录分布如下:

/etc/1551234.conf : 配置文件  
/usr/sbin/test-1551234 : 可执行文件 (1551234-000106 下名为 test, 此处换名)  
/usr/lib64/lib1551234.so : 动态链接库文件  
/usr/1551234/1551234.dat : 1551234.dat 中存放你的学号姓名, 安装后放在 /usr/1551234 目录下 (1551234 目录如果不存在, 新建一个)  
test-1551234.service : 特定目录下

- 4.3. 用 `rpm -e` 卸载后，要依次删除上面的 5 个文件及 1 个目录，但如果 `/usr/1551234` 目录下还有其他文件，则仅删除 `1551234.dat` 文件而保留目录
- 4.4. rpm 包要有相应的提示  
安装开始前，给出提示“准备安装 test-1551234”  
安装完成后，给出提示“完成安装 test-1551234”  
卸载开始前，给出提示“准备卸载 test-1551234”  
卸载完成后，给出提示“完成卸载 test-1551234”
5. 将上述制作 service 及 rpm 包的方法写成文档并单独提交

### 【本次作业目录结构要求及批改方法:】

- 1、首先建立 15-000106 目录（可位于任意目录下）
- 2、每个人的目录结构要求如下（假设学号为 1551234，各人按实修改）：首先建立“学号-000106”子目录（可位于任意子目录下），下层不需要再建子目录，示例如下：  
1551234-000106  
|-- 本次作业的各文件，含 makefile
- 3、提交作业时，每位同学上交一个 linux-rpm.tar.bz2 文件，解压后能得到上述的目录结构，截止时间到后，会从每人的交作业目录中复制出来，全部放在 15-000106 目录中  
示例如下：  
15-000106  
|-- 1551234-linux-rpm.tar.bz2      （第 1 位同学的作业压缩包）  
...  
`-- 1554321-linux-rpm.tar.bz2      （最后 1 位同学的作业压缩包）  
  
依次解压后，能得到如下目录结构：  
15-000106  
|-- 1551234-000106                      （第 1 位同学的作业目录）  
...  
`-- 1554321-000106                      （最后 1 位同学的作业目录）
- 4、进入 15-000106 目录，进行一次 make，就能生成所有可执行文件，示例如下：  
15-000106  
|-- 1551234-000106                      （第 1 位同学的作业目录）  
...  
|-- 1554321-000106                      （最后 1 位同学的作业目录）  
`-- makefile                      （老师事先建好的 makefile 文件，准备编译所有同学的本次作业，具体的实现方式是进入到每个学号对应的目录后调用该目录下的总 makefile）
- 5、无法顺利编译则不能得分，对应学号及子目录名错则不能得分
- 6、作业提交时清除所有的中间文件及生成的可执行文件、源程序备份文件等

### 【作业要求:】

- 1、将上述问题的解答及对应的简单测试程序写成文档，10 月 22 日前网上提交
- 2、每题所占平时成绩的具体分值见网页
- 3、超过截止时间提交作业则不得分