

操作系统实验二验收标准

第一阶段：60 分

1.运行 Fork 测试实例（10 分）

执行我们给出的 fork.c 用户测试程序，能出现如下的结果，得分 10 分。

```
$ ../build.linux/nachos -x fork

1. init var = "parent"
2. i am child. i change var = "child"
3. i am parent. my var = "parent"

Machine halting!

Ticks: total 335158, idle 0, system 33550, user 301608
Disk I/O: reads 0, writes 0
Console I/O: reads 0, writes 0
Paging: faults 0
Network I/O: packets received 0, sent 0
```

其他情况，如输出结果不全，出现段错误等，若能正确解释 Fork 的执行流程，得分 3 分，否则 0 分。

2.运行 Exec 测试实例（10 分）

执行我们给出的 exec.c 用户测试程序，能出现如下的结果，得分 10 分。

```
$ ../build.linux/nachos -x exec

1. i am child. i am runing 'add'.
   42 + 23 = 65
2. i am parent. i finished after my child

Machine halting!

Ticks: total 1823624, idle 0, system 182400, user 1641224
Disk I/O: reads 0, writes 0
Console I/O: reads 0, writes 0
Paging: faults 0
Network I/O: packets received 0, sent 0
```

其他情况，如输出结果不全，出现段错误等，若能正确解释 Exec 的执行流程，得分 3 分，否则 0 分。

3.运行 Join 测试实例（10 分）

执行我们给出的 join.c 用户测试程序，能出现如下的结果，得分 10 分。

```
$ ../build.linux/nachos -x join

1. i am parent. i am waitting my childID=3
2. i am child. i am runing 'add'. please wait...
   42 + 23 = 65
3. parent finished

Machine halting!

Ticks: total 2447651, idle 0, system 244810, user 2202841
Disk I/O: reads 0, writes 0
Console I/O: reads 0, writes 0
Paging: faults 0
Network I/O: packets received 0, sent 0
```

其他情况，如输出结果不全，出现段错误等，若能正确解释 Join 的执行流程，得分 3 分，否则 0 分。

4.回答问题（30 分）

我们会提问三个相关问题，如实现某个系统调用的具体过程，相关类里某个函数的作用等问题。每个问题回答正确各得 10 分，由于存在主观因素，故没有具体的评判标准。

第二阶段：40 分

1.运行 Shell 测试实例（10 分）

执行补全完整的 shell.c 用户测试程序，能够在 shell 中运行单命令（5 分），多命令（5 分）。

```
nachos >> ./add
42 + 23 = 65

nachos >> ps
Unable to open file ps
exec cmd by Linux
  PID TTY          TIME CMD
 18087 pts/3        00:00:00 bash
 24660 pts/3        00:00:00 nachos
 24663 pts/3        00:00:00 sh
 24664 pts/3        00:00:00 ps

nachos >> ./2;./add;./2;./2;./2
42 + 23 = 65
i am testshell
i am testshell
i am testshell
i am testshell

nachos >> 
```

2.验证进程动态优先级调度算法（10 分）

在 shell 中执行多命令（./2;./add;./2;./2;./2;./2）后，能在 test 目录的 0schedule_info.txt 文件中观察到正确的动态优先级调度结果：正确的进程状态信息和优先级变化结果（10 分）。

```
50
51 -----one switch-----
52 running: tid=4  name=2                status=RUNNING  pri=222
53
54 Ready list contents:
55     tid=3  name=2                status=READY   pri=225
56     tid=1  name=main             status=READY   pri=148
57
58 -----one switch-----
59 running: tid=3  name=2                status=RUNNING  pri=224
60
61 Ready list contents:
62     tid=4  name=2                status=READY   pri=224
63     tid=1  name=main             status=READY   pri=150
64
65 -----one switch-----
66 running: tid=4  name=2                status=RUNNING  pri=223
67
68 Ready list contents:
69     tid=3  name=2                status=READY   pri=226
70     tid=1  name=main             status=READY   pri=152
71
72
```

1

2

3

3.回答问题（20 分）

我们会提问一些相关问题，包括代码实现的具体流程，以及相关代码的作用等问题。由于存在主观因素，故没有具体的评判标准。