实验6——CPU调度算法

1.实验内容

- 任务一

了解CPU的几种调度算法，了解其工作原理及其优先级设定等一些相关参数的设置。

使用pthread_attr_getschedpolicy方法获取指定线程的调度算法，利用sched_get_priority_max和sched_get_priority_min可以获取到指定调度算法可以设置的最大/最小优先级，pthread_attr_setschedpolicy方法可以设置线程的调度算法。

创建一个线程，获取该线程当前的调度算法以及该调度算法可以设置的最大/最小优先级，然后再获取到除该调度算法外的另外两种调度算法的可以设置的最大/最小优先级。获取当前线程的优先级，并尝试手动设置该线程的调度策略，最后恢复该线程的调度策略。

2.实验示例代码

- 需要用到的头文件

```
#include <unistd.h>
#include <pthread.h>
#include <sched.h>
#include <stdio.h>
```

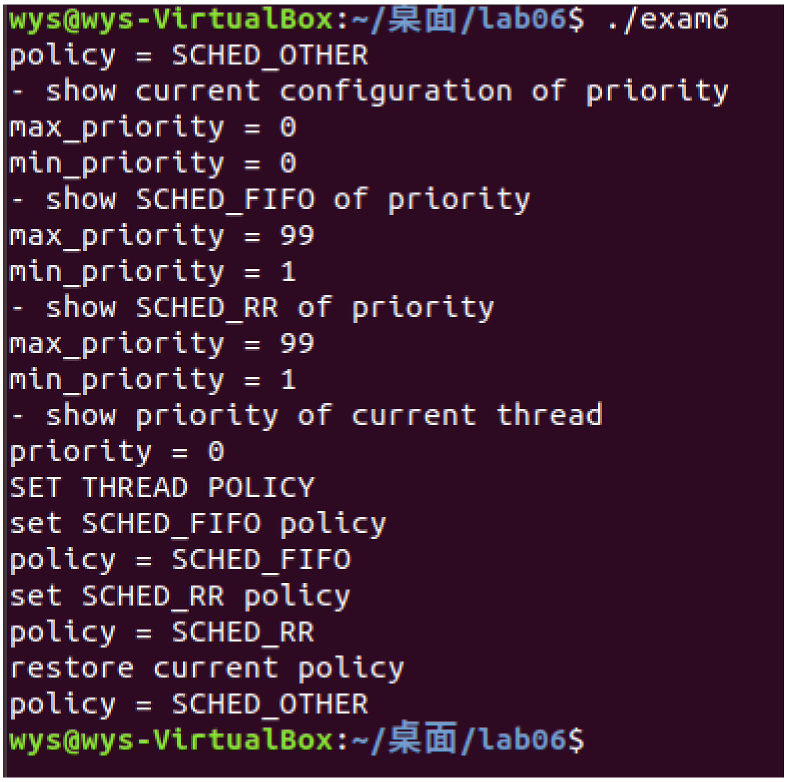
- 任务中可能用到的一些方法

```
//获取线程的调度算法
static int get_thread_policy(pthread_attr_t attr)
{
    int policy;
    pthread_attr_getschedpolicy(&attr, &policy);
    switch(policy)
    {
        //实时调度策略 先到先服务
        case SCHED_FIFO:
            printf("policy = SCHED_FIFO\n");
            break;
        //时间片轮转调度算法
        case SCHED_RR:
            printf("policy = SCHED_RR\n");
            break;
        //分时调度算法
        case SCHED_OTHER:
            printf("policy = SCHED_OTHER\n");
            break;
        default:
            printf("policy = UNKOWN\n");
            break;
    }
    return policy;
}
```

```c

//获取指定调度算法可以设置的最大/最小优先级
static void show_thread_priority(pthread_attr_t attr,int policy)
{
    int priority = sched_get_priority_max(policy);
    printf("max_priority = %d\n",priority);
    priority = sched_get_priority_min(policy);
    printf("min_priority = %d\n",priority);
}

//获取给定线程的优先级
static int get_thread_priority(pthread_attr_t attr)
{
    struct sched_param param;
    pthread_attr_getschedparam(&attr, &param);
    printf("priority = %d\n", param.sched_priority);
    return param.sched_priority;
}

//设置线程的调度算法
static void set_thread_policy(pthread_attr_t attr,int policy)
{
    pthread_attr_setschedpolicy(&attr,policy);
    get_thread_policy(attr);
}
```

3.实验结果示例

注意，本实验中用到了pthread库，需要在编译时指明链接库，例如“gcc -o xxx xxx.c -lpthread”



```
wys@wys-VirtualBox:~/桌面/lab06$ ./exam6
policy = SCHED_OTHER
- show current configuration of priority
max_priority = 0
min_priority = 0
- show SCHED_FIFO of priority
max_priority = 99
min_priority = 1
- show SCHED_RR of priority
max_priority = 99
min_priority = 1
- show priority of current thread
priority = 0
SET THREAD POLICY
set SCHED_FIFO policy
policy = SCHED_FIFO
set SCHED_RR policy
policy = SCHED_RR
restore current policy
policy = SCHED_OTHER
wys@wys-VirtualBox:~/桌面/lab06$
```

示例