

Task 1

张配天-2018202180

4.4

对于一个需要调用 I/O 设备的进程, 在单核处理器的计算机下 (不能同时运行两个线程), 考虑其在两种模式下工作的情况:

- i. **单线程情况:** 进程在发送 I/O 请求后会一直在就绪态/就绪挂起态等待 I/O 设备的响应; 此时进程的其他部分都无法得到处理器资源, 不会运行; 直到得到 I/O 设备响应才会继续运行。
- ii. **多线程情况:** 不妨假设进程分为两个线程, 调用 I/O 设备的和其余部分, 记为 T1 和 T2; 这样, 在 T1 请求 I/O 设备进入等待时, T2 不会受到影响, 仍然可以得到处理器资源, 可以继续运行。

因此, 多线程情况的效率显然高于单线程。

4.5

利用公式:

$$time_ratio = \frac{1}{(1 - percent) + \frac{percent}{processor}}$$

计算得到新速度约为原执行速度的 2.5 倍

4.7

- a. 程序完成对一个 list 链表结构体中 val 值为正数的个数的计数
- b. 在此种情况下, 程序可以正确执行。因为在这种情况下, 线程 b 的内容不会被激活执行, 即线程 b 不会对该全局变量进行修改。而传入的为所有值均为负的链表, global_positive 的初值为 0, 因此结果正确。

4.9

a. 程序在主函数中输出 ‘o’，在定义的线程中输出 ‘.’，同时两者都使 myglobal 值加 1

b. 显然不是想得到的结果。

记主函数的线程为 T1, 子线程为 T2, 由于 T1 和 T2 隶属于同一个进程 (thred2.c), 两者资源共享, 即共享 myglobal。又因为子线程中不是直接执行 myglobal++ 来更新 myglobal 的值, 而是分成三步:

$$j = myglobal \quad (1)$$

$$j = j + 1 \quad (2)$$

$$myglobal = j \quad (3)$$

考虑这种情况:myglobal 初值为 0。

处于激活态的线程	操作系统动作	myglobal 值	j 值 (T2 中)
.....			
T2	执行到 (3) 前, 由于调用 I/O 设备而导致 T2 阻塞, 激活 T1	0	1
T1	myglobal = myglobal+1	1	1
T1	由于调用 I/O 设备而导致 T1 阻塞, 激活 T2	1	1
T2	执行 (3)	1	1
.....			

因此, 在这样一次循环中, 程序并没有完成其目的 (让 myglobal=myglobal+2)

即使加入 sleep(), 也无法保证 T1,T2 运行的互不干扰。因此, 按照这种方式运行的程序, 其 myglobal 最终结果一定大于等于 20 且小于等于 40, 而具体是多少, 以及字符出现的顺序, 都是不可期的。

疑问

- 4.7 中若传入的是含有正数的链表, 结果会如何?
- 将 4.9 中代码复制到 ubuntu 中运行, 虽然字符出现顺序不同, 但 myglobal 最终值永远是 21, 这是为什么?(debug 失败)