High Performance Computing

2023 Fall

Lab 4. Multiprocessing

Yongzhuo Ma

September 25, 2023

Chapter 1

Introduction to the Environment

1.1 Host Machine

Item	Value
OS version	macOS Ventura 13.5.2
Apple clang version	14.0.3
CPU	Apple M2 Max
CPU Frequency	3.54 - 3.70 GHz
CPU Cores	12
Memory	64 GB

1.2 Virtual Machine

Item	Value
Virtualization	Parallels
OS version	Ubuntu 22.04.3 LTS
gcc version	11.4.0
CPU	Apple M2 Max
CPU Frequency	3.66 GHz
CPU Cores	2
Memory	2 GB

Chapter 2

Differences of time control : Spin() vs sleep()

Spin() and sleep()

Spin() is a function defined in **common.h**, while **sleep()** is a function declared in **unistd.h**.

Spin(1) and sleep(1) both mean pausing the program for 1 second, but there is a slight different when doing multiprocessing. The code is modified as followings:
In parent and child processes,

(1) Get the start time.

(2) Spin(1) or sleep(1)

(3) Get the end time.

(4) Print relevant data.

p1.c [using sleep(1)]

```
1 hello world (pid:7833)
2 hello, I am parent of 7834 (pid:7833)
3 hello, I am child (pid:7834)
4 Sleep 1 – Child : 1695642590.178802 , 1695642591.179124
5 Sleep 1 – Parent: 1695642590.178602 , 1695642591.181993
```

Time of Child = 1.000322 s, Time of Parent = 1.003391 s

p2.c [using Spin(1)]

```
1 hello world (pid:7876)
2 hello, I am parent of 7877 (pid:7876)
3 hello, I am child (pid:7877)
4 Sleep 1 – Parent: 1695642600.605678 , 1695642601.605678
5 Sleep 1 – Child : 1695642600.605819 , 1695642601.605819
```

Time of Child = 1.000000 s, Time of Parent = 1.000000 s

The results show that the function **sleep()** will be affected if the system resources has been used. By the way, this function is not precise for most situation.

Things changed when using **Spin()**, the durations of the two processes has been controlled to 1 second precisely. It is possible to explain this seemingly weird phenomenon by looking into the details of the two functions.

As shown in **common.h**, **Spin()** is based on comparison. The while statement in that function will end when the duration is exactly 1 second. However, **sleep()** pauses the thread of this program. Let us say, sleep(1) means the thread will be load into a queue and wait for 1 second there. But the system may not be available to dequeue this thread just at the precise 1 second. So it is common to encounter a slight delay.

DESCRIPTION OF sleep()
The sleep() function suspends execution of the calling thread until either ___ seconds have elapsed or a signal is delivered to the thread and its action is to invoke a signal-catching function or to terminate the thread or process. System activity may lengthen the sleep by an indeterminate amount.