

COP 5614 U01 - Operating Systems - Fall 2017

Warm-up project report

Student: Yezehao Huai

PID: 5965823

E-mail: yhuai001@fiu.edu

1. Introduction

I am a student major in Finance and studying CS for Financial Engineering, and this is the first time I touch something like that. It's amazing! I learnt a lot from this project. First is how to use Oracle VM VirtualBox create another OS in my laptop, and study how to use Linux Ubuntu. Then I learnt something of Pthread and I tried my best to understand it and finish the project.

All the work I do are referring the warm-up project orientation and links in it. All the files include screenshots, program, and source code are in the 5965823.zip.

2. Description and Output

I used Gedit to do the coding. Then I compiled it by using “**gcc –pedantic –Wall –o OSwarmup test2.c –lpthread**”. The program name should be **OSwarmup** and the source code is **test2.c**. They all in the home folder. Finally I run it in Terminal by using “**./OSwarmup**”. Sometimes the varivable value is as same as the threads number, sometimes the different threads get the same value.

Here are the screenshots:

```
rocky@rocky-VirtualBox:~$ ./OSwarnup
Creating thread 0
Creating thread 1
Creating thread 2
Creating thread 3
Creating thread 4
thread #0 sees value 0
thread #1 sees value 1
thread #4 sees value 2
thread #2 sees value 3
thread #3 sees value 4
rocky@rocky-VirtualBox:~$ ./OSwarnup
Creating thread 0
Creating thread 1
Creating thread 2
Creating thread 3
Creating thread 4
thread #0 sees value 0
thread #1 sees value 1
thread #3 sees value 2
thread #4 sees value 3
thread #2 sees value 4
rocky@rocky-VirtualBox:~$ ./OSwarnup
Creating thread 0
Creating thread 1
Creating thread 2
Creating thread 3
Creating thread 4
thread #0 sees value 0
thread #1 sees value 0
thread #4 sees value 1
thread #2 sees value 2
thread #3 sees value 3
thread #0 sees value 4
rocky@rocky-VirtualBox:~$ gcc -pedantic -Wall -o OSwarnup test2.c -lpthread
rocky@rocky-VirtualBox:~$ ./OSwarnup
Creating thread 0
Creating thread 1
Creating thread 2
Creating thread 3
Creating thread 4
thread #1 sees value 0
thread #0 sees value 1
thread #3 sees value 2
thread #4 sees value 3
thread #2 sees value 4
```

```
*test2.c (-/-) - gedit
#include <pthread.h>
#include <stdio.h>
#include <stdlib.h>
#define THREADS 5

int SharedVariable = 0;

void *SimpleThread(void *threadid)
{
    long tid;
    tid = (long)threadid;
    int val;

    val = SharedVariable;
    printf("thread #id sees value %d\n", tid, val);
    SharedVariable = val + 1;
    pthread_exit(NULL);
}

/* The part was given in guidance */

int main(int argc, char *argv[])
{
    pthread_t threads[THREADS];
    long i;
    for (i = 0; i < THREADS; i++)
    {
        printf("creating thread %ld\n", i); /* Show which thread is being created */
        pthread_create(&threads[i], NULL, SimpleThread, (void *)i);
        /* Create threads and let them run the SimpleThread */
    }

    pthread_exit(NULL); /* exit main() */
}
```

```
Test Editor
rocky@rocky-VirtualBox:~$ ./OSwarnup
Creating thread 2
Creating thread 3
thread #0 sees value 0
thread #1 sees value 1
thread #4 sees value 2
thread #2 sees value 3
thread #3 sees value 4
rocky@rocky-VirtualBox:~$ ./OSwarnup
Creating thread 0
Creating thread 1
Creating thread 2
Creating thread 3
Creating thread 4
thread #1 sees value 0
thread #0 sees value 1
thread #3 sees value 2
thread #4 sees value 3
rocky@rocky-VirtualBox:~$ ./OSwarnup
Creating thread 0
Creating thread 1
Creating thread 2
Creating thread 3
Creating thread 4
thread #0 sees value 0
thread #1 sees value 0
thread #2 sees value 1
thread #3 sees value 2
thread #4 sees value 3
rocky@rocky-VirtualBox:~$ ./OSwarnup
Creating thread 0
Creating thread 1
Creating thread 2
Creating thread 3
Creating thread 4
thread #2 sees value 1
thread #0 sees value 2
thread #1 sees value 3
thread #4 sees value 4
rocky@rocky-VirtualBox:~$ gcc -pedantic -Wall -o OSwarnup test2.c -lpthread
rocky@rocky-VirtualBox:~$ ./OSwarnup
Creating thread 0
Creating thread 1
Creating thread 2
Creating thread 3
Creating thread 4
thread #0 sees value 0
thread #1 sees value 1
thread #3 sees value 2
thread #4 sees value 3
thread #2 sees value 4
rocky@rocky-VirtualBox:~$
```