

2019 Fall Introduction to Operating Systems

Homework 3

0616015 劉姿利

1 Briefly describe the design for the sort and merge function and the thread management in the multi-thread program.

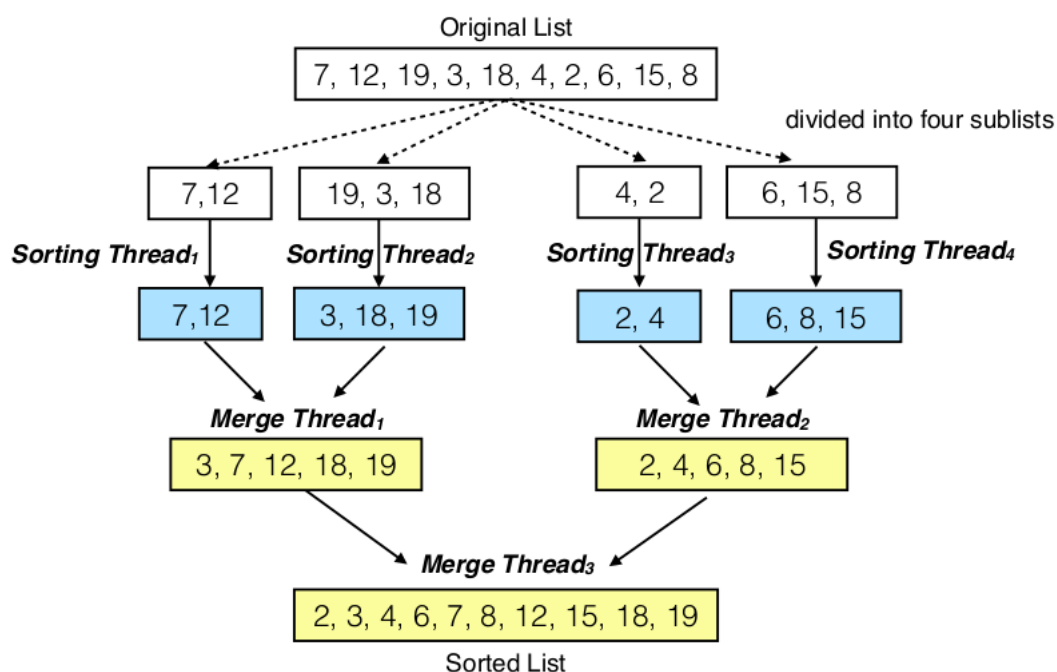
1.1 Sort

Brute-force bubble sort.

1.2 Merge

Two-pointer sorting.

1.3 Thread Management



- Let each thread be either `sorting_thread` or `merge_thread`.
- For each `sorting_thread`, just simple creat it and execute.
- For each `merge_thread`, we have first wait for other 2 threads before execution.

- merge_thread_1 waits for sorting_thread_1 and sorting_thread_2.
- merge_thread_2 waits for sorting_thread_3 and sorting_thread_4.
- merge_thread_3 waits for merge_thread_1 and merge_thread_2.

1.3.1 Waiting Design for Merge_threads

I pass the thread indexes to wait for by passing a structure S.

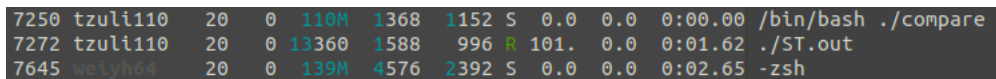
```
void *_merge(void* data) {
    S* ptr = (S*)data;
    int id1 = ptr->id1;
    int id2 = ptr->id2;

    pthread_join(id[id1], NULL);
    pthread_join(id[id2], NULL);

    ...
}
```

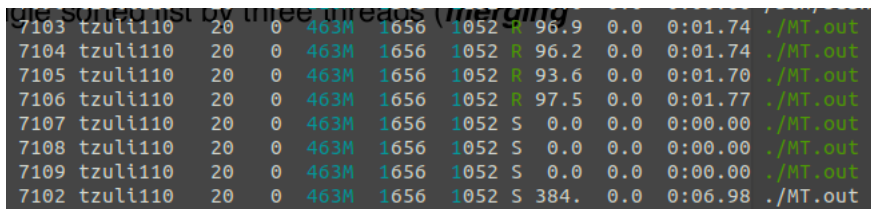
2 Show the thread information screenshot while running the single-thread/multi-thread program.

2.1 Single-thread



```
7250 tzuli110 20 0 110M 1368 1152 S 0.0 0.0 0:00.00 /bin/bash ./compare
7272 tzuli110 20 0 1336M 1588 996 R 101.0 0.0 0:01.62 ./ST.out
7645 welyh04 20 0 139M 4576 2392 S 0.0 0.0 0:02.65 -zsh
```

2.2 Multi-thread



```
7103 tzuli110 20 0 463M 1656 1052 R 96.9 0.0 0:01.74 ./MT.out
7104 tzuli110 20 0 463M 1656 1052 R 96.2 0.0 0:01.74 ./MT.out
7105 tzuli110 20 0 463M 1656 1052 R 93.6 0.0 0:01.70 ./MT.out
7106 tzuli110 20 0 463M 1656 1052 R 97.5 0.0 0:01.77 ./MT.out
7107 tzuli110 20 0 463M 1656 1052 S 0.0 0.0 0:00.00 ./MT.out
7108 tzuli110 20 0 463M 1656 1052 S 0.0 0.0 0:00.00 ./MT.out
7109 tzuli110 20 0 463M 1656 1052 S 0.0 0.0 0:00.00 ./MT.out
7102 tzuli110 20 0 463M 1656 1052 S 384.0 0.0 0:06.98 ./MT.out
```

3 Show the time speedup between single-thread and multi-thread.

3.1 Single-thread

```
tzuli110@linux1:~/Work/2019-OS/HW3$ g++ -Wall 0616015_ST.cpp -o ST.out
tzuli110@linux1:~/Work/2019-OS/HW3$ time ./ST.out < input2.txt > output2_ST.txt
real    0m40.426s
user    0m40.376s
sys     0m0.016s
```

3.2 Multi-thread

```
tzuli110@linux1:~/Work/2019-OS/HW3$ g++ -Wall 0616015_MT.cpp -lpthread -o MT.out
0616015_MT.cpp: In function 'void* _sort(void*)':
0616015_MT.cpp:84:1: warning: no return statement in function returning non-void [-Wreturn-type]
    }
    ^
0616015_MT.cpp: In function 'void* _merge(void*)':
0616015_MT.cpp:114:1: warning: no return statement in function returning non-void [-Wreturn-type]
    }
    ^
tzuli110@linux1:~/Work/2019-OS/HW3$ time ./MT.out < input2.txt > output2_MT.txt
real    0m2.424s
user    0m9.293s
sys     0m0.006s
```

4 What I learned from doing homework 3.

- How to use pthread api.
- How to check the number of threads by using htop.
- How to use stucture and type conversion to pass arguments into threads.