

**Lab report**

|  |  |
| --- | --- |
| **Course**: | Operating System Principle |
| **Semester**: | 2nd semester of the academic year **2020-2021** |
| **Major**: | Software Engineering |
| **Class**: | 2019 |
| **Student Name**: | 冯春霖 |
| **Student ID:** | 222019321062074 |
| **Teacher:** | ZHAO, Hengjun (赵恒军) |

**School of Computer and Information Science**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Name | | Pthread Library and Concurrent Programming | | | |
| Date | | April，2021 | Type | | √ Confirmatory  √ Design  √ Comprehensive |
| 1. **Objective & Requirements**    1. Grasp the Pthreads API for thread creation, termination operations    2. Grasp concurrent programming skills | | | | | |
| 1. **Experimental environment (**platform and software**)**   Virtualbox + Ubuntu (or other platform+linux system combinations) | | | | | |
| 1. **Experimental content and design** (Main Content, Procedure, Codes and Results) 2. Task1    * 1. Create a new thread in the main thread      2. Pass to integers to the new thread and calculate the sum of the two integers by the new thread 3. Task2    * 1. Define an integer array of length 200000      2. Randomly initialize the integer array      3. Sort the initialized integer array and measure the time cost (hint: use the time command) 4. Task3    * 1. Write a C program to merge two sorted integer arrays in to a single sorted integer array 5. Task4   Write a multithreaded sorting program that works as follows:   * + 1. Set the number of CPUs of your virtual machine to at least 2 in VirtualBox     2. Define two GLOBAL integer arrays **a** and **b**, both of length 200000     3. Randomly initialize the array **a**     4. In the main thread, create two new threads to sort the first half and the second half of array **a** respectively     5. The main thread waits for the two new threads to terminate, and then merge the sorted first and second half of array **a** into array **b**     6. Compare the time cost of your multithreaded program with the time you obtained in Task 2, and compute the speedup. (hint: **time** command)      1. Please provide your procedure and source codes to perform the tasks. | | | | | |
| 1. **Result analysis and discussion**（Analysis of experimental results and summing up the harvest and the existing problems） | | | | | |
| Comments & Evaluation | Content & Design (A-E) | | |  | |
| Procedure & Codes (A-E) | | |  | |
| Results (A-E) | | |  | |
| Analysis & Discussion (A-E) | | |  | |
| Score (A-E):  Feedback comments: | | | | |