

CS2106 Lab 4 bonus grading scheme and grader

The grader for the lab 4 bonus (exercise 6) simply does the following:

1. First, allocate (userswap_alloc) 2048 pages.
2. Next, map (userswap_map) 2048 pages of a newly created empty file.
3. Spawn 8 threads, each of which do the following simultaneously:
 1. Alternately read each page of the allocation and map in sequence.
 2. Randomly read pages from the allocation and map. Each thread follows a different sequence.
 3. Write to each page of the allocation and map in sequence. The value written depends solely on the address and the thread.
 4. Alternately read each page of the allocation and map in sequence again.
 5. Randomly read pages from the allocation and map again. Each thread follows a different sequence.
 - Each thread reads/writes to a fixed offset within the page that depends solely on the thread index.
 - At the end of each stage, one thread will count the number of resident pages before all threads proceed to the next stage.

The grader exits upon any of the following:

- Wrong value seen on read.
- Allocation or map failure.
- Number of resident pages seen at the end of a stage exceeds 2106 (the default LORM).
- 1 minute elapses.

0.4 points are awarded for each complete stage (all 8 threads complete the stage before any error or crash), and 0.2 points are awarded for an incomplete stage (some threads complete the stage before an error is detected).

To run the grader

Compile your file with the grader file as follows:

```
gcc -g -std=c18 -O2 -pthread -o run_bonus lab4-bonus-grader.c userswap.c
```

Then run the resulting executable. When a thread completes a stage, the corresponding stage number is printed.