

CS 130 Operating Systems 1

Homework Assignment #1

(Due at 11:59pm Nov. 2nd, 2020)

- Please type the solutions using WORD, LaTeX, etc, or write by hand very neatly and legibly, comparable to typing
- Please pay special attention to the DUE DATE – no late turn in or special case consideration
- Please submit your homework as a PDF file to Gradescope
- Please work on your homework individually
- The file name should be in a form of “id-YourName-hw1”, for example “12345-ZhangSan-hw1.pdf”

1. [15 points] What are the THREE types of user-mode to kernel-mode transfers? (3 short phrases are enough, you don't need to provide paragraphs to describe them.)

Interrupt, processor exceptions (or exceptions), system calls (or syscal) (5 points each)

2. [10 points] What is an Atomic Operation? Please name two operations that have to be atomic operations?

Indivisible operations that cannot be interleaved with or split by other operations (4 points)

Load/ store (3 points each)

3. [15 points] Can you please briefly explain why a thread is faster to create than a process?

Process creation is a resource intensive operation, in terms of memory allocation and also inter process communication is also pretty much expensive when they need to share data.

4. [20 points] Consider an online reading server that can support at MAX 500 readers concurrently to read. How can you add

semaphores to the following code to ensure a strict limit of 500 readers connected at a time? Assume that this server can create semaphores and share them amongst the reader threads.

```
void read_session(struct server s){
    login{s};
    read{};
    logout{s};
}
```

Introduce a semaphore for each server, initialized to 500, to control the ability to login to the server. A reader will wait() the semaphore before connecting, and signal() the semaphore after disconnecting.

The **order here is important** – wait() the semaphore after login but before reading means that there is no block on the login() call, and signal() the semaphore before logout could lead to "zombie" reader, who were pre-empted before logout. Both of these cases mean that the limit of 500 could be violated.

5. [40 points] Five batch jobs (A, B, C, D, and E), arrive at a computer center at almost the same time. They have estimated running times of 20, 12, 4, 8, and 16 minutes. Their (externally determined) priorities are 3, 5, 2, 1, and 4, respectively, with 5 being the highest priority. For each of the following scheduling algorithms, determine the mean process turnaround time. Ignore process switching overhead.

- (a) [10 points] Round robin.
- (b) [10 points] Priority scheduling.
- (c) [10 points] First-Come, First-Served (run in order 20, 12, 4, 8, 16).
- (d) [10 points] Shortest job first.

For (a), assume that the system is multi-programmed, and that each job gets its fair share of the CPU.

For (b), (c), and (d), assume that only one job at a time runs, until finishes.

All jobs are completely CPU bound.