

CSE 4/521 - Introduction to Operating Systems  
Summer 2018  
Homework 1

**Objective:**

1. This homework is designed to analyze the reading comprehension ability of a student in browsing through and understanding the PintOS source code.
2. Team submission is advocated in-order to promote better communication of technical concepts among team members, develop a sense-of-oneness, help understand each other's ability, and be responsible for each other in a group setting.

*Total Points:* 100 (+15 points bonus)

*Submission deadline:* Wednesday- June 20, 2018 (Inclass)

*Submission guidelines:* One submission per team

*Grading criteria:* All members of the team receive the same score

2 bonus questions in this homework to challenge you to get more than 100 points.

You can (add/negate) 5 homework buffer days of this semester towards the submission of this homework.

Please note the academic integrity policy at: <http://academicintegrity.buffalo.edu/policies>

For pintos documentation that would help answer these questions, refer the extensive documentation at: <https://web.stanford.edu/class/cs140/projects/pintos/pintos.html> . But the most viable way to answer the questions in a time-efficient manner is to read and run the source code and by discussing these questions with your team members. The reading comprehension ability of the source code is an essential requirement before beginning Project/Checkpoint 1.

1. In PintOS, how do you write your own function and have it invoked on all threads? Write down an example. (15)
2. What is the function of 'timer' in PintOS? (5)
3. What are interrupt handlers? What does timer\_interrupt handler do in PintOS? (5)
4. What is the function call for threads in PintOS to block itself from executing? Similarly, what is the call to wake a thread up? (10)
5. Name the different states of a process as discussed in class. Similarly, what are the different states of a thread in PintOS? (5)
6. How would you avoid 'busy waiting' in timer\_sleep() function in devices/timer.c? Explain your intuition. (15)
7. What does alarm-multiple test do? (15)
8. How many tests are currently passing in the threads directory of PintOS? (You might want to run make check) (5)
9. What are external interrupts in PintOS? Give one example of external interrupt in PintOS and the part of the source code where you found it. What are their special characteristics in-terms of pre-emption and nesting? (5)
10. How many interrupts found in PintOS? Name 5 of them. Where did you find them? (10)
11. How many processors are simulated in PintOS? Would inhibiting interrupts for critical section execution work in PintOS environment? Explain. (10)
12. (Bonus) What are the advantages of spinlock over mutual exclusion? (10)
13. (Bonus) Is PintOS a pre-emptive or non-preemptive kernel? Explain your understanding. (5)