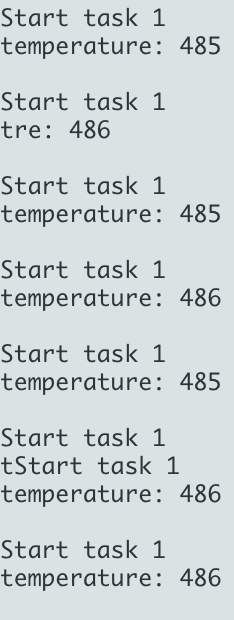
CS 244P demo8 report:

a)

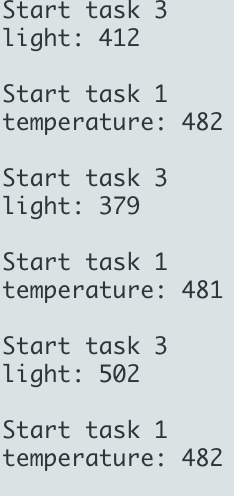
when don’t set delay, the result is:



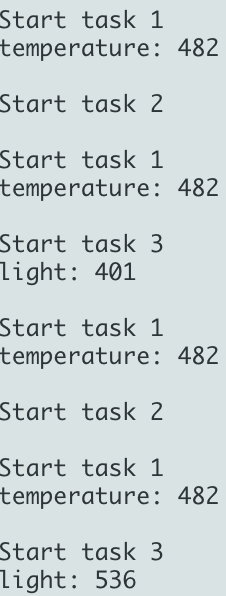
It keeps in executing task1, because it has the highest priority: Before the first time, all 3 tasks is ready and task1 will get the SemaphoreHandle\_t and execute, after it frees SemaphoreHandle\_t and all 3 tasks will again apply for the SemaphoreHandle\_t, and again the task1 will get it because it has the highest priority.

b)

when setting delay with vTaskDelay(1), it will only execute task 1 and task 3 which are two of highest priority tasks: Before the first time, all 3 tasks is ready and task 1 will get the SemaphoreHandle\_t and execute, after it frees SemaphoreHandle\_t and delay for 1ms then only task 2 and task 3 are ready(cause task1 been blocked) so the task 3 will get the SemaphoreHandle\_t and execute. The task 1 only delay for 1 ms so task 1 will be ready before task 3 frees SemaphoreHandle\_t. Thus after task 3, the task 1 will get the SemaphoreHandle\_t again.



c)



when setting priority of task 2 to 2, which is same with task 3. It will execute both 3 tasks: Also, before the first time, all 3 tasks are ready and task1 will be executed because it has highest priority. After that, task1 delays and task 2 and task 3 have same priority so it will use a time sliced round robin scheduling scheme to executed these two tasks. According to the Round Robin Schedule, the schedule will give jobs with same priority same time period to execute, so the task 2 and task 3 will have equal chance to been executed.