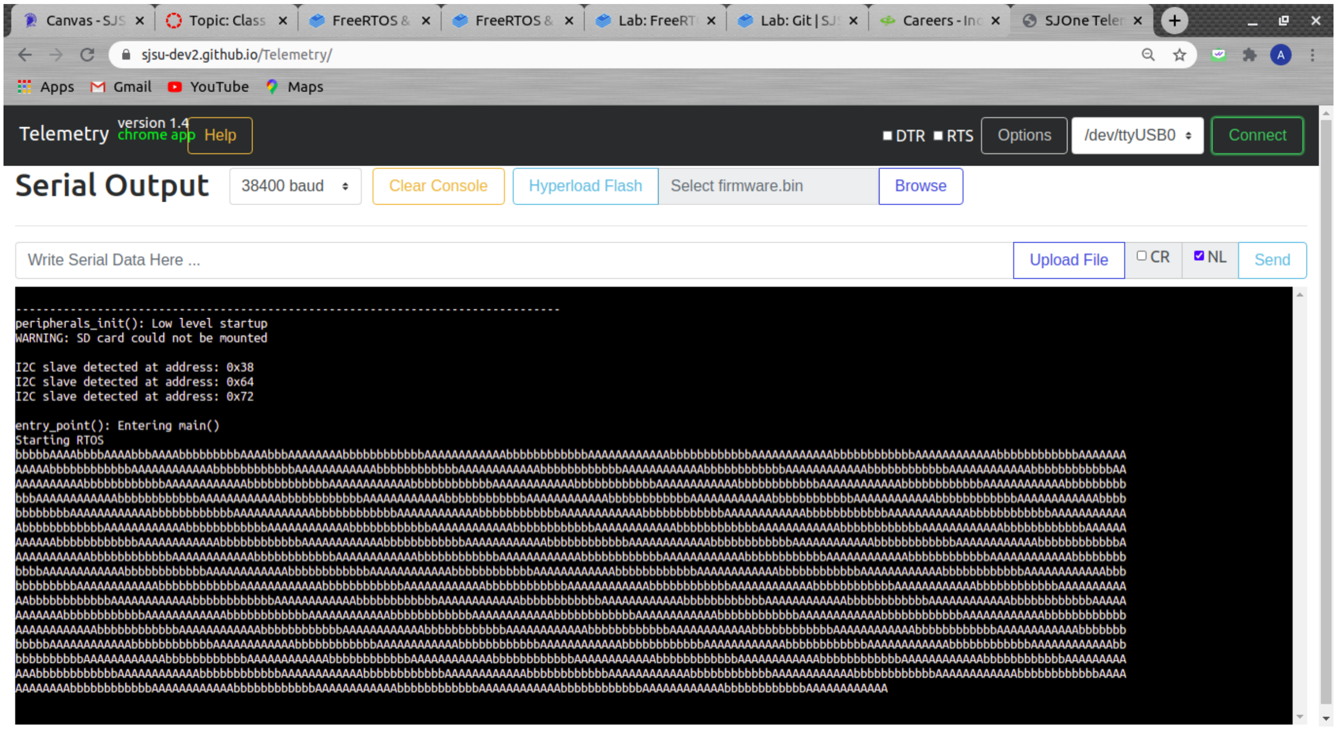
Name: Akshat Bhutiani
Student ID: 014538238

Output for same priority tasks -



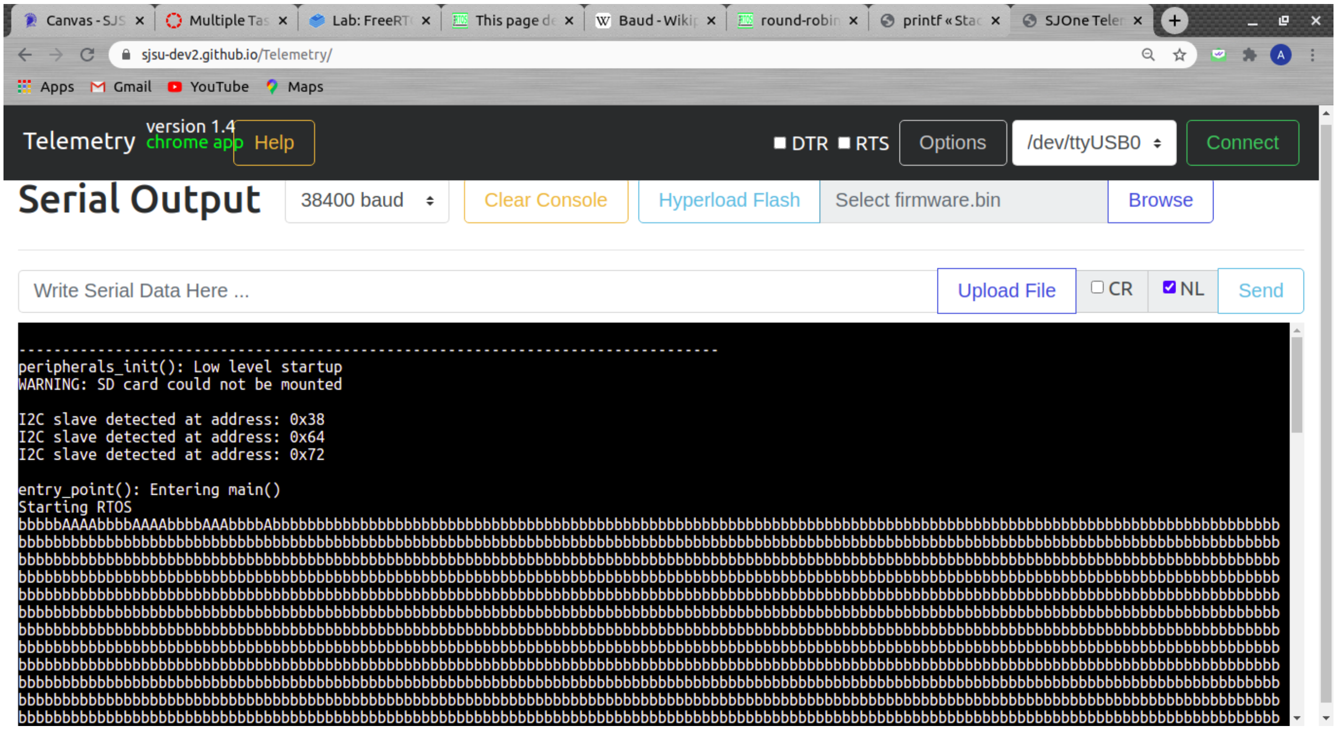
```
--------------------------------------------------------------------------------
peripherals_init(): Low level startup
WARNING: SD card could not be mounted

I2C slave detected at address: 0x38
I2C slave detected at address: 0x64
I2C slave detected at address: 0x72

entry_point(): Entering main()
Starting RTOS
bbbbbAAAAbbbbAAAAbbbAAAAbbbbbbbbbAAAAbbbAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAA
AAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAA
AAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbb
bbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbb
bbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAA
AbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAA
AAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbA
AAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbb
bbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbb
bbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAA
AAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAA
AAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbb
AAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbb
bbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbb
bbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAA
AAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAA
AAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAA
```

Explanation of the output -

Assume that our scheduler randomly selects task 1 (“A”) for execution. The tick rate of our scheduler is 1 ms. The speed of our UART is 3840 characters/second and hence in 1ms it would print 3840/1000 = 3.8 characters (approx).

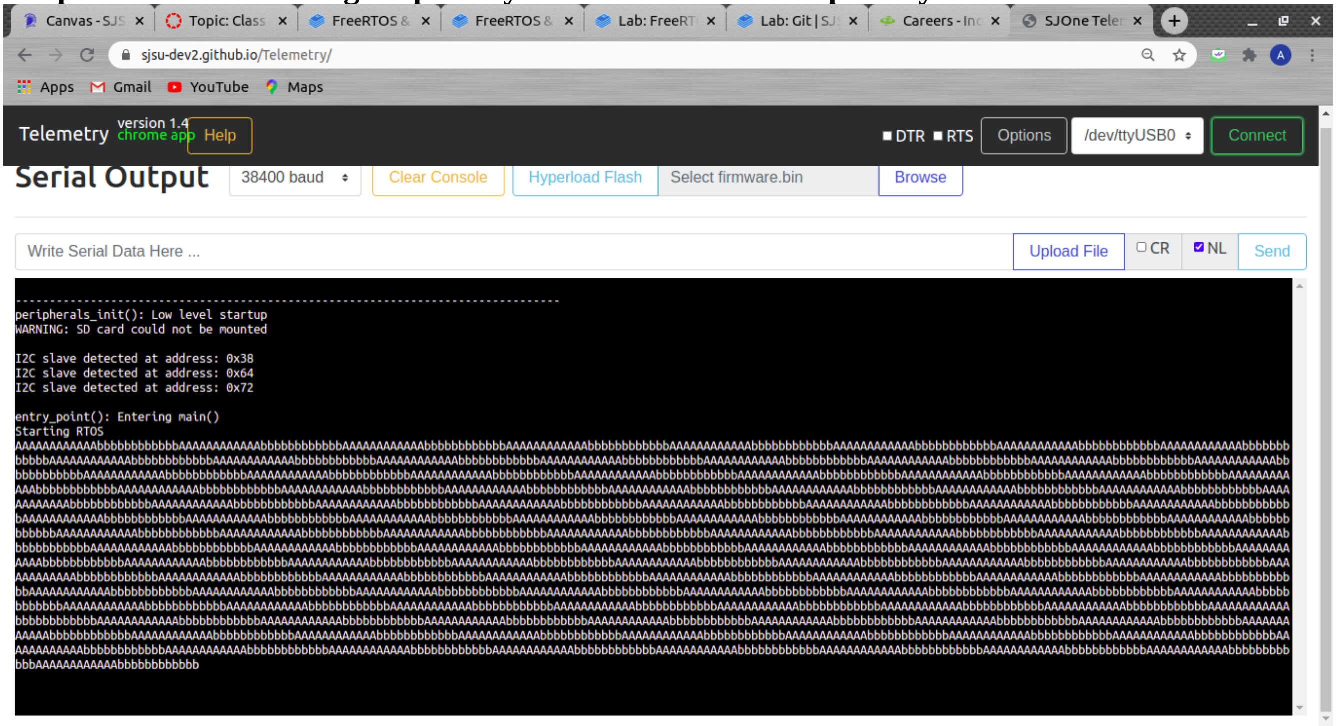
Now task 2 (“B”) comes and prints approximately 3.8 characters. This cycle continues. Eventually 1 of the tasks reaches 12 characters (length of string) and hence the next instruction that the compiler encounters is vTaskDelay(100). The task in which vTaskDelay() is encountered will sleep for 100 ms and the other task will run. This cycle continues till the memory is full.

To prove this theory, I set vTaskDelay for Task 1(“A”) as 1000 and I removed vTaskDelay() from the second task(“B”). This would mean that after 12 A’s are printed, we would see a steady string of B’s. This is indeed what happened. I have attached proof for your reference -



```
--------------------------------------------------------------------------------
peripherals_init(): Low level startup
WARNING: SD card could not be mounted

I2C slave detected at address: 0x38
I2C slave detected at address: 0x64
I2C slave detected at address: 0x72

entry_point(): Entering main()
Starting RTOS
bbbbbAAAAbbbbAAAAbbbbAAAbbbbAbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbb
bbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbb
bbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbb
bbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbb
bbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbb
bbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbb
bbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbb
bbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbb
bbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbb
bbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbb
bbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbb
bbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbb
```

Output for task 1 with higher priority and task 2 with lower priority -

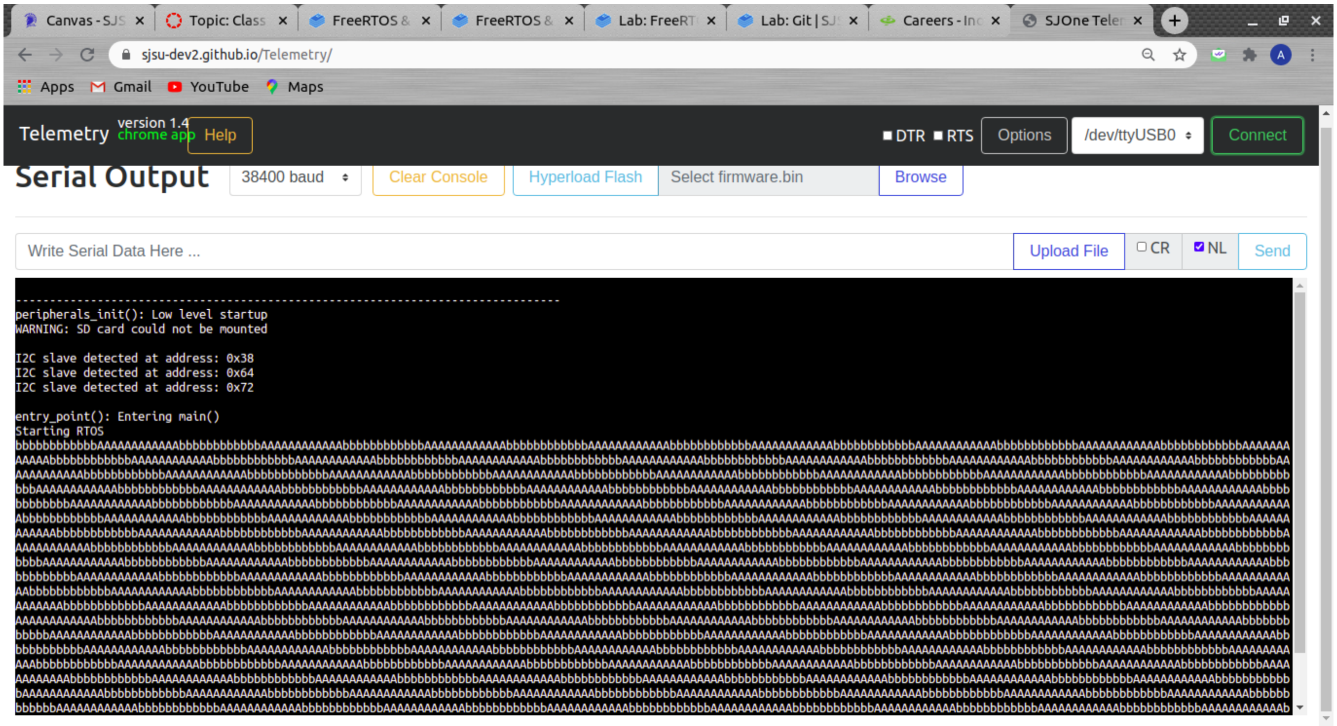


```
--------------------------------------------------------------------------------
peripherals_init(): Low level startup
WARNING: SD card could not be mounted

I2C slave detected at address: 0x38
I2C slave detected at address: 0x64
I2C slave detected at address: 0x72

entry_point(): Entering main()
Starting RTOS
AAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbb
bbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbb
bbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAA
AAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAA
AAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbb
bAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbb
bbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbb
bbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAA
AAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAA
AAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbb
bbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbb
bbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAb
bbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAA
AAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAA
AAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbbbbAAAAAAAAAAAbbbbbbbbb
bbbAAAAAAAAAAAbbbbbbbbbbb
```

Explanation -

Since task 1 is at higher priority, it starts 1st and runs till completion. Then vTaskDelay(100) is encountered . At this time, task 1 goes to sleep and task 2 is executed. So task 2 also runs to completion. Then task 2 sleeps and the cycle continues.

Output for task 2 having higher priority and task 1 having low priority -



38400 baud

/dev/ttyUSB0

☐ CR ☑ NL

```
--------------------------------------------------------------------------------
peripherals_init(): Low level startup
WARNING: SD card could not be mounted

I2C slave detected at address: 0x38
I2C slave detected at address: 0x64
I2C slave detected at address: 0x72

entry_point(): Entering main()
Starting RTOS
bbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAA
AAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAA
AAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbb
bbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbb
bbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAA
AbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAA
AAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbA
AAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbb
bbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbb
bbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAA
AAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAA
AAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbb
AAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbb
bbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbb
bbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAA
AAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAA
AAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbb
bAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbb
bbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAbbbbbbbbbbbbAAAAAAAAAAAAb
```

Explanation -

Since task 2 is at a higher priority, it is scheduled first and runs till completion. Once task 2 is complete, vTaskDelay(100) of Task 2 is called and task 2 goes to sleep. Now task 1 is called and task 1 runs till completion. Then task 1 sleeps and the cycle repeats.