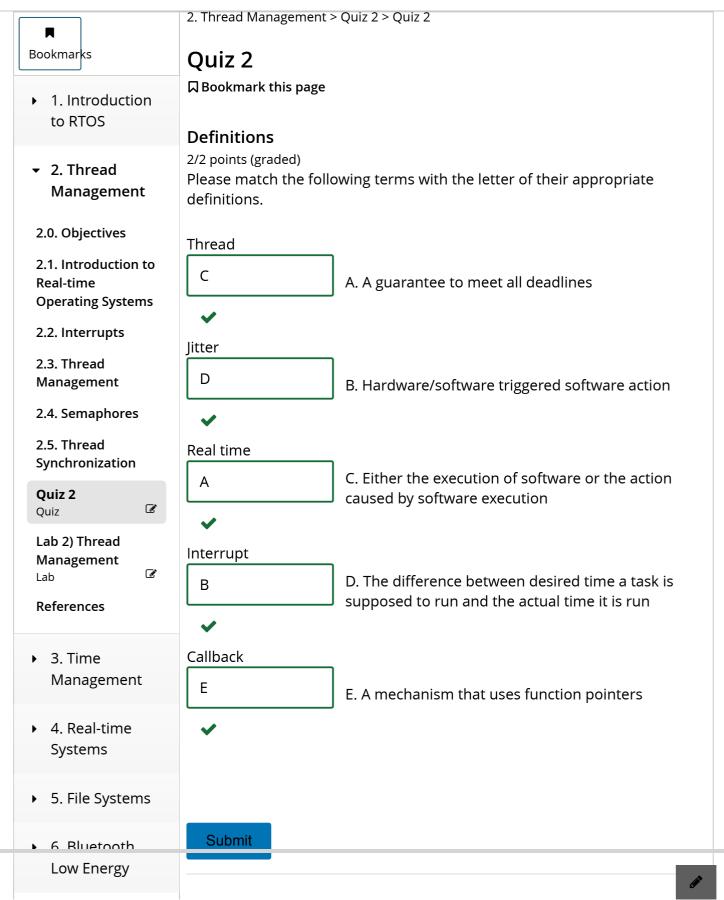


### UTAustinX: UT.RTBN.12.01x Realtime Bluetooth Networks

Help



### Glossary

# DiscussionBoard

### **Schedulers**

2/2 points (graded)

Please match the following terms with the letter of their appropriate definitions.

### Round robin



A. A dynamic scheduler that shifts importance depending on if the thread ran to the completition of its time slice.

### Priority



B. Run the ready threads in circular fashion, giving each the same amount of time to execute

## **~**

### Exponential queue



C. Assign importance according to these periods with more frequent tasks having higher importance.

# Rate Monotonic



D. Threads themselves decide when to stop running



### Cooperative



E. Run the most important ready threads first, running less important threads only if there are no important threads ready



### Submit

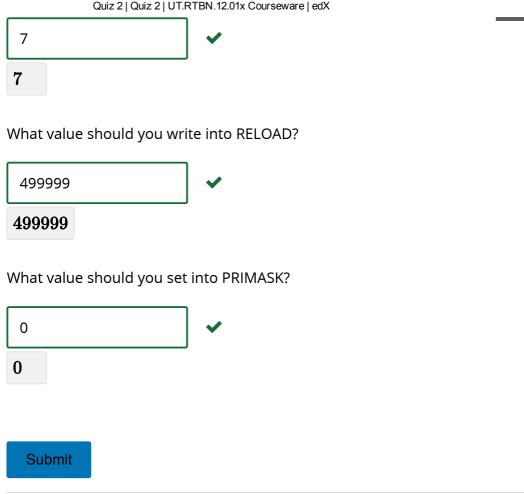
### SysTick Interrupts

2/2 points (graded)

Assume the bus clock is 50 MHz or 20 ns. You wish to configure SysTick to interrupt every 10ms.

What value should you write into STCTRL?





# Bug or no bug

2/2 points (graded)

We can use semaphores to limit access to resources. In the following example both threads need access to a printer and an SPI port. The binary semaphore **sPrint** provides mutual exclusive access to the printer and the binary semaphore **sSPI** provides mutual exclusive access to the SPI port. Consider the following scenario to see if it has any bugs.

```
Thread 1
                                Thread 2
bwait(&sPrint);
                                bwait(&sSPI);
bwait(&sSPI);
                                bwait(&sPrint);
OutSPI(4);
                                OutSPI(5);
printf("Hasta luego");
                                printf("tchau");
OutSPI(6);
                                OutSPI(7);
bsignal(&sPrint);
                                bsignal(&sSPI);
bsignal(&sSPI);
                                bsignal(&sPrint);
```

Does the above scenario have bugs?

Yes there is a bug, the two **OutSPI** should be together



Quiz 2   Quiz 2   0 1.11 BN. 12.01X Oodi Seware   cux
No bug per se, but the semaphores are not needed
Yes there is a bug, this has a possible deadlock
Yes there is a bug, there are two semaphores and three outputs, one more semaphore is needed
Yes there is a bug, the order of the two signals needs to be switched
No, this code has no bugs
Three tasks  1/1 point (graded)  You have three tasks. Task 1 takes a maximum of 1 ms to execute and runs every 10 ms. Task 2 takes a maximum of 0.5 ms to execute and runs every 1 ms. Task 3 takes a maximum of 1 ms to execute and runs every 100 ms.  Is there a possible scheduling algorithm for these three tasks?
No, there is not enough time
Maybe, but we need to know more information
● Yes ✔
There is no way to tell without running it
Submit
Submit

### Four tasks

1/1 point (graded)

You have four tasks. Task 1 takes a maximum of 1 ms to execute and runs every 5 ms. Task 2 takes a maximum of 0.5 ms to execute and runs every 2 ms. Task 3 takes a maximum of 1 ms to execute and runs every 20 ms. Task 4 takes a maximum of 6 ms to execute and runs every 10 ms.

Is there a possible scheduling algorithm for these three tasks?

- No, there is not enough time
- Maybe, but we need to know more information
- Yes
- There is no way to tell without running it

#### **Answer**

Correct: think about the total time each tasks runs in 1 second

Submit

© All Rights Reserved



© 2016 edX Inc. All rights reserved except where noted. EdX, Open edX and the edX and Open EdX logos are registered trademarks or trademarks of edX Inc.



















