Solutions

Midterm 1

Answer in space provided in booklet and submit booklet only.

Please Read:

INSTRUCTIONS: Please work on this midterm independently. The midterm is open book. Use of books, notes, calculators, laptops, and Internet resources is allowed. Collaboration is ***not allowed*** (which includes use of electronic communication such as chat, Skype, etc). Please read each question carefully and answer what the question asks for. Do not assume that the answer necessarily needs you to use all the data in the question. One of the hallmarks of knowing your field is to know which pieces of information are pertinent to the problem.

Good Luck!

Midterm 1

Q1 (Data and system reliability, independent versus non-independent failures, etc): A probe is sent to explore the chemical composition of an asteroid that approaches Earth. The probe lands on the asteroid and records scientific observations in a local database that it will bring back to Earth. The database stores all recorded observations in four independent copies, each on a different physical storage device. These devices are extremely reliable and will not alter, miss, or incorrectly record the data. The *only* failure mode to worry about is if the protective shield of a device is punctured by a physical collision with falling debris around the asteroid in which case all data on the device is lost. The probability that a single device is punctured with falling debris is 1/500 per day of contact with the asteroid.

a) How many storage devices must survive and return to Earth in order for it to be possible to retrieve all data correctly? **(1 point)**

1

b) If the probe remains in contact with the asteroid for 3 days, what is the probability that scientists are unable to retrieve data from the probe upon return to Earth (assuming device failures are independent)? **(2 points)**

$[1-(499/500)^3]^4 = 1.28 * 10^{-9}$

c) In reality failures are not independent. Further analysis reveals that the probability of experiencing a shower of falling debris around the probe on any one day is 1/250. Moreover, the probability that any one storage device is punctured due to collision with such debris, given that such a shower occurs, is 50%. If there was no other way for the device to fail, what is the probability that scientists are unable to retrieve any data from the probe upon return to Earth? As before, assume that the probe remains in contact with the asteroid for 3 days. **(2 points)**

Approx. $= 1-(1-(1/250)(0.5)^4)^3 = 0.00075$

Exact $= 3(1/250)(1-0.5)^4 + 3(1/250)(1/250)\,(1 – (0.5)^2)^4 + (1/250)(1/250)(1/250)(1 – (0.5)^3)^4$
$= 0.000765$

Q2 (Scheduling): For each of the following task sets, please indicate (by circling the right answer) whether the task set is schedulable or not using *EDF* scheduling. In the task sets below, Pi refers to the period of task i, Ci refers to the execution time of task i. All times are in seconds. The tasks's relative deadline, Di, is equal to Pi. **(5 points)**

1. Task set #1: (Schedulable) (Not)

P1=9, C1=4
P2=100, C2=14
P3=10, C3=6
P4=20, C4=1

2. Task set #2: (Schedulable) (Not)

P1=10, C1=2
P2=50, C2=45

3. Task set #3: (Schedulable) (Not)

P1=15, C1=3
P2=40, C2=1

4. Task set #4: (Schedulable) (Not)

P1=50, C1=3
P2=5, C2=0.5
P3=21, C3=10

5. Task set #5: (Schedulable) (Not)

P1=30, C1=10. The task includes a critical section of size 2.
P2=40, C2=4. The task includes a critical section of size 2.

Q3 (Blocking): Consider the table below, where rows indicate tasks (smaller task numbers imply higher priority) and columns indicate resources. A cell at row X and column Y is set to 1 if task X uses resource Y. Each resource is protected by its own semaphore. When a task needs resource Y, it executes a Lock(Y) operation. When it is done, it executes Unlock(Y). The priority *ceiling* algorithm is used together with *rate monotonic* scheduling. Indicate (by circling the right answer) which of the lock/unlock sequences below are possible and which are impossible. Assume that each sequence represents all lock/unlock operations that (presumably) occurred. Assume that no other blocking occurs except on the semaphores below. **(5 points)**

	Resource R1	Resource R2	Resource R3	Resource R4	Resource R5
Task T1		1			1
Task T2		1		1	
Task T3	1				
Task T4		1	1		1

a) T3 locks R1, T2 locks R4, T3 unlocks R1, T2 unlocks R4. (Possible) (Not)

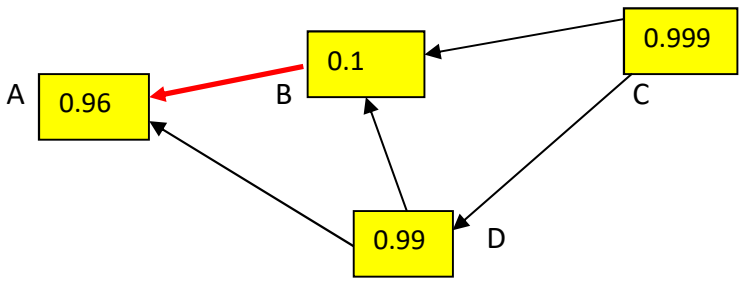
b) T3 locks R1, T2 locks R4, T2 unlocks R4, T3 unlocks R1. (Possible) (Not)

c) T4 locks R2, T4 unlocks R2, T2 locks R4, T2 unlocks R4. (Possible) (Not)

d) T4 locks R2, T2 locks R4, T2 unlocks R4, T4 unlocks R2. (Possible) (Not)

e) T1 locks R2, T2 locks R4, T2 unlocks R4, T1 unlocks R2. (Possible) (Not)

Q4 (Well-formed dependencies): In the diagram below, each box represents a component. A link from component X to component Y indicates that Y depends on X. In other words, a failure in X causes a failure in Y. Each box is labeled by its reliability, when executed independently.



(3 points)

a) Assuming that safety-critical components are more reliable, and that non-critical components are less reliable, are dependencies in the above system well-formed? (Only Yes/No please). If "Yes", skip part **b**.

Answer: _____No_____

b) If your answer above was "No", clearly mark in the figure the link or links that violate(s) well-formed dependencies. (Put a clear "**X**" in the middle of each such link.) See fig.

Q5 (Safety envelopes): A robot's camera can distinguish obstacles when they are within 10ft (or closer) from the robot. It takes 1s to stop the robot once a stop command is issued. A single-core microcontroller runs the robot software. If the time it takes to process a camera frame on this core (to determine whether or not an obstacle is present) is 3s. What is the maximum safe speed of the robot so it can always stop on time when an obstacle is detected? **(2 points)**

Period must be at least equal to processing time = 3.

In the worst case the robot will stop 7 seconds after entering range (3 before picture is taken, then 3 to process it, then 1 to stop).

Max speed = 10/7 = 1.4286

Thank you!
