Midterm Review

PART I

Reliability

Reliability

- Reliability for a giving mission duration t, $R(t)$, is the probability of the system working as specified (i.e., probability of no failures) for a duration that is at least as long as t.
- The most commonly used reliability function is the exponential reliability function:

$$R(t) = e^{-\lambda t}$$

From queueing theory: Probability of zero independent arrivals in t time units (Poisson arrival process)

where λ is the failure rate.

Reliability

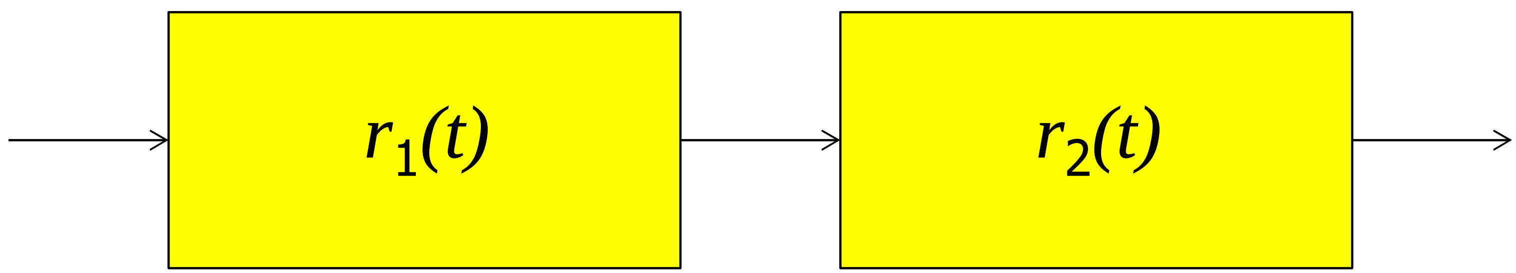
- The most commonly used reliability function is the exponential reliability function:

$$R(t) = e^{-\lambda t}$$

where λ is the failure rate.

- Mean time to failure (MTTF): $1/\lambda$

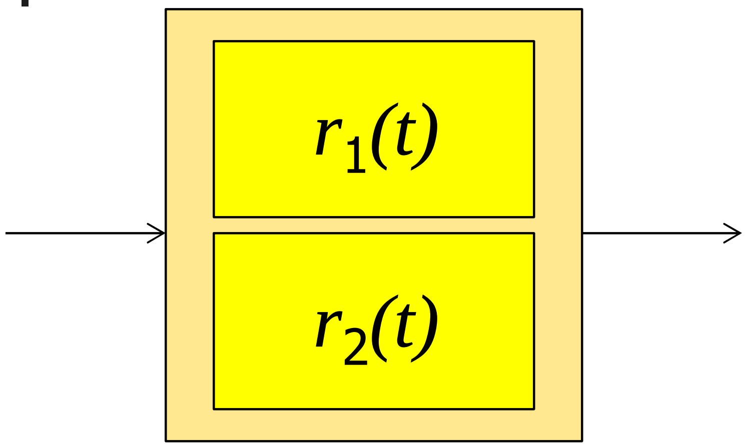
Simple Reliability Modeling



- Total failure rate = $\lambda_1 + \lambda_2$
- Mean time to failure = $1/(\lambda_1 + \lambda_2)$
- Total reliability:

$$R(t) = r_1(t) r_2(t) = e^{-(\lambda_1 + \lambda_2)t}$$

Simple Reliability Modeling

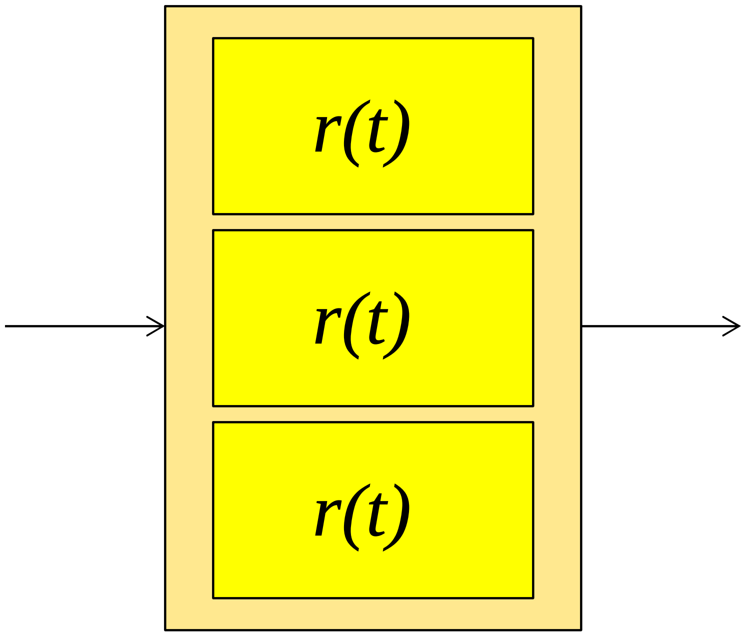


Note: This system needs at least one of the two components to function.

- Total reliability:

$$R(t) = 1 - (1 - r_1(t))(1 - r_2(t))$$

Triple Modular Redundancy



Note: This system needs at least two of the three components to function.

- Total reliability:

$$R(t) = r^3(t) + 3r^2(t)(1 - r(t))$$

Other Implications

$$R(Effort, Complexity, t) = e^{-kC\,t/E}$$

- Note: splitting the effort greatly reduces reliability.

Well Formed Dependencies

- *Informal intuition:* A reliable component should not *depend* on a less reliable component (it defeats the purpose).

- Design guideline: **Use but do not depend** on less reliable components

Review of Important Theorems

- Total Probability Theorem:

  $P(A) = P(A|C_1)\ P(C_1) + \ldots + P(A|C_n)\ P(C_n)$

  where $C_1, \ldots, C_n$ partition the space of all possibilities

- Bayes Theorem:

  $P(A|B) = P(B|A).\ P(A)/P(B)$

- Other: $P(A,B) = P(A|B)\ P(B)$

Two Sensor Example

- Remember: If burglar enters, motion alarm fires 99% of the time and vibration alarm fires 90% of the time. Burglaries occur once a year, motion alarm fires 3 times a year, and vibration alarm fires 10 times a year.
- What are the odds of burglary if both sensors fire?
- P (Burg|A, Vib) = ?
- P (B|A,V) = P(A,V|B) P(B)/P(A,V)

Now what?

OK to say $P(A,V|B) = P(A|B)P(V|B)$

$\cancel{P(A,V) = P(A)P(V)?}$

Remember: If burglar enters, motion alarm fires 99% of the time and vibration alarm fires 90% of the time. Burglaries occur once a year, motion alarm fires 3 times a year, and vibration alarm fires 10 times a year.

Two Sensor Example

- P (Burg|A, Vib) Solution steps:
  - Find the probability of false alarms from:

    $P(A) = P(A|B)\,P(B) + P(A|\overline{B})\,P(\overline{B})$

    $P(V) = P(V|B)\,P(B) + P(V|\overline{B})\,P(\overline{B})$
  - Find the probability of both sensors firing:

    $P(A,V) = P(A,V|B)\,P(B) + P(A,V|\overline{B})\,P(\overline{B})$

    where $P(A,V|B) = P(A|B)P(V|B)$

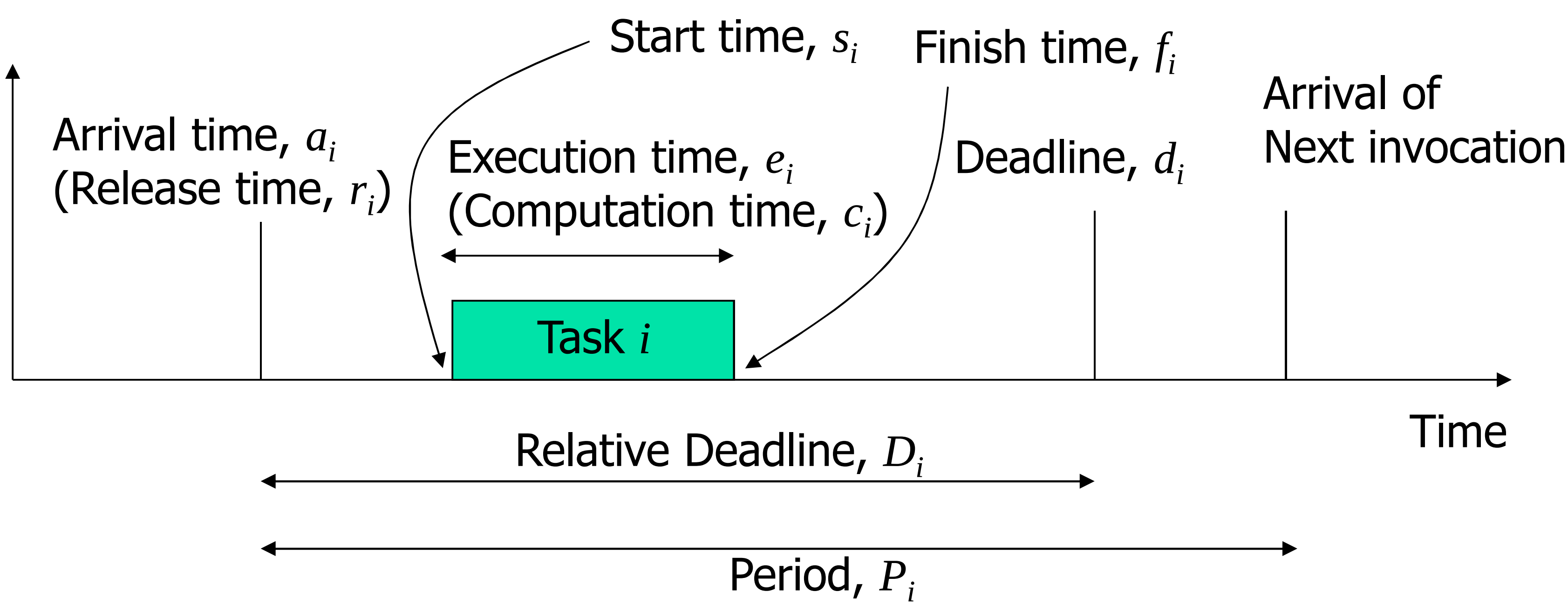
    $P(A,V|\overline{B}) = P(A|\overline{B})P(V|\overline{B})$
  - $P\,(B|A,V) = P(A,V|B)\,P(B)/P(A,V) = 94.62\%$

PART II

Timeliness

Some Terminology

- Tasks, periods, arrival-time, deadline, execution time, etc.



The Schedulability Condition

For n independent periodic tasks with periods equal to deadlines:

The utilization bound of EDF = 1.

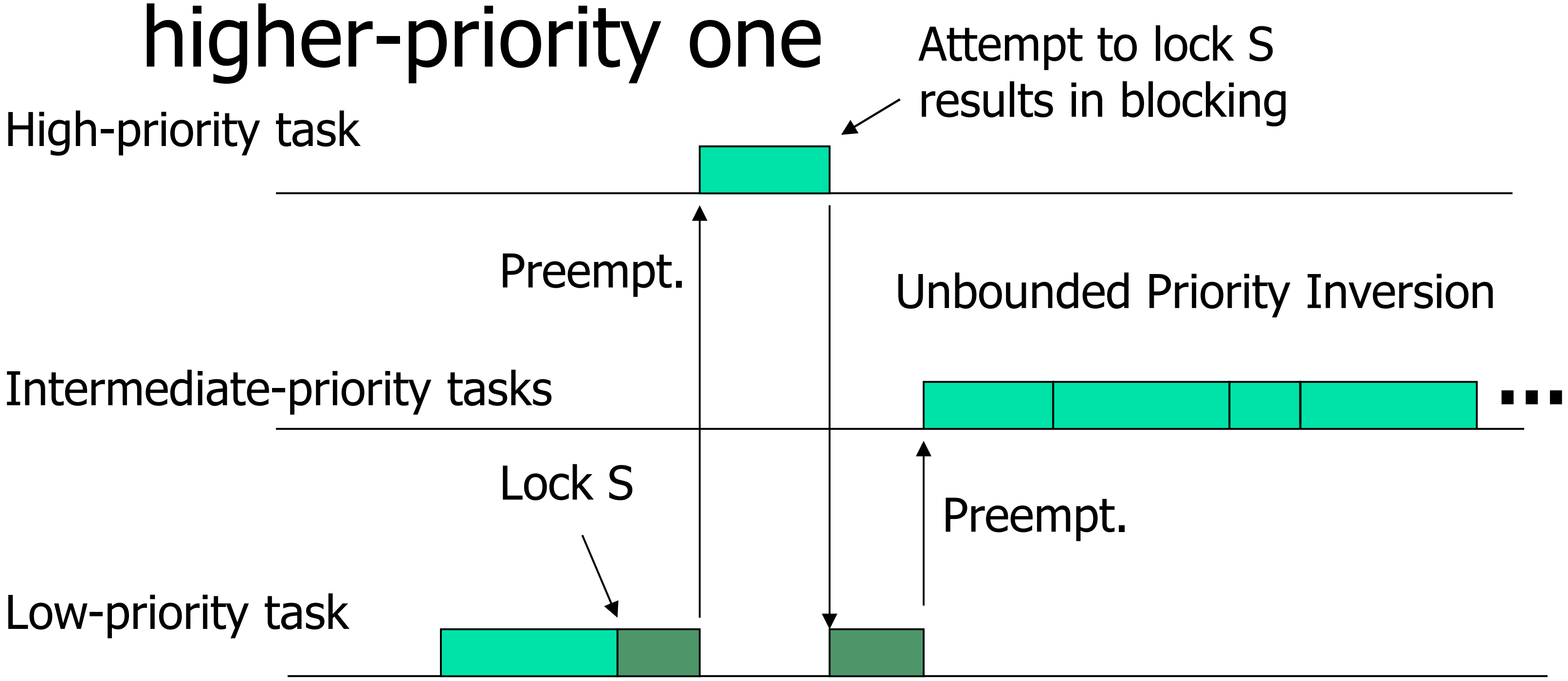
The Utilization bound of RM is:

$$U = n\left(2^{1/n} - 1\right)$$

$$n \rightarrow \infty \quad U \rightarrow \ln 2$$

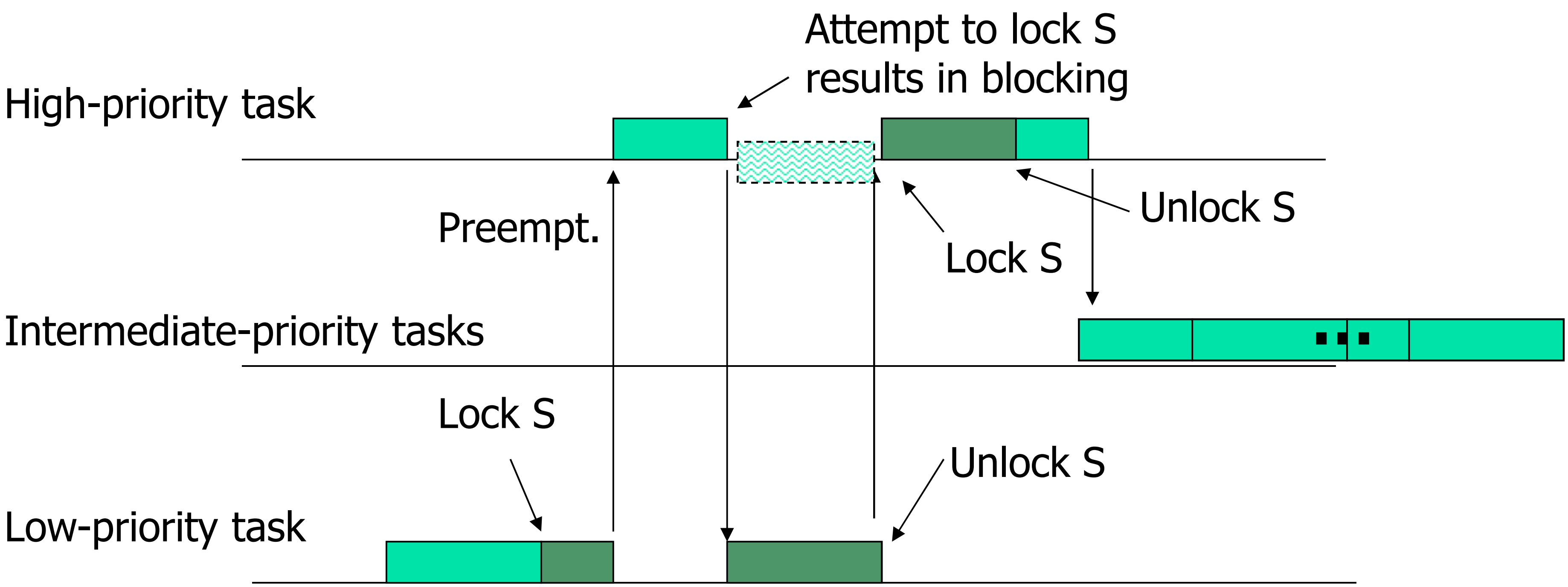
Blocking and Priority Inversion

- Consider the case below: a series of intermediate priority tasks is delaying a higher-priority one



Priority Inheritance Protocol

- Let a task inherit the priority of any higher-priority task it is blocking



Maximum Blocking Time

- If all critical sections are equal (of length B):
  - Blocking time = $B \min(N, M)$ (Why?)
- If they are not equal
  - Find the worst (maximum length) critical section for each resource
  - Add up the top $\min(N, M)$ sections in size
- The total priority inversion time for task i is called B_i

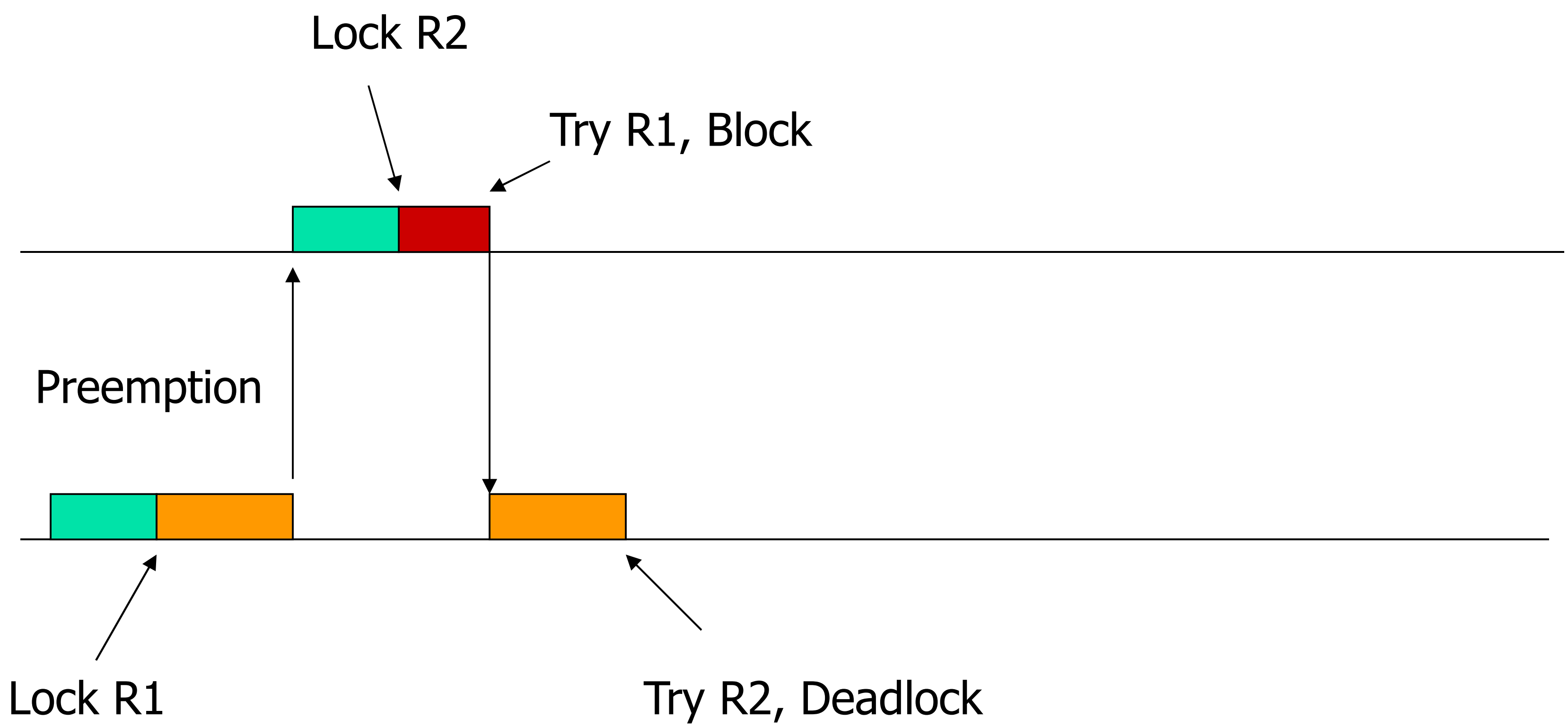
Schedulability Test

$$\forall i, 1 \le i \le n,$$

$$\frac{B_i}{P_i} + \sum_{k=1}^{i} \frac{C_k}{P_k} \le i(2^{1/i} - 1)$$

Problem: Deadlock

Deadlock occurs if two tasks locked two semaphores in opposite order



Priority Ceiling Protocol

- Definition: The priority ceiling of a semaphore is the highest priority of any task that can lock it
- A task that requests a lock R_k is denied if its priority is not higher than the highest priority ceiling of all currently locked semaphores (say it belongs to semaphore R_h)
  - The task is said to be blocked by the task holding lock R_h
- A task inherits the priority of the top higher-priority task it is blocking

Practice Question #1

- The probability that a window breaks in a house on any one day is 1/10,000, except when there is a hurricane.
- The probability that a window breaks during a hurricane is 0.3
- The probability that a hurricane passes nearby on any given day is 1/1000
- What are the odds that all 6 windows in Jeff's house break on the same day?

Practice Question #1

- The probability that a window breaks in a house on any one day is 1/10,000, except when there is a hurricane.
- The probability that a window breaks during a hurricane is 0.3
- The probability that a hurricane passes nearby on any given day is 1/1000
- What are the odds that all 6 windows in Jeff's house break on the same day?

- Answer: $1/1000 * (0.3)^6 + 999/1000\ (1/10{,}000)^6$

  $= 1/1000 * (0.3)^6$ (approx.)

Practice Question #2

- The probability of falling debris on planet X is 1/500. The probability that a storage device on a robot breaks when there is falling debris is 0.5. What is the probability that all 4 devices break? (Assume there is no other way for these devices to break.)

Practice Question #2

- The probability of falling debris on planet X is 1/500. The probability that a storage device on a robot breaks when there is falling debris is 0.5. What is the probability that all 4 devices break? (Assume there is no other way for these devices to break.)

- Answer: $1/500\ (0.5)^4$

Observation

- One of the main reasons for failure of large systems is that designers did not properly account for the possibility of correlated failures, and instead viewed them as independent (and hence highly improbably in combination)

Elapsed Time and Reliability

- If the probability of failure within time X is P, what is the probability of failure in time m.X? What is the probability of surviving for time m.X?

Elapsed Time and Reliability

- If the probability of failure within time X is P, what is the probability of failure in time m.X? What is the probability of surviving for time m.X?

- P(surviving time X) = 1 – P
- P(surviving time mX) = $(1 – P)^m$
- P(failure in time mX) = $1 – (1 – P)^m$

Practice Question #3

- The probability of failure on any given day is 1/1000. What is the probability of failure within 5 days?

Practice Question #4

- The probability of failure on any given day is 1/1000. What is the probability of failure within 5 days?
- P(Fail) = 1 – P(Survive all 5 days)

  $= 1 - (0.999)^5 = 1 - 0.995 = 0.005$

Independence versus Mutual Exclusion

- For independent events E1, E2, the probability $P(E1, E2) = P(E1)\, P(E2)$
- For mutually exclusive events E1, E2 the probability $P\,(E1, E2) = 0$.

Practice Question #5

- Is this task set schedulable using RM?
  - P1=15, C1=3
  - P2=40, C2=1

Practice Question #5

- Is this task set schedulable using RM?
  - P1=15, C1=3
  - P2=40, C2=1
  - Answer:
  - $U = 3/15 + 1/40 < \ln(2)$

    → schedulable using RM!

Practice Question #6

- Is this task set schedulable using EDF?
  - P1=10, C1=3
  - P2=200, C2=14
  - P3=40, C3=11
  - P4=19, C4=6

Practice Question #6

- Is this task set schedulable using EDF?
  - P1=10, C1=3
  - P2=200, C2=14
  - P3=40, C3=11
  - P4=19, C4=6

  - U = 3/10+14/200+11/40+6/19 < 1
    → Schedulable using EDF

Priority Ceiling versus Priority Inheritance

- Blocked how many times?
  - Ceiling: once only (worst case: find largest critical section of a lower-priority task)
  - Inheritance: each low priority task holding a resource can block you at most once (note: assuming you need that resource)

Practice Question #7

- How many times will T1 be blocked by lower priority tasks?

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

Practice Question #7

- How many times will T1 be blocked by lower priority tasks?

Priority ceiling → once
Priority inheritance → 3 times

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

Practice Question #8

- What is the worst case blockage scenario for T1 (assume priority inheritance)?

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

Practice Question #8

- What is the worst case blockage scenario for T1 (assume priority inheritance)?

T4 locks R3 → T3 preempts and locks R5 → T2 preempts and locks R2 → T1 preempts and needs R2 then R5 then R3 (blocking each time)

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