TOPIC: Software Design and Development (ENSF409)

Q.1 Which of the following diagrams describe dynamic behavior of the software system?

a. Class diagram and object diagram
b. Use-case diagram and class diagram
c. Collaboration diagram and deployment diagram
d. Sequence diagram and collaboration diagram
e. None of the above

Q.2 Life cycle of a *class* is represented by …

a. Collaboration diagram
b. State Transition diagram
c. Class diagram
d. Deployment diagram
e. None of the above

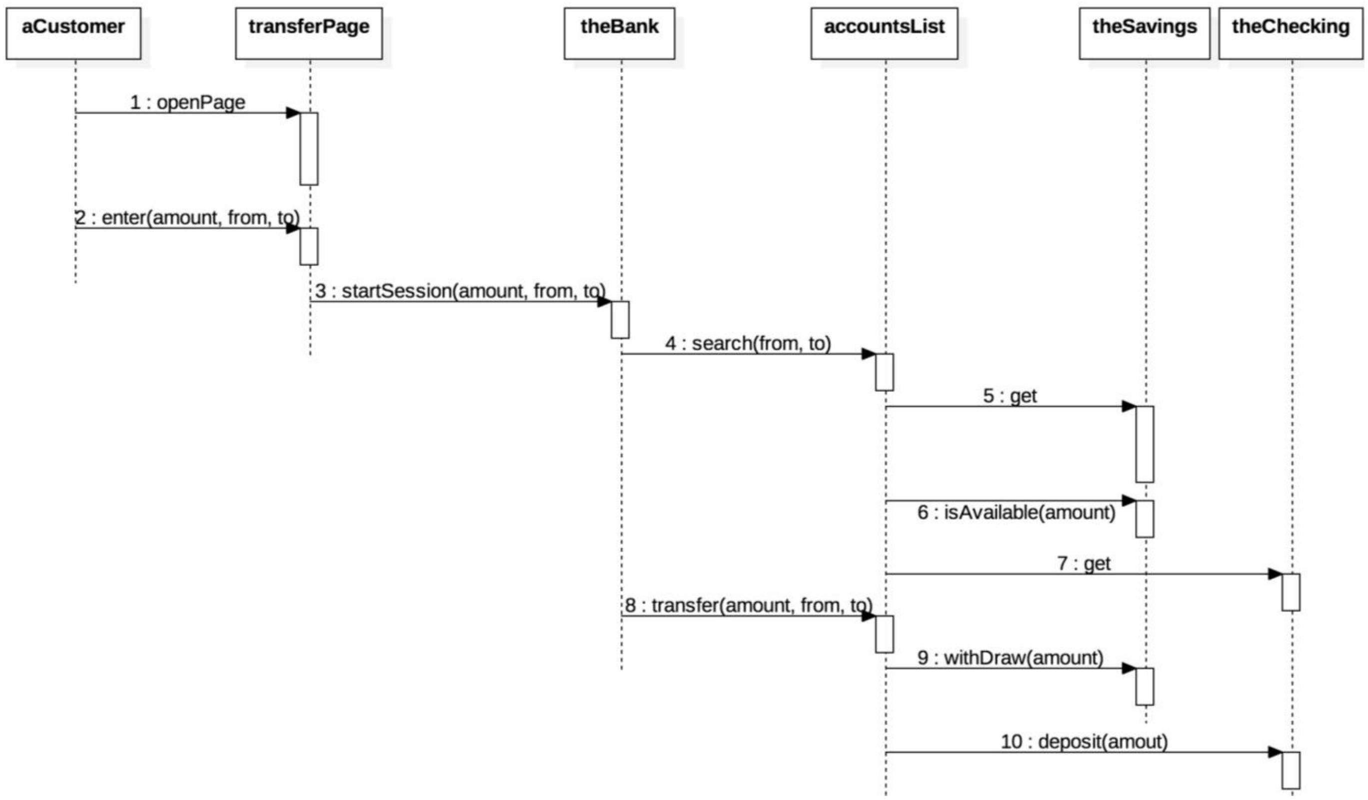
Q.3 An external communication link in a use case diagram might be translated to:

a. An entity object in a class diagram
b. A control class in a class diagram
c. A boundary object in an interaction diagram
d. A boundary class in a class diagram
e. (c) and (d) are both correct answers
f. None of the above

Q.4 In a UML state transition diagram, what does a black circle mean:

a. An end state
b. An initial state
c. A dead state
d. None of the above

Use the following diagram and answer the short answer questions 4, 5, 6, and 7:



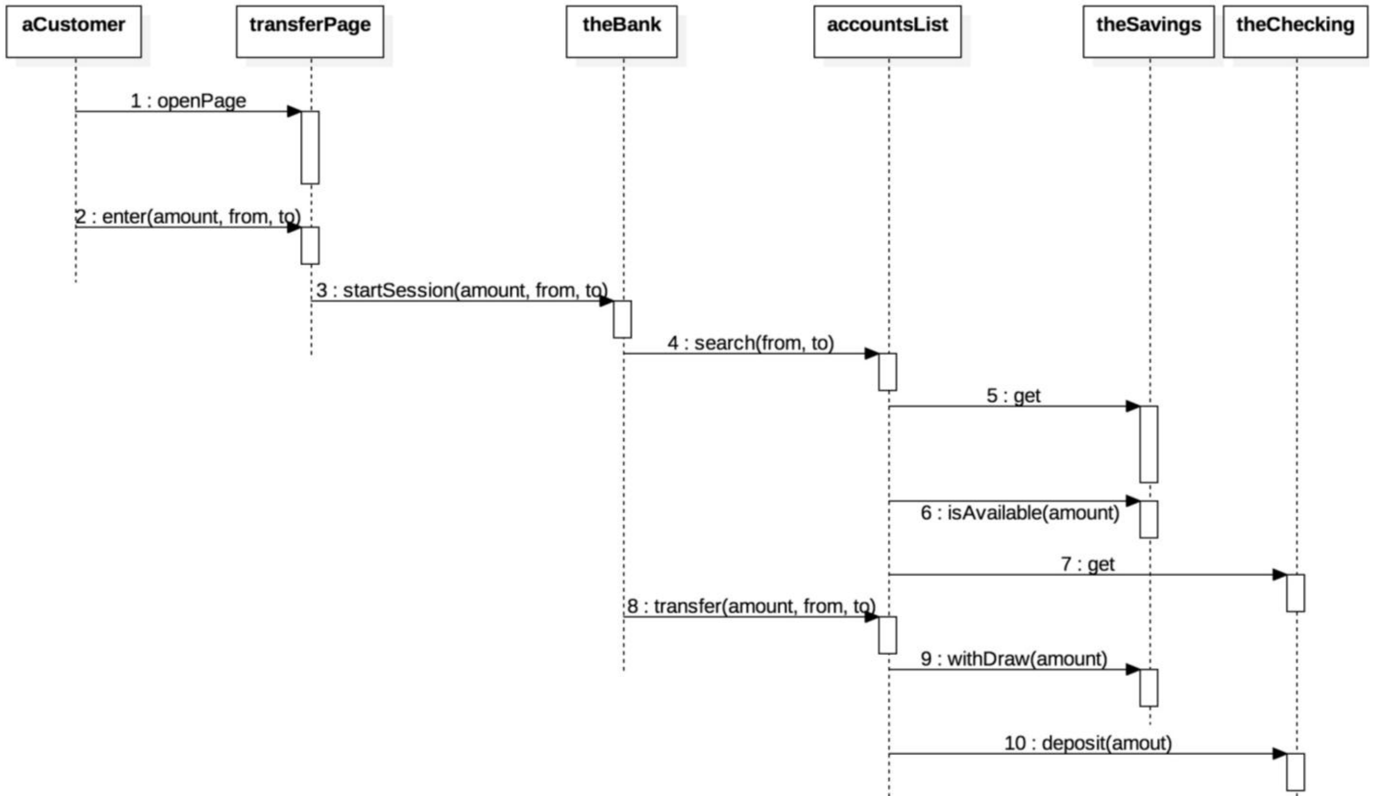
Q.4 What is the name of this diagram?

Q.5 What the wide boxes at the top of the vertical lines in this diagram represent:

Q.6 What the dashed vertical lines are called:

Q.7 What the narrow boxes at the dashed lines in this diagram represent:

Study the following UML digarm carefully and answer the following question:



Q.8 Based on this diagram, draw a design-class digram, showing the classes, their relationships (inharitance, aggregation, association), cardinalities mong them, and ony their major functions this shown in the above diagram. **Don't worry about the class attributes.**

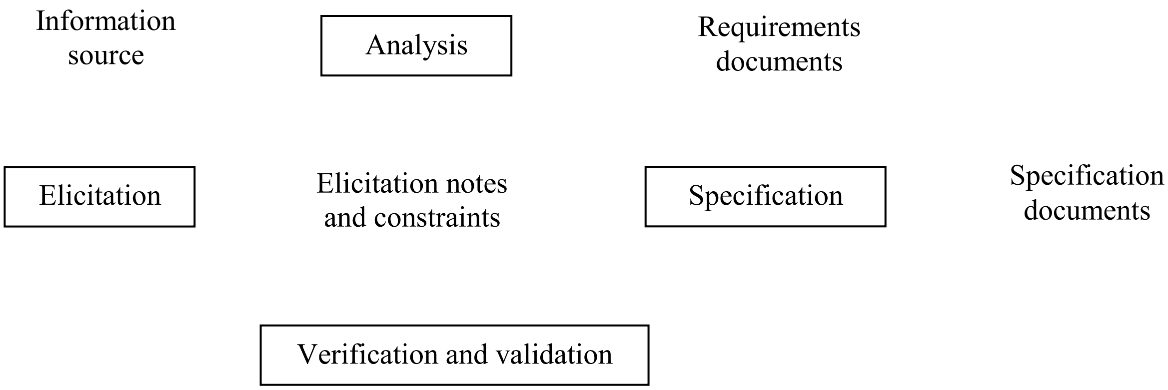
Consider the following problem statement and answer the following question:

In a Geometric Software System, we need to use classes such as Circles, Rectangles, and Triangles; where all of them are a kind-of Shape. Each Shape object needs to have an object of class Point (representing a point on a Cartesian Plane with x and y coordinates) that is used as a reference coordinate for displaying these shapes the Cartesian plane on the screen. For example the reference point for a Circle is its center, the reference point for a Rectangle is its left-top-corner, and the reference for a Triangle is its top corner. The only functionalities that we are expecting from each of these classes are the calculation of the area and the perimeter.

Q.9 In the following area draw a class diagram. In this diagram you should show the classes, their relationship (inheritance, aggregation, association), multiplicities, attributes and functions.

TOPIC: Software Requirements Engineering (SENG471)

Q.1 Indicate the objective of software requirements engineering. Use arrows to connect information below to illustrate the process of software requirements engineering.



Q.2 The following example shows the requirements as initially specified. Indicate your critique of the requirements against the desirable characteristics of a software requirements specification; and present your resulting re-specification. [NOTE: The example depicts only the requirements effort.]

Example – Initial specification:

Software will not be loaded from unknown sources onto the system without first having the software tested and approved.

Critique:

Re-specification:

Q.3 Given the following segment of software requirements for driving an autonomous car (a car does not need a human driver), explain what is incorrect found by verification and what are missing found by validation.

Domain properties D:

- Wheel pulses are on if and only if the wheels of the car are turning.
- Wheels are turning if and only if the car is moving on a road.

Requirement R:

- Braking pulses shall be enabled when the camera of the car senses an obstacle in the moving direction of the car.

Specification S:

- Braking pulses shall be disabled if and only if wheel pulses are on.

Q.4 What is the objective of goal models? Draw a goal model to represent the following information elicited from a student about his/her goal of studying.

The student would like to get a full-time job to earn an income. The job hinders him/her attending lectures and studying hard. Both attending lectures and studying hard help get a good grade however.

TOPIC: Software Process and Project Management (SENG511)

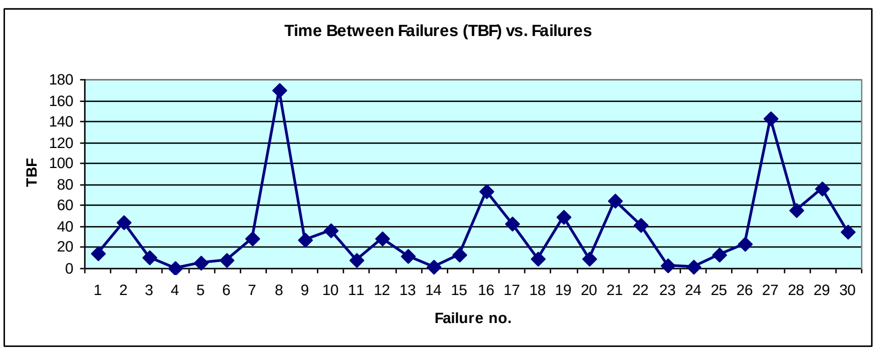
Q.1 Related to the statement “(Software Process and Project Management) Models are always wrong, and sometime useful”

- Explain the meaning
- Illustrate your points by one positive and a negative example
- What can be done to increase the usefulness of models?

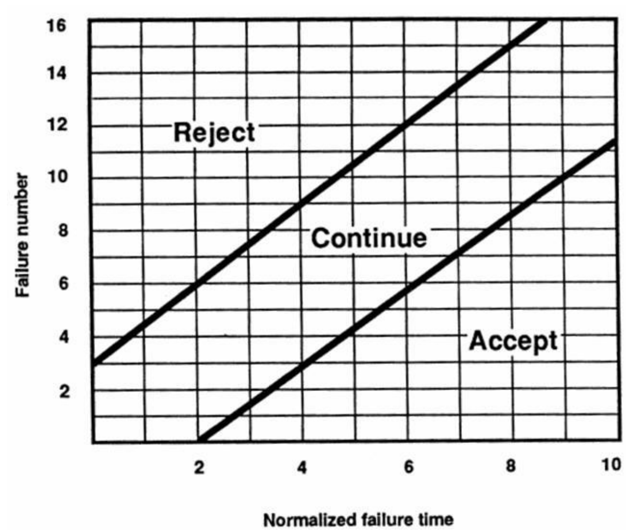
TOPIC: Software Reliability and Testing (SENG521)

Q.1 In order to monitor reliability of a workstation running Windows XP Professional operating system the system log was analyzed. During the testing period 30 errors were recorded in the *eventlog* file as shown below. The MTBF graph is also depicted below.

No.	Type	Time-between-failures (hours)	Error Source
1	Error	14.75	DCOM
2	Error	43.99	Service Control Manager
3	Error	9.89	DCOM
4	Error	0.07	DCOM
5	Error	5.70	DCOM
6	Error	7.89	DCOM
7	Error	28.79	DCOM
8	Error	170.15	DCOM
9	Error	26.83	DCOM
10	Error	36.15	System Error
11	Error	7.99	DCOM
12	Error	28.09	DCOM
13	Error	11.80	DHCP
14	Error	1.78	DHCP
15	Error	12.50	DCOM
16	Error	73.08	DCOM
17	Error	42.60	Service Control Manager
18	Error	9.18	DCOM
19	Error	49.43	DCOM
20	Error	9.19	DCOM
21	Error	64.25	System Error
22	Error	40.90	System Error
23	Error	3.07	Service Control Manager
24	Error	0.75	Service Control Manager
25	Error	13.36	System Error
26	Error	23.02	Service Control Manager
27	Error	143.31	Service Control Manager
28	Error	55.46	System Error
29	Error	75.57	DCOM
30	Error	34.31	DHCP



a) Determine the minimum value of failure intensity objective (λ_F) for which the behaviour of the system will be considered as acceptable. Assuming that MTTF is approximately equal to MTBF, failure intensity changes exponentially with time and Discrimination ratio (γ), Consumer risk (β) and Supplier risk (α) are 2, 10% and 10%, respectively.



b) The system must be rebooted after each "System Error" type error. If the recovery time to reboot the system after each "System Error" is 15 minutes, calculate the availability of this system.

Relationships, Formula and Equations

	Musa's Basic Model	***Logarithmic Poisson Model***
Failure intensity (λ) Mean number of failures (μ) Execution time (τ) Parameters: Initial Failure intensity (λ_0) Total failures (ν_0) Decay parameter (θ)	$\lambda(\mu) = \lambda_0\left(1 - \frac{\mu}{\nu_0}\right)$ $\lambda(\tau) = \lambda_0 e^{-\left(\frac{\lambda_0}{\nu_0}\right)\tau}$ $\mu(\tau) = \nu_0\left(1 - e^{-\left(\frac{\lambda_0}{\nu_0}\right)\tau}\right)$	$\lambda(\mu) = \lambda_0 e^{-\theta\mu}$ $\lambda(\tau) = \frac{\lambda_0}{\lambda_0\theta\tau + 1}$ $\mu(\tau) = \left(\frac{1}{\theta}\right)\ln(\lambda_0\theta\tau - 1)$
System availability (A) Downtime per failure (t_m) Mean-time-to-failure (MTTF) Mean-time-to-repair (MTTR)	$\lambda = \frac{1-A}{A\,t_m}$ $A = \frac{uptime}{uptime + downtime}$	$A = \frac{MTTF}{MTTF + MTTR}$
Time required for testing the system to reach the target failure intensity (Δτ) Present failure intensity (λ_p) Target failure intensity (λ_F)	$\Delta\tau = \frac{\nu_0}{\lambda_0}\ln\frac{\lambda_P}{\lambda_F}$ $\Delta\mu = \frac{\nu_0}{\lambda_0}(\lambda_P - \lambda_F)$	$\Delta\tau = \frac{1}{\theta}\left(\frac{1}{\lambda_F} - \frac{1}{\lambda_P}\right)$ $\Delta\mu = \frac{1}{\theta}\ln\frac{\lambda_P}{\lambda_F}$

Exponential Model		***Binomial Distribution***
$f(t) = \lambda e^{-\lambda t}$ $F(t) = 1 - e^{-\lambda t}$ $R(t) = 1 - F(t) = e^{-\lambda t}$ $MTTF = \frac{1}{\lambda}$	*f(t)* Probability Density Function (PDF) *F(t)* Cumulative Density Function (CDF) *R(t)* Reliability function MTTF Mean-time-to-failure λ failure intensity	Probability of exactly *x* successes: $f(x) = \frac{n!}{x!\,(n-x)!}\,p^x\,q^{n-x} = \binom{n}{x}p^x\,q^{n-x}$ Probability of having up to *r* successes: $F(r) = \sum_{x=0}^{r}\binom{n}{x}p^x\,q^{n-x}$
R_{system}: System reliability *n*: number of connected components F_{system}: System unreliability	Serial: $R_{system} = \prod_{i=1}^{n} R_i$ Parallel: $F_{system} = \prod_{i=1}^{n} F_i$ $R_{system} = 1 - F_{system}$ $R_{system} = 1 - \prod_{i=1}^{n}(1 - R_i)$	*n* : number of trials *f(x)* : probability of *x* successes in *n* trials *p* : probability of success *q* : probability of failure *p + q =1*
m-out-of-n Redundancy	$R_{system} = 1 - \sum_{i=0}^{m-1}\binom{n}{i}R^i(1-R)^{n-i}$	$\binom{n}{i} = \frac{n!}{i!\,(n-i)!}$

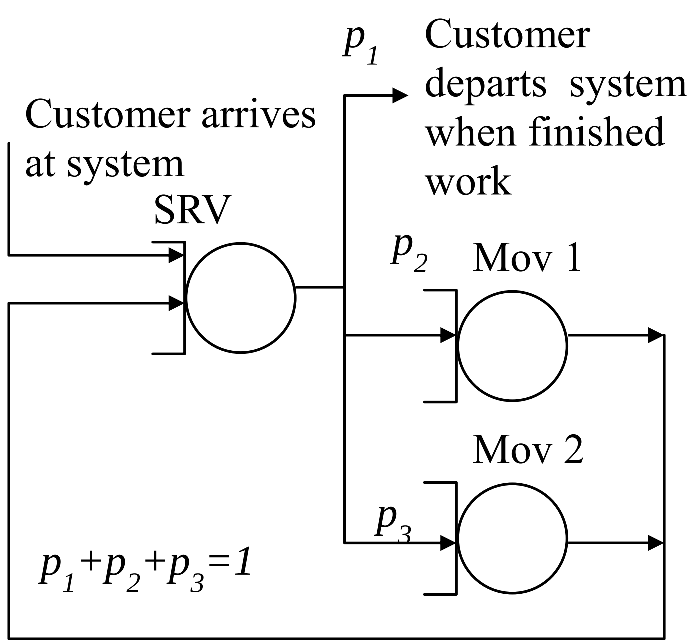
TOPIC: Formal methods (SENG523)

Q.1

a) What are formal methods and what are the roles of formal methods in software engineering in supplement to empirical technologies?
b) Why are formal methods demanded in software engineering particularly in safety-critical, large-scale, and complex projects?
c) In which processes of software engineering where formal methods are necessarily required?

TOPIC: Software Performance Evaluation (SENG533)

Q.1 Consider a system that serves short movie clips modeled as shown below. The system contains a media server resource SRV interacting with two movie file repository resources MOV1, and MOV2. A customer's movie clip download transaction requires 0.5 seconds of service (over all visits) from the SRV resource. A typical customer transaction requires 20 visits to SRV. After completing a visit to SRV, a customer can leave the system or visit either MOV1 or MOV2 with equal probability. After visiting MOV1 or MOV2, the customer returns to SRV as shown in the figure. Measurements revealed that MOV1 takes 50 milliseconds to satisfy a visit while MOV2 takes 30 milliseconds to satisfy a visit. Measurements also showed that MOV1 was servicing 15 visits per second.



a) Calculate the following:

i) The number of visits per customer transaction to MOV1 and MOV2
ii) The number of visits served by SRV and MOV2 per second
iii) The number of customers serviced by the system per second
iv) The utilizations of SRV, MOV1, and MOV2

b) What kind of an upgrade would you recommend to improve the performance of the system?

c) Further measurements showed that the average queue length, i.e., average number of requests, at SRV, MOV1, and MOV2 were 8.5, 3.5, and 1.5, respectively. Calculate the following performance metrics:

i) The per-visit average response time at SRV, MOV1, and MOV2

ii) The average response time encountered by a customer at SRV, MOV1, and MOV2

iii) Let's change the above system to a closed system where a customer "thinks" and re-visits SRV instead of leaving the system. What is the average number of customers in the system assuming an average customer think time of 2 seconds?

Q.2 Consider the system of Question 1-section c – sub-section iii. Plot the asymptotic optimistic throughput and response time bounds for this system. Clearly annotate the various regions of these plots with relevant information.

Q.3 A system consists of 50 terminals, a CPU, and two disks. The system was observed for 10,000 seconds and the following data was collected. The average response time for an interactive job was 5 seconds. On an average, each job visited the CPU 10 times between think times, and there were 200,000 service completions at the CPU. What was the average total number of jobs at the CPU and the disks?

Q.4. Consider a database server with eight processors and two disk subsystems. The eight processors allow the database server to run concurrent server processes. The use of a benchmark provided the following speedup factor function over a one-processor configuration.

$$\alpha_{processor} = \begin{cases} 0.56 + 0.44 * j & \text{for } j <= 8 \\ 4.08 & \text{for } j > 8 \end{cases}$$

The system is used for processor intensive transactions, explaining the need for an 8-processor server. The demands at the two disk subsystems are 0.008 and 0.011 seconds, respectively. The processor service time is 0.033 seconds and the processor visit count is 1. Compute the average server throughput, average server response time, and average processor utilization when the number of concurrent server processes is 3. Clearly state any assumptions you make.

Q.5 A Web server farm has traffic arriving over 8 input links. Requests arrive over each of the input links at a rate of *L* requests per second. The input links feed into a load balancer which routes requests across 8 servers using 8 output links. Requests arriving at the load balancer are routed to servers 1-7 with probability 0.1 each and to server 8 with probability 0.3. The time to process a request at a server is exponentially distributed with mean 4/(5*L). What is the average queue length of each of the 8 servers?

Q.6 Consider question 5 where the 8^{th} server was unstable and had an infinite queue length. The system architect modifies the behaviour of the 8^{th} server to mitigate this problem. This server now rejects a request if the number of requests waiting and getting serviced at the server is 4. Furthermore, the rejected request is routed with equal probability to one of the other 7 servers. Calculate the mean response time of each of the 8 servers under this configuration.

Q.7 Five independent experiments were conducted for determining the average response time of a Web server. One hundred observations were taken in each experiment, the means of which are reported below:

3.07, 3.24, 3.14, 3.11, 3.07

Based on this data, could we say that the true average response time **exceeds** 3.00 at a confidence level of 99.5%? (Marks: 5)

t-distribution table available at http://www.uwlax.edu/faculty/matchett/tables/tablt.pdf

Field of Study Examination, Feb 24 2017

Subject area: Software Engineering

This question paper has 8 pages (not including this cover page).

This question paper has 6 questions.

Answer a minimum of one question and at most three questions from this subject area.

Use a separate booklet (i.e., blue booklet) for the answers to questions in this subject area.

1. This question has four parts (a)-(d).

   (a) Provide brief answers to the following

      i. State true or false: The performance of RAM and CPU have been increasing at roughly the same rate.

      ii. State one reason for the recent shift from uni-processors to multi-core processors.

      iii. Gains in hardware bandwidth have been significantly outpacing gains in hardware latency. Describe one software-level performance optimization that can take advantage of this phenomenon.

      iv. State with a brief explanation one CPU overhead resulting from the use of RPC mechanisms.

      v. Explain briefly the terms natural bottleneck and shared bottleneck as they pertain to a multi-class system.

   (b) Which of the following expressions involves an assumption that cannot be verified operationally, i.e., by using measurements? Provide a brief justification

      i. $U = XD$

      ii. $X_k = XV_k$

      iii. $R_k = D_k(1 + Q_k)$

      iv. $N = XR$

   (c) Measurements of a Web server on a 16-core system serving a certain workload showed that each core was utilized 20% on an average. During the measurement period, the server was handling 10,000 HTTP requests per second. Calculate the average CPU demand for a request in this workload.

   (d) Consider two systems *Sys1* and *Sys2* where transactions arrive at the rate of λ transactions per second. *Sys1* has one resource and each transaction places a demand of D seconds on that resource. *Sys2* has K resources with each transaction placing demands of $D_1, D_2, ...D_K$ on the resources. The sum of the demands placed by a transaction of *Sys2* on all K system resources equals D. Which system will have a lower mean transaction response time? Justify your answer analytically.

2. This question has four parts (a)-(d).

   (a) In a typical dependency relationship between two classes

      i. The constructor of one class calls the constructor of the other class
      ii. Method of one class uses the object (s) of the other class in its input or its local variable
      iii. One class is a subclass of the other class
      iv. None of the above

   (b) How does polymorphism promote extensibility?

   (c) What is multiple inheritance? In the context of multiple inheritance, what is the dimond problem, and how can it be avoided?

   (d) You are asked to design an Analysis Class Diagram for the National Hockey League (NHL) based on the following problem statement: "NHL has 30 hockey teams. Each hockey team has several members which include: players, coaches and support staff. There is a schedule for all teams. The schedule consists of several games. Since each game is played between two teams, each game has the information of the two teams that are scheduled to play. Also, each game has a time and a date. A user can use this system to get information about the teams and their members, and also about the scheduled games."

   Draw a class diagram. In this diagram, you should show the classes, their relationships (inheritance, aggregation, association, etc.), and multiplicities. You DO NOT need to show any attributes or operations in your class diagram.

3. The following examples show requirements as initially documented by someone. Indicate your critique of the requirement against the desirable characteristics of a software requirements specification; and present your re-specification. NOTE: These examples depict only the inspection effort.

   Example 1 Initial specification

   3.4.6.3 The system shall prevent the processing of duplicate electronic files by checking a new record. An e-mail message shall be sent.

   Critique

   Re-specification

   3.4.6.3 The system shall

   Example 2 Initial specification

   3.2.7.1 The system shall purge records and files that are older than the operator or technical user-specified retention period.

   Critique

   Re-specification

   3.2.7.1 The system shall

4. This question has three parts (a)-(c).

   Related to the statement “(Software Process and Project Management) Models are always wrong, and sometimes useful”

   (a) Explain the meaning. What are the purposes of models in software engineering?
   (b) Illustrate your points by two positive (“appropriate”) and one negative (“inappropriate”) usage of models in software engineering.
   (c) What can be done to increase the usefulness of models?

5. This question has three parts (a)-(b). Part (a) has six subparts (i)-(vi) and part (b) has two subparts (i)-(ii).

(a) Answer the following questions

i. Probability of failure-free operation for a specified time in a specified environment for a given purpose is called ____________
(A) safety
(B) dependability
(C) reliability
(D) failure intensity

ii. Which one of the followings cannot usually be calculated by a reliability growth model?
(A) Prediction of probability of failure of a component or a system
(B) Estimation the mean time to the next failure
(C) Decision of system being accepted or rejected
(D) Estimation of failure per kilo-line of code

iii. A ____________ is a set of failures that have the same per-failure impact on users using a failure classification criteria
(A) failure intensity
(B) failure severity class
(C) failure mode

iv. In a serially connected system, its reliability is always ____________ than individual reliability of its components
(A) smaller
(B) larger
(C) same

v. In a parallel (redundantly) connected system, its reliability is always ____________ than individual reliability of its components
(A) smaller
(B) larger
(C) same

vi. ____________ is a graphical representation of how the components of a system are connected from reliability point of view.
(A) Fault tree
(B) Reliability Block Diagram (RBD)
(C) Failure Mode and Effect Analysis (FMEA)

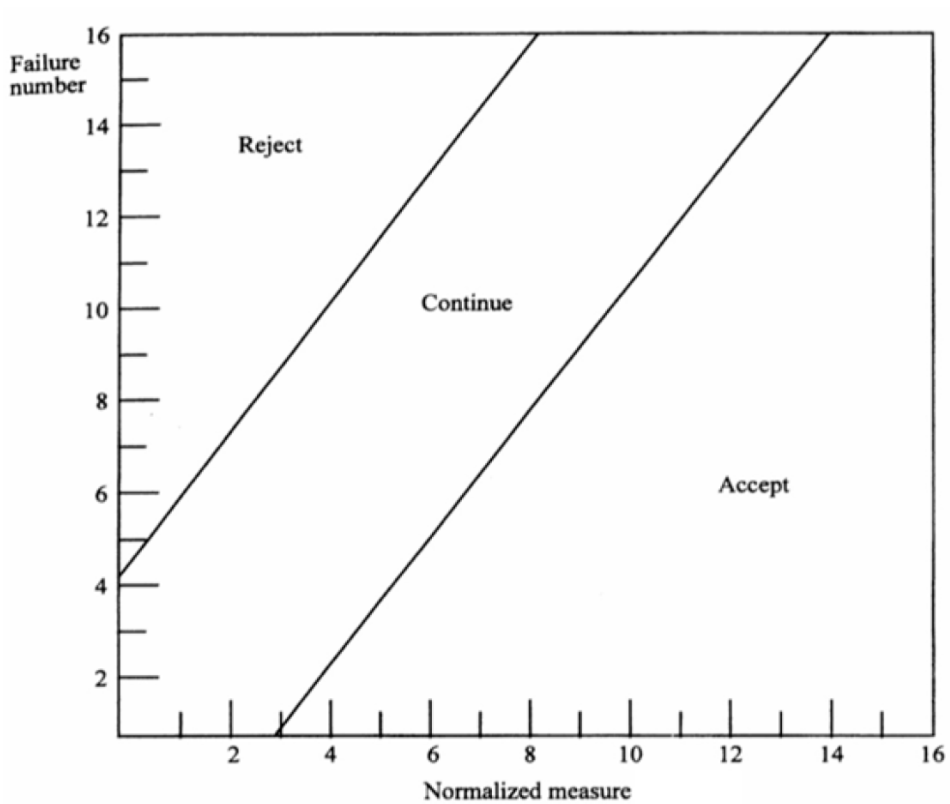
(Question 5 is continued on next page)

(b) Assume you run a certification test in order to decide whether to accept or reject a recently developed software system. Your target failure rate is 1 failure per 20 hours of operation. Assume you detect a failure after 8, 16, 18, 28, 30, 32, 38 and 40 test hours, respectively.

i. Complete the table below by calculating "normalized" failure numbers.

Failure number	Test hours	Normalized failures
1	8	
2	16	
3	18	
4	28	
5	30	
6	32	
7	38	
8	40	

ii. Use the Reliability Demonstration chart below the table to take a decision. Which decision do you take (circle one)?
(A) Accept
(B) Reject



6. This question has seven parts (a)-(g). Part (g) has two subparts (i)-(ii)

(a) Formal methods for software engineering are characterized by the following statements *except*:

i. A mathematics-based notation system

ii. A specification language with rigorous semantics and rules

iii. A documentation of the design for a software system

iv. A refinement scheme supporting top-down specifications

(b) Which of the following expressions is not a general mathematical model of software S, the top-level system dispatching mechanisms, or an embedded relation of software processes?

i. $S = PM \times SM$

ii. $S = E \times SM$

iii. $S = E \times PM \times SM$

iv. $P = \mathop{R}\limits_{i=1}^{n-1} (p_i \; r_{ij} \; p_j), j = i + 1$

(c) What is/are the purpose(s) for adopting timers in real-time system specifications and modeling?

i. For monitoring a pending event

ii. For freeing the processor (CPU) resource to more urgent tasks

iii. For adapting the processor to slow human responses

iv. All the above

(d) What is the unit of software quality Q according to the formal model $Q = \int_0^T W(t)dt$ where $W(t)$ is work done by (or dynamic performance of) the software system within its lifecycle T?

i. Number of bugs

ii. Function-hour (FHr)

iii. Mean-time between failures (MTTF)

iv. Rate of testing coverage

(e) The *error rate of a work product* can be reduced upto n folds of the average error rate of individuals r_e in a group via n-nary *peer reviews* based on the random nature of error distributions and independent nature of error patterns of different individuals, i.e. $R_e = \prod_{k=1}^{n} r_e(k)$. Given the average error rate at all levels of a software quality insurance system as $r_e(k) = 10\%$, how many levels (n) of quality insurance will be required in order to reach a reduced error rate at $R_e = 0.00001$ for the software project?

i. 4

ii. 5

iii. 6

iv. 7

(Question 6 is continued on next page)

(f) The big-R notation of RTPA may be used to denote the following list of system specifications *except*:

   i. Recurring structural entities
   ii. Repetitive software behaviors
   iii. System event captures
   iv. Iterations and recursions

(g) A typical event scan algorithm is implemented by two cyclic scans and a comparative analysis. For instance, a line scanner in the telephone switching system (TSS) adopts the following event capturing logic as shown in the table.

No.	LineScanner(i|N)|SM. **CurrentScan|BL**	LineScanner(i|N)|SM. **LastScan|BL**	LineScanner(i|N)|SM. **Status|N**
0	F	F	0 – Idle
1	T	F	1 – Hook-off ☏
2	F	T	2 – Hook-on ☎
3	T	T	3 – Busy

   i. Specify a simple structure model (SM) for modeling the structure of LineScanner|SM:

   ii. Specify a simple process model (PM), LineScan|PM, for modeling the behavior of event detection based on LineScanner|SM: