

SAMPLE QUESTIONS FOR PhD QUALIFYING EXAMINATION

CS 6352-Performance Computer Networks

1. Into a single queue with unlimited buffer size, customers **always** arrive in **pairs** but line up one after another in the waiting line for service. One of an arriving pair joins the line ahead of the other, by a random choice. The single server takes up one customer at a time from the head of the queue and serves. Each customer leaves immediately upon completion of his/her service (that is, without waiting for the completion of the partner's service). Arrivals of **pairs** are Poisson with a rate of 1 **pair** per hour. Service times of **individual customers** are independent and identically distributed Exponential random variables with a mean of 0.25 hour for each customer (and not for a pair). Determine the expected response time of an **individual customer**.

2. Consider a two-processor server where each processor is enabled with technology that allows the processors to operate at a higher speed under heavy loads, and to operate at a reduced speed under low loads to conserve power. Users submit jobs according to a Poisson process with rate λ jobs per second. If there is one job in the system, only a single processor is busy, and the processor operates at a speed of 500 MHz (500 million cycles per second). If there are two jobs in the system, both processors are busy, and each operates at a speed of 1000 MHz. If there are three or more jobs in the system, both processors are busy, and each processor operates at a speed of 1500 MHz. The length of a job is modeled as exponentially distributed and requires an average of 500 million CPU cycles. Buffer size is unlimited.

- (a) Draw the state diagram for the system, clearly labeling transition rates.
- (b) Find the steady-state probabilities for the number of jobs in the system.
- (c) Find the condition for λ in order for the system to be stable.

CS 6353-Compiler Construction

Syntax Analysis

Explain why the following grammar is LALR(1) but not SLR(1).

$$S \rightarrow albAc|dclbda$$
$$A \rightarrow d$$

Intermediate Code Generation

Give a syntax-directed translation scheme for the while-loop, i.e.

$$\text{Rule1} \rightarrow \text{while (E1) do S1; \{ define your action here \}}$$

The while-loop has the common meaning and you can use either two pass approach or back-patching. Synthesized attributes can be defined as necessary. In addition, there are two predefined functions as follows:

- newlabel(): return an unused new label L.
- gen(X): generate three address code as indicated by X.

CS 6354-Software Engineering

System Description

Consider a system for maintaining patient records in a doctor's office. The system should allow the user to create records for new patients, update the information, and delete patient records. The system should also generate billing information for insurance companies, record payments from the companies and patients, generate bills for patients, and send reminders to patients regarding follow-up treatments.

For each patient, the system stores the name, social security number, date of birth, address, and telephone number of the patient as well as the patient's insurance company, employer, and name of parent or guardian (if under 18 years), name of spouse (if married). The system also maintains the medical history of each patient, including the date of each visit, the diagnosis, prescribed treatment, duration of treatment, and the result.

The system must enable the user to perform the following functions:

1. For adding a new patient to the system, the user fills out a form containing the patient information and submits it to the system. Various validation checks are performed by the system (duplicate name, reasonable age, correct insurance company, etc.) and the record is accepted only if it passes these checks. The user can also delete patient records.
2. For updating the record of a patient, the user fills out or updates the information (including date of visit, diagnosis, total charge, patient and insurance payments, change of address, etc.). The system accepts the updates if the validation checks are passed.
3. If the user wishes to generate bills for insurance companies, the system goes through all the patient records and generates bills for new payments or payments that are past the due date. Letters and addresses are prepared and printed.
4. Similarly, the user can generate bills for patients as well as send reminders for patients who need follow-up treatment.

1. Requirements elicitation and analysis.

- (i) Draw a Use Case diagram showing the functions of the system. Show the system boundary, use case(s), and actors.

2. Requirements analysis.

- (i) Consider the following entity classes:

PatientRecordSystem (keeps track of the information about all the patients in the system); CompanyInfo (name, address of the company and the name and telephone number of a point of contact); Date (month, day, year); Status (information about the status of the patient, including starting treatment, continuing treatment, recovered, etc.); History (keeps information for each visit by a patient); PatientInfo (name, address, telephone number, parent/guardian info, date of birth, social security number, marital status and spouse's info, history, employer, insurance company); PersonInfo (name, address, telephone number), VisitInfo (date, problem description, diagnosis, prescribed treatment, duration - start & end dates, status, payment info); PaymentInfo (responsible party, amount billed, amount paid);

Draw the Class diagram. Use all the classes listed above and do not use any classes not listed above. Show all the associations, inheritances, aggregations, multiplicities, and roles.

3.

Consider an issue that arises in the implementation of the database for the patient information system, namely, how to implement the database subsystem. Give details of the following rationale management scenario that might arise (give concise answers in English, not UML, with just a few sentences in the space provided here).

- (a) State the problem.
- (b) Identify some proposals
- (c) Give some evaluation criterion (criteria).
- (d) Give some arguments.
- (e) Describe the resolution.
- (f) List possible action items

CS6360 - Database Systems

Problem 1

Consider the following relations:

Student (*snum: integer*, sname:string, major:string, level:string, age:integer)

Class (*name:string*, meets_at:time, room:string, fid:integer)

Enrolled (*snum:integer, cname: string*)

Faculty (*fid:integer*, fname:string, deptid:integer)

The meaning of these relations is straightforward; for example, Enrolled has one record per student-class pair such that the student is enrolled in the class.

Write the following queries in *SQL*.

- a. Find the age of the oldest student who is either a History major or enrolled in a course taught by I. Teach.
- b. Find the names of all classes that either meet in room R128 or have five or more students enrolled.
- c. Find the names of all students who are enrolled in two classes that meet at the same time.
- d. Find the names of faculty members who teach in every room in which some class is taught.
- e. Find the names of students who are enrolled in the maximum number of classes.
- f. For each age value that appears in Students, find the level value that appears most often. For example, if there are more FR level students aged 18 than SR, JR, or SO students aged 18, you should print the pair (18, FR).

CS 6361-Requirements Engineering

1. Functional– Structural/Behavioral & Non-Functional

a. RML (Requirements Modelling Language) pioneered in World Modeling, and defined a formal semantics for the “Generalization” relationship. Using an example, concisely and precisely explain what it means to define the generalization relationship as a *partial order* relation.

b. Consider the following definitions:

- *IN* (*i*, *C*, *t*): predicate asserting that token *i* is an instance of class *C* at time *t*;

- *IS-A* (*C1*, *C2*): time-independent predicate asserting that class *C1* is a subclass of class *C2*;

- *PROPDEF* (*C*, *a*): time-independent function which gives the class to which the value of attribute *a* for instances of *C* must belong.

Using an example, precisely describe what the following axiom means:

$[\text{PROPDEF}(C, a) = E \text{ AND IS-A}(D, C)] \rightarrow \text{exists } F [\text{PROPDEF}(D, a) = F \text{ AND IS-A}(F, E)]$

c. Give a simple example of a state transition using (Augmented) Petri Net formalism, and then describe what it would mean to apply the three epistemological primitives (i.e., structural dimensions) of object-orientation to the state(s) in your example.

d. (5) Using a (softgoal) dependency graph, represent the following process as precisely as possible:

The developer states the non-functional requirement “*Accounts should be secure*”, and refines it into three sub-goals, for Integrity (guarding against unauthorized update or tampering), Confidentiality (guarding against unauthorized disclosure), and Availability (guarding against interruption of service) of the account. The developer further refines the Integrity goal into two sub-goals, for Accuracy and Completeness of the account.

Now focusing on the Confidentiality Requirement in moving towards a secure target design, the developer considers a list confidentiality assurance techniques and decides to use an Authorization Technique. Repeating this process, the Authorization Technique is further refined to goals for Identification, Authentication, and Access Rule Validation. At this moment, the developer notices that Access Rule Validation has a positive impact on the accuracy of accounts, as ill-intentioned users can be denied access and prevented from committing forgery. The developer now further refines Authentication into a goal for “Require additional ID”, but feels that this goal would have a negative impact on User-friendly access. Although User-Friendly access hasn't yet been posted, it turns out that it is fairly important and should have been posted in the very beginning. So, the developer now posts it as a goal, although an Authorization may be seen negatively towards User-Friendly access if the Authorization involved something personal.

CS 6362-Software Architecture and Design

1. Consider the following declarations:

module M1
 provides: w, n

requires: m
consist-of: module M12, function F11

```
module M12
  provides: n;
  requires: m;
  has-access-to: module M2
  int n, Boolean m
end M12
```

```
function F11
  provides: w;
  requires: n;
  boolean w, int n
end F11
```

```
module M2
  provides: m,q;
  int m, char q
end M2
```

1.1 List the set of variables that M12 can access

1.2 The specification above has one inconsistency. What is the inconsistency?

2. Suppose you are to build a system to help buyers and sellers, in a real estate market, who are interested only in the price and location of the real estate.

2.1 Depict a diagram of a software architecture in an implicit invocation style with control for the system.

2.2 Depict graphically how the architecture in 4.1 can be implemented using the Java Event Model.

CS 6363-Algorithms

1. Describe a polynomial time reduction from CLIQUE to VERTEX COVER. Prove that the reduction is correct, *i.e.* prove that $x \in \text{CLIQUE}$ if and only if $f(x) \in \text{VERTEX COVER}$, where f is the function you define for the reduction.

2. Give a simple example of a directed graph with negative weight edges for which Dijkstra's algorithm produces incorrect answers. Explain.

3. Does the standard Ford-Fulkerson network algorithm for computing maximum flow from a given source to a given sink, where the edges have capacity constraints, allow an arbitrary selection for an augmenting path from the source to the sink in each successive residual network and still manage to guarantee eventual termination with maximum flow? Explain why or why not.

CS 6364-Artificial Intelligence

1. The knowledge base KB has one Boolean formula:

$$(x_1 \equiv x_2) \rightarrow (x_2 \vee x_3)$$

Find an equivalent knowledge base in Conjunctive Normal Form (CNF).

2. Consider a student that passes the PhD qualifiers test on AI, and the PhD qualifiers test on ALGORITHMS. Use Naïve Bayesian reasoning to decide if the student would produce a good dissertation based on the following information:

1. 10% of all students can produce a good dissertation.
2. Among students that produce good dissertations 90% can pass the AI qualifiers, and 80% can pass the ALGORITHMS qualifiers.
3. 50% of the students that cannot produce a good dissertation are able to pass the AI qualifiers test, and 30% are able to pass the ALGORITHMS qualifiers test.

Explain your conclusion.

CS 6367-Software Testing

1. Given a complicated software system to be tested, assume you have all the source code as well as the final executable system. Assume also you have some documentation even though it might be incomplete and/or out-of-date.

Assume that testing techniques such as equivalence partitioning, boundary-value analysis, control flow-based testing (e.g., branch testing), dataflow-based testing (e.g. “all-uses”), fault-based testing are available. Make a suggestion on how to conduct unit testing, integration testing, and system testing so that the selected testing techniques complement each other to provide a **cost-effective** approach for detecting as many faults as possible.

Justify your answer and explain all the assumptions you make, if any.

2. (a) Given the pseudo-code below:

```
integer x,y;
read x;
read y;
x = abs(x) + 2;
z = 3;
if (even(x/y)){
    for( int i = 0 ; i < (x+y) ; i++){
        if ((i mod x) == 0)
            z = z * x ;
        else
            z = z * y;
    }
    print z
}
print x,y
```

- 1) draw a control flow graph showing all the def, c-use, p-use actions.
- 2) construct the set of items to be covered for “all-uses” (i.e.,

the (def,c-use), (def, p-use) pairs).

- 3) Ensuring that each test increases the coverage, create test cases to satisfy the "all-uses" strategy (explicitly state each def-p-use, def-c-use pair covered by each test case).

- (b) An exception is a signal that indicates that some sort of exceptional condition (e.g., a run-time error) has occurred. When an exception occurs, an "exception-handler" is invoked to handle the exception. Explain if, how the possibility of exceptions would affect the way you carry-out "all-uses".

CS 6371-Software and Design Programming Language

1. It is well known that tail recursive programs are efficient because they can be converted automatically into a program that contains only iteration. Consider a class of programs called "almost tail recursive" which have tail recursion as the second last operation. The last operation is usually a very simple operation involving an addition or multiplication. For example, factorial is an almost tail recursive program.

fac(0) => 1

fac(n) => n*fac(n-1)

How can you convert an "almost tail recursive program" into a tail recursive program? You can demonstrate your technique by showing how it transforms the factorial program into a tail recursive factorial.

CS 6375-Machine Learning

Naïve Bayes Learning

Table below indicates training example for the target concept *Play Tennis*.

Day	Outlook	Temperature	Humidity	Wind	PlayTennis
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

Use Naïve Bayes Learning Algorithm to decide how the new instance should be classified. Show all the computation steps.

Outlook Temp Humidity Wind
< Sunny Cool Normal Strong >

CS 6378-Advanced Operating Systems

Question 1:

- (a) The Schiper-Eggli-Sandoz protocol for causally ordered message delivery states that the communication channels can be non-FIFO. However, the description does not state anything about the reliability of channels. What will be the impact of message losses on the operation of the Schiper-Eggli-Sandoz protocol? Provide explanations to support your answer.
- (b) Assume a system of N nodes. If all communication is broadcast (all nodes are destinations of every message) is it still necessary for every message to carry $O(N^2)$ dependency information? Would it be possible to ensure causally ordered delivery while having each message carry less than $O(N^2)$ dependency information? If yes, how much dependency information should each message carry and why? Explain your answer.

Question 2:

- (a) The three-phase commit protocol is said to be non-blocking and resilient to single site failure. This means that in spite of the failure of a single site, all the operational sites agree on the outcome of the transaction by examining their local states. Is the three-phase commit protocol resilient to the failure of more than one site? If so, prove your answer. If not, provide a counter-example to support your answer.
- (b) The state transition and the resultant sending of message(s) in the three-phase commit protocol is atomic (either the message is sent to all the intended recipients or to none). Would the correctness of the protocol be affected by relaxing the atomicity requirement? Explain.

CS 6385-Algorithmic Aspects of Telecom Networks

1. A graph $G = (V, E)$ is 2-connected if it is connected and the removal of its any node does not disconnect the graph (i.e. the remaining graph is connected). Briefly describe a $O(|E|+|V|)$ -time algorithm for determining whether or not any input graph G is 2-connected.
2. We are given 3 queuing systems, each having a 56Kbps line of 50% utilization. Assume there is no limit on the buffer size and the messages are transmitted in the FIFO order. Messages arrive independently, and the average message length is 2000 bits. (a) The first has exponentially distributed message lengths. (b) The second has constant (2000-bit) message length. (c) In the third, half the messages are 800 bits long and half are 3200 bits long. Calculate the waiting time T_w for each of these 3 cases.
3. We would like to design a 2-connected network topology on a given set of nodes. Assume that a cost matrix is given that contains the cost of connecting any two nodes. Consider the following solution.
 - Step 1: Build a minimum cost spanning tree T_1 (with any known algorithm)
 - Step 2: Identify the leaf nodes of T_1 . Let L be this set of leaf nodes.
 - Step 3: Build a minimum cost spanning tree T_2 on the set L (with any known algorithm), ignoring the rest of the nodes. Let $G = T_1 \cup T_2$ be the created network topology, that is, the union of the two trees.

- (a) Show that the constructed graph G is 2-node-connected, that is, the removal of any single node cannot disconnect it.
- (b) Prove that this algorithm is not optimal, by providing a small example in which it does *not* find the least expensive 2-node-connected network topology.
- (c) Is it true that the cost of the solution that the above algorithm provides is at most twice the cost of a minimum cost spanning tree? Justify your answer!

4. Assume we want to build a network with a simple ring topology. We can use links of 150 Mbit/s speed. The traffic demand between each pair X, Y of nodes is 1 Mbint/s, both from X to Y and from Y to X (the $X \rightarrow Y$ traffic is carried independently of the $Y \rightarrow X$ traffic). Under these conditions, how large ring can be built?

CS 6388-Software Project Planning & Management

1. Develop a W-level entity process for the design phase that includes the activities between the time the requirements specification has been completed, and the time the algorithms and data structures have been defined.
2. Suppose you are asked to perform the V&V project function during the Integration and Test of a 64 KDSI semi-detached software package for producing environmental impact reports. Should you agree to do the job alone, or should you ask for help? Explain.

CS 6390-Advanced Computer Networks

1. Network Calculus

1. Let A be the cumulative arrival function into a system, B be the cumulative departure function of the system. Give the definition of a service curve, i.e., if β is the service curve, express $\beta(t)$ as a function of $A(t)$ and $B(t)$.

2. Assume you have two systems S_1 and S_2 concatenated to each other, where β_1 is the service curve of S_1 and β_2 is the service curve of S_2 . What is the service curve of the entire system $S_1; S_2$? (i.e., the service curve of S_1 concentrated with S_2).

2. FCFS

What is the importance of the delay theorem for FCFS by Chlamtac, Farago and Fumagalli (the paper entitled “A Deterministic Approach to the End-to-End Analysis of Packet Flows in Connection-Oriented Networks”) as opposed to the original results of FCFS multiplexors by R. L. Cruz (paper title “A Calculus for Network Delay Part I and II”)? i.e., how does the former improve upon the results of the latter?

3. What is quality of service (QoS)?

Give an example of QoS definition in ATM network;

4. Describe and compare integrated service architecture (ISA) and differentiated service (Diffserv) model.

5a. Assume that a router in the path between two hosts communicating using TCP can sniff all the IP packets and send them to a distant host C reliably. If host A is downloading a file f from B, how can C get the entire contents of the file?

5b. If the sequence number field in TCP is to be increased to 64 bits and the data rate of the communication path between the two end hosts is 100 Mbps, how long does it take for the sequence number to wrap around?

6. Route optimization is mobile IP is used when we want to avoid the inefficient routing of packets from the source to the home agent and then tunneling the packets from the home agent to the destination host at the foreign network. Explain how this is done and what are the disadvantages of this scheme?