**Next:** About this document

**Department of Computer Science**
**Graduate Comprehensive Exam**
**Fall 1997**

- **Answer the questions on the paper supplied. Be sure to answer question 1, question 2, and question 3.**
- **Start each question on a new page.**
- **Do NOT write on the back of the paper.**
- **Write your SSN in the top right corner of each page of your answer. Do NOT put your name anywhere on the answers.**
- **Put the number of the question being answered in the top left corner of each page of your answer.**
- **If the answer to a question is written on more than one page, number the pages consecutively.**

1. CS5329 Algorithm Design and Analysis

   Design an algorithm to find a vertex in a connected undirected graph whose removal does not disconnect the graph. The algorithm should run in linear time. As a consequence, prove that every connected graph contains such a vertex.

2. CS5318 Design of Programming Languages

   **A.**

   Discuss the readability, writability, and costs of each of the following C language design features:

   **a.** inclusion of for, while, and do loops;

   **b.** inclusion of functions but not procedures;

   **c.** syntax of the if statement

   **d.** overloading of + as an operator on char, int, float and double;

   **e.** use of call-by-value as the single parameter passing mechanism.

   **B.**

   Write an algebraic specification for an abstract data type bstree (binary search tree) with the following operations:

   **a.**

create: -> bstree

**b.**
make: bstree x element x bstree -> bstree

**c.**
empty: bstree -> boolean

**d.**
left: bstree -> bstree

**e.**
right: bstree -> bstree

**f.**
data: bstree -> element

**g.**
isin: bstree x element -> boolean

**h.**
insert: bstree x element -> bstree

3. CS5391 Survey of Software Engineering

   There are five principal phases to the linear sequential or "waterfall" life cycle model. Please discuss each of these phases, addressing the activities that are performed during each phase. Also identify the inputs and outputs associated with each phase. Compare and contrast the linear sequential model with the incremental and spiral life cycle models.

---


- About this document ...

**Next:** About this document


*Carol Hazlewood*
*Mon Nov 23 14:29:06 CST 1998*

**Next:** About this document

**Department of Computer Science**
**Graduate Comprehensive Exam**
**Fall 1998**

- **Answer the questions on the paper supplied.**
- **Answer question 1. Answer question 2. Answer *one of* questions 3 *or* 4, depending on how you registered.**
- **Start each question on a new page.**
- **Write on only one side of the paper.**
- **Write your SSN in the top right corner of each page of your answer. Do NOT put your name anywhere on the answers.**
- **Put the number of the question being answered in the top left corner of each page of your answer.**
- **If the answer to a question is written on more than one page, number the pages consecutively.**

1. CS5346 Advanced Artificial Intelligence

    1. Show by examples how artificial intelligence techniques are different from conventional techniques.
    2. Using the A* algorithm, generate a search tree for the following start node of the 8-puzzle. Show that your heuristic function satisfies the A* condition.

    tabular25

2. CS5329 Algorithm Design and Analysis

    1. Sorting.
        1. Show the steps in sorting the array <5,3,1,2,4> using *two* of the following algorithms: Insertion Sort, Quicksort, or Counting Sort.
        2. Explain how Quicksort can be slower than both Insertion Sort and Counting Sort.
        3. What kind of input produces worst-case time complexity in Quicksort?
        4. State and solve a recurrence for the time complexity of Quicksort if there is always an even split.
    2. Design Techniques.
        1. Discuss how divide-and-conquer, dynamic programming, and greedy techniques work.

2. Describe characteristics of problems that make them amenable to solution by each technique.
3. Give examples of problems that can be solved by each technique and explain why that technique is advantageous.

3. 5310 Network and Communication Systems

   Describe the Fourier series. Explain in detail its significance with respect to data communications.

4. CS5332 Database Theory and Design

   Discuss the relationships between relational algebra and SQL. In each case, please list their essential features first. Give an example to demonstrate the similarities and differences between these two query languages.

---

- About this document ...

**Next:** About this document

*Carol Hazlewood*
*Mon Nov 23 14:39:58 CST 1998*

**Department of Computer Science**
**Graduate Comprehensive Exam**
**in**
**Computer Science**
**Fall 1999**

- **Answer the questions on the paper supplied.**
- **Answer all questions.**
- **Start each question on a new page.**
- **Write on only one side of the paper.**
- **Write your SSN in the top right corner of each page of your answer. Do NOT put your name anywhere on the answers.**
- **Put the number of the question being answered in the top left corner of each page of your answer.**
- **If the answer to a question is written on more than one page, number the pages consecutively.**

1. CS5329 Algorithm Design and Analysis
    1. Solve the following recurrences:
        1. eqnarray23
        2. eqnarray27
    2. Design Techniques.
       Discuss the divide-and-conquer and greedy techniques. Describe each technique, list characteristics of problems that make them amenable to solution by each technique, and show how the techniques are used in examples of problems that can be solved by each technique.
2. CS5338 Formal Languages
    1. Give a definition for a Turing machine.
    2. Construct a deterministic finite-state automaton that recognizes the regular language (ab + aba)*.
    3. Construct a pushdown automaton that accepts the palindrome language L ={wcw | w is in (a+b)*}. Describe in a short paragraph how the machine works.

3. CS5391 Survey of Software Engineering
   There are five principal phases to the linear sequential or "waterfall" life cycle model. Please discuss each of these phases, addressing the activities that are performed during each phase. Also identify the inputs and outputs associated with each phase. Compare and contrast the linear sequential model with the incremental and spiral life cycle models.

---

---

*Carol Hazlewood*
*Mon Nov 22 15:28:09 CST 1999*

# Department of Computer Science
# Graduate Comprehensive Exam
# in
# Computer Science
# Fall 2000

- **Answer the questions on the paper supplied.**
- **Answer question 1. Answer question 2. Answer *either* question 3 *or* question 4.**
- **Start each question on a new page.**
- **Write on only one side of the paper.**
- **Write your SSN in the top right corner of each page of your answer. Do NOT put your name anywhere on the answers.**
- **Put the number of the question being answered in the top left corner of each page of your answer.**
- **If the answer to a question is written on more than one page, number the pages consecutively.**

1. CS5346 Advanced Artificial Intelligence

   (a) Explain "admissible heuristic". Prove the optimality of A* algorithm. Also, explain the complexity of A* algorithm.

   (b) Explain Minmax Search Method including Alpha-Beta Pruning.

2. CS5391 Survey of Software Engineering

   There are five levels to the capability maturity model (CMM). Discuss each of these levels, including the key practices that are in place at each level and which practices are needed to advance to the next level.

3. CS5318 Design of Programming Languages

   (a) discuss in detail and with examples the role of abstraction in programming language design.

   (b) discuss in detail and with examples the use of context-free grammars to specify operator precedence and associativity

4. CS5338 Formal Languages

   (a) Give a precise definition of a Turing machine. Describe how a Turing machine operates.

   (b) Construct a minimal finite-state automaton to recognize the language $L = \{a^*b^*a^*\}$. Discuss why $L$ can be recognized by a finite-state automaton but the language $K = \{a^nb^na^n \mid n \geq 0\}$ cannot.

# Department of Computer Science
# Graduate Comprehensive Exam
# in
# Computer Science
# Fall 2001

- **Answer the questions on the paper supplied.**
- **Answer question 1. Answer question 2. Answer *either* question 3 *or* question 4.**
- **Start each question on a new page.**
- **Write on only one side of the paper.**
- **Write your SSN in the top right corner of each page of your answer. Do NOT put your name anywhere on the answers.**
- **Put the number of the question being answered in the top left corner of each page of your answer.**
- **If the answer to a question is written on more than one page, number the pages consecutively.**

1. CS5391 Survey of Software Engineering

   There are five levels to the capability maturity model (CMM). Discuss each of these levels, including the key practices that are in place at each level and which practices are needed to advance to the next level.

2. CS5329 Algorithm Design and Analysis

   (a) State and solve a recurrence relation for the running time of the following algorithm. Assume the original call is
   rec-srch(A, 1, n, x).

   ```
   // given an array A indexed from l to r, of elements that can be compared,
   // and a key x, rec-srch(A,l,r) returns the location of the first
   // occurrence of key x in A[l..r], if x is in the array, and 0 otherwise.
   rec-srch(A, ℓ, r, x)
     if (ℓ > r) return 0
     if (x == A[ℓ])
       then return ℓ;
       else return rec-srch(A, ℓ+1, r, x);
   ```

   (b) Discuss one of the following algorithms: mergesort, heapsort, or quicksort. A complete answer will include a statement of the problem solved by the algorithm, any special characteristics of the algorithm, discussion of space and time requirements, an example of the way the algorithm works, and a discussion of the design technique(s) (for example, divide-and-conquer or dynamic programming) embodied by the algorithm.

3. CS5318 Design of Programming Languages

   (a) Discuss in detail and with examples the role of abstraction in programming language design.

   (b) Discuss in detail and with examples the use of context-free grammars to specify associativity and precedence in programming languages.

4. CS5338 Formal Languages

   (a) Give a precise definition of a pushdown automaton. Discuss how a pushdown automaton works.

   (b) Construct a minimal-state finite-state automaton to recognize the langaage $L = \{a^*b^*\}$. Discuss why $L$ can be recognized by a finite-state automaton but the language $K = \{a^nb^n\}$ cannot.

**Department of Computer Science**
**Graduate Comprehensive Exam**
**in**
**Computer Science**
**Fall 2002**

- **Answer the questions on the paper supplied.**
- **Answer question 1. Answer *either* question 2 *or* question 3. Answer *either* question 4 *or* question 5 *or* question 6.**
- **Start each question on a new page.**
- **Write on only one side of the paper.**
- **Write your SSN in the top right corner of each page of your answer. Do NOT put your name anywhere on the answers.**
- **Put the number of the question being answered in the top left corner of each page of your answer.**
- **If the answer to a question is written on more than one page, number the pages consecutively.**

1. CS5346 Advanced Artificial Intelligence

   (a) Explain the use of Forward and Backward chaining in Expert Systems in detail with examples.

   (b) It is known that Chris is the father of Mary. Prove that Chris is not the mother of John. You will need to write rules and represent the given facts in predicate logic. Use the resolution principle in your proof.

   (c) Using the A* algorithm, generate a search tree for the following initial node of the 8-puzzle. Use the best possible heuristic function and show how it satisfies the A* condition.

| | | |
|---|---|---|
| 5 | 4 | □ |
| 6 | 1 | 8 |
| 7 | 3 | 2 |

2. CS5318 Design of Programming Languages

   (a) One of the criteria used to evaluate programming languages is cost. Discuss in detail and with examples three costs associated with programming languages.

   (b) Discuss attribute grammars. A complete answer will include a comparison of the power or expressivity of attribute grammars and context-free grammars, an example of an attribute grammar, and an example of a decorated parse tree.

3. CS5338 Formal Languages

   (a) Show that the difference of two regular languages is regular.

   (b) Describe *in words* the operation of a Turing Machine with one or more tapes that accepts strings of the form $a^n b^n c^n$. The Turing Machine should start in its start state with a blank symbol in the leftmost tape square followed by the input string on the first (or only) tape, and, if the string is accepted, should stop in the halt state with the read head(s) over the leftmost tape square(s). Show the initial and final configurations , including state, tape contents, and read head position, for the string $a^2 b^2 c^2$. Do *not* construct the Turing Machine itself.

4. CS5310 Advanced Networks

   Describe the essential elements required for a client and a server to perform a session of TCP/IP based communication. For both connectionless and connection-oriented communications, describe how the client and the server will obtain those required essential elements.

5. CS5332 Advanced Database Systems

   Answer *either* part a *or* part b. If you answer both, only part a will be graded.

   (a) Discuss the significance of theory of functional dependencies. Discuss the relationships between 3NF and BCNF. Why are 3NF and BCNF are preferred over 2NF?

   (b) Complete the E/R model for the following entities, including necessary attributes, primary keys, foreign keys, necessary normalization and relationships:

   STUDENT CLASS STATUS CLASSROOM CLASS INSTRUCTOR COURSE

   Note:

   a. You may make appropriate assumptions if needed. But please state such assumptions clearly and concisely;
   b. Use 'P' for primary key and 'F' for foreign key.
   c. Class has section number and semester; Course has no section number.

**Department of Computer Science**
**Graduate Comprehensive Exam**
**in**
**Computer Science**
**Fall 2003**

- **Answer the questions on the paper supplied.**
- **Answer question 1. Answer** *either* **question 2** *or* **question 3. Answer** *either* **question 4** *or* **question 5** *or* **question 6. You should answer a total of three questions.**
- **Start each question on a new page.**
- **Write on only one side of the paper.**
- **Write your SSN in the top right corner of each page of your answer. Do NOT put your name anywhere on the answers.**
- **Put the number of the question being answered in the top left corner of each answer page.**
- **If the answer to a question is written on more than one page, number the pages consecutively.**

1. CS5391 Survey of Software Engineering

   The Capability Maturity Model (CMM) consists of five levels that seek to characterize the stability and maturity of the software development process within an organization. There are also a variety of software development lifecycle models available to software engineers. You are to describe two of the CMM levels (levels 2 and 4). In each description, provide the list of key practices that are in place at that level. Also include in the description the lifecycle model you feel is most likely to be in place in an organization at that CMM level. Give justification for your choice of lifecycle model in each case, making it clear exactly why you believe the lifecycle model you selected is the best fit at that CMM level.

2. CS5318 Design of Programming Languages

   (a) One of the criteria used to evaluate programming languages is cost. Discuss in detail and with examples three costs associated with programming languages.

   (b) Discuss attribute grammars. A complete answer will include a comparison of the power or expressivity of attribute grammars and context-free grammars, an example of an attribute grammar, and an example of a decorated parse tree.

3. CS5338 Formal Languages

   (a) Show that the intersection of two regular languages is regular.

   (b) Describe *in words* the operation of a Turing Machine with one or more tapes that accepts strings of the form $a^n b^n$. The Turing Machine should start in its start state with a blank symbol in the leftmost tape square followed by the input string on the first (or only) tape, and, if the string is accepted, should stop in the halt state with the read head(s) over the leftmost tape square(s). Show the initial and final configurations , including state, tape contents, and read head position, for the string $a^2b^2$. Do *not* construct the Turing Machine itself.

4. CS5310 Advanced Networks

   Describe the essential elements required for a client and a server to perform a session of TCP/IP based communication. For both connectionless and connection-oriented communications, describe how the client and the server will obtain those required essential elements.

5. CS5332 Advanced Database Systems

   Consider the following relational schema :

   Suppliers(sid, sname, address)

   Parts(pid, pname, color)

   Catalog(sid, pid, cost)

   Assume that the ids are integers, that cost is a real number, that all other attributes are strings, that the sid field in Catalog relation is a foreign key containing a supplier id, and that the pid in Catalog relation field is a foreign key containing a part id.

   Write SQL statements to answer each of the following queries:

   (a) Find the names of suppliers who supply some red part.

   (b) Find the sids of suppliers who supply some red or green part.

   (c) Find the sids of suppliers who supply some red part and some green part.

   (d) Find the sids of suppliers who supply every part.

   (e) Find pairs of sids such that the supplier with the first sid charges more for some part than the supplier with the second sid.

   (f) Find the pids of parts that are supplied by at least two different suppliers.

   (g) Find the pids of the most expensive part(s) supplied by suppliers named "Yosemite Sham".

6. CS5332 Advanced Database Systems

   Given a university environment, develop an E/R diagram (model) for the following possible relations: Student, Professor, Employee, Department, Course (in catalog; no class index), Class (in schedule; with class index number), and Classroom.

   Notes: You need to create appropriate attributes, primary keys, and optional foreign keys for each relation. You may also need to create additional relations and must normalize all relations whenever needed. Finally, specify all required relationships among entities.

**Department of Computer Science**
**Graduate Comprehensive Exam**
**in**
**Computer Science**
**Fall 2004**

- **Answer the questions on the paper supplied.**
- **Answer** *one of* **question 1 or question 2.. Answer** *one of* **question 3 or question 4. Answer** *one of* **question question 5, question 6, question 7, or question 8. You should answer a total of three questions.**
- **Start each question on a new page. Write on only one side of the paper.**
- **Write your Texas State ID in the top right corner of each page of your answer. Do NOT put your name anywhere on the answers.**
- **Put the number of the question being answered in the top left corner of each answer page.**
- **If the answer to a question is written on more than one page, number the pages consecutively.**

1. CS5391 Survey of Software Engineering

   (a) As a manager you ask a software engineer to add features to a software product. The engineer responds that adding these features makes the program incorrect since the specification for these features directly contradicts the specification for this product, and customers will not purchase an incorrect program. You agree with your engineer, however, you want to convince him that the effect of these specification discrepancies is not that serious and they shall appear only in few circumstances. How would you build you argument?

   (b) What percentage of the total cost of a software product goes to the maintenance stage? Explain the difference between corrective, adaptive, and perfective maintenances and show the percentage of the total maintenance cost that goes to each of these maintenances.

   (c) Suppose that your software product design is based on the pipe-and-filter architectural style. Even though the development is going according to a schedule, your manager suggests that a product may be completed faster by adding more software engineers to parallelize the development of this product. Do you agree with this suggestion? State why.

2. CS5391 Survey of Software Engineering

   Define cyclomatic complexity for a flow graph. How is cyclomatic complexity used in deriving test cases for source code?

   Generate the flow graph and compute the cyclomatic complexity for the following:

```
bool binarySearch(int *data,int &index,int lastPos, int target)
{
   int first, last, middle;
   first = 0;
   last = lastPos;
   while(last>first){
       middle=(last+first)/2;
       if(data[middle] == target)
           first=last+1;
       else if(data[middle]<target)
           first=middle+1;
       else
           last=middle-1;
   }
   if(data[middle] == target){
        index = middle;
        return true;
   }
   return false;
}
```

   SHOW ALL WORK!!

3. CS5318 Design of Programming Languages

   (a) Discuss three computational paradigms. A complete answer will include the model for the paradigm, one or more example languages, and characteristics of the paradigm.

   (b) Discuss attribute grammars. A complete answer will include a comparison of the power or expressivity of attribute grammars and context-free grammars, an example of an attribute grammar, and an example of a decorated parse tree.

4. CS5338 Formal Languages

   (a) Find a regular expression for each of the following languages.

      i. $\{w \in \{a, b\}* : w$ has $abab$ as a substring$\}$

      ii. $\{w \in \{a, b\}* : w$ has neither $aa$ nor $bb$ as a substring$\}$

   (b) Present Turing machines that decide the following languages over $\{a, b\}$:

      i. $L = \{a\}$

      ii. $L = \{a\}*$

      iii. $L = \{ab\}*$

5. CS5332 Advanced Database Systems

   *(20 marks total)* Consider the following typed relational schema describing books and borrowers in a library:

   Book(<u>isbn:integer</u>, title:string, publisher:string, year:integer)
   Author(<u>isbn:integer</u>, <u>name:string</u>, rank:integer)
   Borrower(<u>bid:integer</u>, name:string, address:string)
   Borrowings(<u>isbn:integer</u>, <u>borrower:integer</u>, whenTaken:date, whenReturned:date)

   Assume the following:

   - the fields that are underlined are the primary keys of the relations
   - there is a referential integrity constraint between "borrower" in relation Borrowings and "bid" in relation Borrower
   - there are also referential integrity constraints between "isbn" in relation Book and "isbn" in relations Author and Borrowings
   - dates are represented as a count of the number of days since Jan 1 2004
   - there is a system function `today` that returns today's `date`
   - the relational operators $<$ and $>$ allow you test *before* and *after* on `date`s
   - all loans are for a period of 14 days
   - the `whenReturned` field is `null` until the book is returned
   - the `rank` field specifies whether the author is the first, second, ... author

   Implement solutions to the following:

   (a) *(4 marks)* Write an SQL query to show all the books (`title` only) written by a particular author called "Ullman".

   (b) *(4 marks)* Write an SQL query to show which books (`title` only) have multiple authors.

   (c) *(4 marks)* Write an SQL query to show which books (`title` only) have never been borrowed.

   (d) *(4 marks)* Write an SQL query to show which books (`title` only) are currently out on loan.

   (e) *(4 marks)* Write an SQL query to list overdue books (`title`) and the borrower's name (Hints: you need to join relations Book, Borrowers and Borrowings).

6. CS5332 Advanced Database Systems

   Consider the following set of requirements for a university database that is used to keep track of student's transcripts.

   (a) The university keeps track of each student's name, student number, social security number, current address and phone, permanent address and phone, birthdate, sex, class (freshman, sophomore, ..., graduate), major department, minor department (if any), and degree program (B.A., B.S., ..., Ph.D.). Some user applications need to refer to the city, state, and zip code of the student's permanent address and to the student's last name. Both social security number and student number have unique values for each student.

   (b) Each department is described by a name, department code, office number, office phone, and college. Both name and code have unique values for each department.

   (c) Each course has a course name, description, course number, number of semester hours, level and offering department. The value of course number is unique for each course.

   (d) Each section has an instructor, semester, year, course, and section number. The section number distinguishes sections of the same course that are taught during the same semester/year; its values are 1,2,3,..., up to the number of sections taught during each semester.

   (e) A grade report has a student, section, letter grade, and numeric grade (0, 1, 2, 3, or 4).

   Design an ER schema for this application, and draw an ER diagram for that schema. Specify key attributes of each entity type and structural constraints on each relationship type. Note any unspecified requirements, and make appropriate assumptions to make the specification complete.

   **Remarks: ER schema consists of a set of normalized tables up to third normal form. You must place attributes of each normalized table in each entity of the ER diagram and indicate PK and FK for each entity. You must also describe the meaning of the notations you use for relationships. Your result of this problem should be a large ER diagram with description and notation explanation at the bottom of the ER diagram.**

7. CS5306 Advanced Operating Systems

   Describe three mutual exclusion methods in distributed systems, point out their advantages and disadvantages, and compare the number of messages communicated among processors.

8. CS5310 Advanced Networks

   Describe the essential elements required for a client and a server to perform a session of TCP/IP based communication. For both connectionless and connection-oriented communications, describe how the client and the server will obtain those required essential elements.

**Department of Computer Science**
**Graduate Comprehensive Exam**
**in**
**Computer Science**
**Fall 2005**

- **Answer the questions on the paper supplied.**
- **Answer question 1. Answer question 2 or question 3. Answer question 4 or question 5 or question 6. You should answer a total of three questions.**
- **Start each question on a new page. Write on only one side of the paper.**
- **Write your SIX-DIGIT Texas State ID in the top right corner of each page of your answer. Do NOT put your name anywhere on the answers.**
- **Put the number of the question being answered in the top left corner of each answer page.**
- **If the answer to a question is written on more than one page, number the pages consecutively.**

1. CS5391 Survey of Software Engineering

   Each of the following testing techniques can be used during software development. Describe the methodology for each technique, its goal and at which phase of software development it is used.

   (a) Program inspection.

   (b) Black–box testing.

   (c) White–box or structural testing.

   (d) Loading or stress testing.

   (e) Regression testing.

   (f) Incremental integration testing.

2. CS5318 Design of Programming Languages

   (a) Programming languages are sometimes classified by the computational paradigms they support. Discuss two computational paradigms. A complete answer will include the model underlying the paradigm, characteristics of languages that support the paradigm, and an example of a language that is most closely associated with the paradigm.

   (b) Discuss, with examples, the roles of context free grammars and attribute grammars in programming language design.

3. CS5338 Formal Languages.

   (a) Give a precise definition of a Turing Machine.

   (b) Describe in a phrase or sentence the language accepted by DFA $M$:

$$M = (\{q_0, q_1, q_2\}, \{0, 1\}, \delta, q_0, \{q_2\})$$

   where $\delta$ is given by the following table:

| $\delta$ | 0 | 1 |
|---|---|---|
| $q_0$ | $q_1$ | $q_0$ |
| $q_1$ | $q_2$ | $q_0$ |
| $q_2$ | $q_2$ | $q_0$ |

   (c) Minimize the following DFA:

$$M = (\{q_0, q_1, q_2\}, \{0, 1\}, \delta, q_0, \{q_1, q_2\})$$

   where $\delta$ is given by the following table:

| $\delta$ | 0 | 1 |
|---|---|---|
| $q_0$ | $q_1$ | $q_0$ |
| $q_1$ | $q_2$ | $q_0$ |
| $q_2$ | $q_2$ | $q_0$ |

   (d) Discuss the closure properties of regular languages. A complete answer will include a list of operations under which the regular languages are closed and examples.

4. CS5306 Advanced Operating Systems

   In a certain system, each process typically uses a critical section many times before another process requires it. Explain why Ricart and Agrawalas multicast-based mutual exclusion algorithm is inefficient for this case, and describe how to improve its performance.

5. CS5310 Network and Communication Systems

   Briefly describe the seven layer ISO's OSI model (For each layer briefly describe its services and design issues). Explain why the top four layers are called "end-to-end" layers. You should support your explanation with an application example.

6. CS5332 Database Theory and Design *(26 marks total)*
   The grade book database consists of the relations defined in the six tables shown below:

   Catalog(<u>cno</u>, ctitle)
   Students(<u>sid</u>, fname, lname, middlenameInitial)
   Courses(<u>term, lineno</u>, cno, A, B, C, D)
   Components(<u>*term, lineno*</u>, <u>compname</u>, maxpoints, weight)
   Enrolls(<u>*sid, term, lineno*</u>)
   Scores(<u>*sid, term, lineno, compname*</u>, points)

   Primary keys are underlined. Foreign keys are shown in italic font. Primary keys that are also foreign keys are shown in underline italic font. For example, the term and lineno attributes in Components relation is part of the primary key for Components and is also a foreign key of the table Courses.

   The Catalog relation contains information about course numbers and the title of the course. For example cs5332 is a course number for 'Database Theory and Design'. The Student relation contains information about the students with sid as the primary key for that relation. In Courses relation, the *term* attribute corresponds to the term (Fall2000 or Spring2005) in which the course was taught; the *lineno* is a unique section number assigned by the registrar of the university. A, B, C and D are attributes that represent numerical values that correspond with each letter grade. The Components relation contains information about the various grading components (such as quizzes, mid-term, final, lab) for a particular course. The combination of *sid*, *term*, *lineno* and *compname* forms the primary key.

   The Enrolls relation records information about which student was enrolled in which course. The Scores relation records the grading component scores (or points) for each student enrolled in a course.

   Write SQL to answer the following queries:

   (a) *(5 marks)* Get the names of students enrolled in the 'Database Theory and Design' class in the 'Spring2005' term?
   (b) *(5 marks)* Get the SID values of students who did not enrolled in any class during the Spring2005 term.
   (c) *(5 marks)* Get the SID values of students who have enrolled in cs5332 or cs5369A.
   (d) *(5 marks)* Get the SID values of students who have enrolled in cs5332 and cs5369A
   (e) *(6 marks)* Get the SID values of students who have enrolled in all the courses in the Catalog (hint: division)

**Department of Computer Science**
**Graduate Comprehensive Exam**
**in**
**Computer Science**
**Fall 2006**

- **Answer the questions on the paper supplied.**
- **Answer question 1 or question 2. Answer question 3. Answer question 3 or question 4 or question 5 or question 6. You should answer a total of three questions.**
- **Start each question on a new page. Write on only one side of the paper.**
- **Write your SIX-DIGIT Texas State ID in the top right corner of each page of your answer. Do NOT put your name anywhere on the answers.**
- **Put the semester and year of the exam in the appropriate blank at the top of each answer page.**
- **Put the number of the question being answered in the appropriate blank on each answer page.**
- **If the answer to a question is written on more than one page, number the pages consecutively.**

1. CS5329 Algorithm Design and Analysis
   (a) Write the QUICK SORT algorithm with its PARTITION function.
   (b) Use a Decision Tree in best case partition to describe the total running time as $\Theta(n \lg n)$.

2. CS5329 Algorithm Design and Analysis
   (a) Recurrences.
      i. Setup a recurrence relation for the worst-case time complexity of the following algorithm. This algorithm moves n rings from spindle a to spindle b using spindle c for temporary storage. Assume that the function MOVE1(A,B) runs in constant time.

```
RINGS(n,a,b,c)
  if (n ≤ 0) then return;
  if (n == 1) then  MOVE1(a,b);
  RINGS(n-1,a,c,b);
  MOVE1(a,b);
  RINGS(n-1,c,b,a);
```

      ii. Can the recurrence relation from part 2(a)i be solved using the Master Method?
      iii. Show the steps in solving the following recurrence:

$$T(n) = \begin{cases} 1 & n = 1 \\ 3T(n/3) + n^2 & n > 1 \end{cases}$$

   (b) Design Techniques. Discuss the methodology, or elements, for algorithm design using dynamic programming. A complete answer will include a description of each step, or element, and the application of each step to a problem that has an efficient dynamic programming solution.

3. CS5346 Advanced Artificial Intelligence

(a) Verify the soundness of the unit resolution rule by constructing a truth table.

(b) In the Block World problem, the following operations are used:

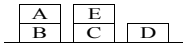
STACK($x$, $y$)
- P: CLEAR($y$) ^ HOLDING($x$)
- D: CLEAR($y$) ^ HOLDING($x$)
- A: ARMEMPTY ^ ON($x$, $y$)

UNSTACK($x$, $y$)
- P: ON($x$, $y$) ^ CLEAR($x$) ^ ARMEMPTY
- D: ON($x$, $y$) ^ ARMEMPTY
- A: HOLDING($x$) ^ CLEAR($y$)

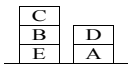
PICKUP($x$)
- P: CLEAR($x$) ^ ONTABLE($x$) ^ ARMEMPTY
- D: ONTABLE($x$) ^ ARMEMPTY
- A: HOLDING($x$)

PUTDOWN($x$)
- P: HOLDING($x$)
- D: HOLDING($x$)
- A: ONTABLE($x$) ^ ARMEMPTY

The following are the start and goal states. Using the Goal Stack Planning method, describe each step which needs to be performed to transform the start state to the goal state.




start: ON(A, B) ^
ON (E, C) ^
ONTABLE(B) ^
ONTABLE(C) ^
ONTABLE(D) ^
ARMEMPTY ^
CLEAR(A) ^
CLEAR(E) ^
CLEAR(D)




goal: On(B, E) ^
ON(C, B) ^
ON(D, A) ^
ONTABLE (E)
ONTABLE (A)
CLEAR (C)
CLEAR (D)
ARMEMPTY

4. CS5306 Advanced Operating Systems

   (a) Why do stateless servers have to include a file offset in each request? Is this also needed for statefull servers?

   (b) A file is replicated on 10 servers.

      i. explain the voting algorithm technique (Giffords algorithm).

      ii. List all combinations of read quorum and write quorum that are permitted by the voting algorithm.

   (c) Describe two election algorithms in distributed systems, point out their advantages and disadvantages.

5. CS5310 Network and Communication Systems

   What is modulation? Why modulations are needed? What are popular modulation techinques? Please provide three examples of real applications where modulations are essential. Please provide necessary details to support your argument.

6. CS5332 Database Theory and Design

   From what you understand about the Texas State University system, draw an Entity-Relationship Diagram (E/R Model) including at least, but not limited to, the following entities:

   CLASS, CLASSROOM
   STUDENT, COURSE, INSTRUCTOR

   CLASS is the class for a particular semester; COURSE is the course described in the catalog.

   You must figure out correct primary keys, correct foreign keys and correct relationships. You must normalize all entities (tables). You must specify other essential attributes for all entities. You must also create other entities (tables) if needed.

7. CS5332 Database Theory and Design The grade book database consists of the relations defined in the six tables shown below:

   Catalog(<u>cno</u>, ctitle)
   Students(<u>sid</u>, fname, lname, middlenameInitial)
   Courses(<u>term, lineno</u>, cno, A, B, C, D)
   Components(<u>*term, lineno*</u>, <u>compname</u>, maxpoints, weight)
   Enrolls(<u>*sid, term, lineno*</u>)
   Scores(<u>*sid, term, lineno, compname*</u>, points)

   Primary keys are underlined. Foreign keys are shown in italic font. Primary keys that are also foreign keys are shown in underline italic font. For example, the term and lineno attributes in Components relation is part of the primary key for Components and is also a foreign key of the table Courses.

   The Catalog relation contains information about course numbers and the title of the course. For example cs5332 is a course number for 'Database Theory and Design'. The Student relation contains information about the students with sid as the primary key for that relation. In Courses relation, the *term* attribute corresponds to the term (Fall2000 or Spring2005) in which the course was taught; the *lineno* is a unique section number assigned by the registrar of the university. A, B, C and D are attributes that represent numerical values that correspond with each letter grade. The Components relation contains information about the various grading components (such as quizzes, mid-term, final, project) for a particular course. The combination of *sid*, *term*, *lineno* and *compname* forms the primary key.

   The Enrolls relation records information about students who were enrolled in a specific course. The Scores relation records the grading component scores (or points) for each student enrolled in a course.

   Write SQL to answer the following queries:

   (a) *(5 marks)* Get the names of students enrolled in the 'Database Theory and Design' class in the 'Spring2005' term?
   (b) *(5 marks)* Get the SID values of students who did not enrolled in any class during the Spring2005 term.
   (c) *(5 marks)* Get the SID values of students who have enrolled in cs5332 or cs5369a.
   (d) *(5 marks)* Get the SID values of students who have enrolled in cs5332 and cs5369a
   (e) *(6 marks)* Get the SID values of students who have enrolled in all the courses in the Catalog (hint: division)

**Department of Computer Science**
**Graduate Comprehensive Exam**
**in**
**Computer Science**
**Fall 2007**

- **Answer the questions on the paper supplied.**
- **Answer question 1. Answer question 2 or question 3. Answer question 4 or question 5 or question 6 or question 7. You should answer a total of three questions.**
- **Start each question on a new page. Write on only one side of the paper.**
- **Write your SIX-DIGIT Texas State ID in the top right corner of each page of your answer. Do NOT put your name anywhere on the answers.**
- **Put the semester and year of the exam in the appropriate blank at the top of each answer page.**
- **Put the number of the question being answered in the appropriate blank on each answer page.**
- **If the answer to a question is written on more than one page, number the pages consecutively.**

1. CS5346 Advanced Artificial Intelligence

   (a) Explain in detail with example, the difference between the Uniform Cost Search and A* Search.

   (b) Transform the following English sentences first into a predicate logic form and then either to conjunctive normal form or implicative normal form:

      1. Marcus was a man.
      2. Marcus was a Pompeian.
      3. All Pompeians were Romans.
      4. Caesar was a ruler.
      5. All Romans were either loyal to Caesar or hated him.
      6. Everyone is loyal to someone.
      7. People only try to assassinate rulers they are not loyal to.
      8. Marcus tried to assassinate Caesar.

      Using the refutation rule, prove Marcus hated Caesar.

2. CS5329 Algorithm Design and Analysis

   (a) State and solve a recurrence relation for the running time of the following algorithm. Assume the original call is
   rec-srch(A, 1, n, x).

```
// given an array A indexed from l to r, of elements that can be compared,
// and a key x, rec-srch(A,l,r) returns the location of the first
// occurrence of key x in A[l..r], if x is in the array, and 0 otherwise.
rec-srch(A, ℓ, r, x)
   if (ℓ > r) return 0
   if (x == A[ℓ])
      then return ℓ;
      else return rec-srch(A, ℓ+1, r, x);
```

   (b) Discuss in detail and with examples the greedy algorithm design technique. A complete answer will include a discussion of characteristics of problems for which a greedy strategy yields an efficient solution and a comparison with other design techniques such as divide-and-conquer and dynamic programming.

3. CS5329 Algorithm Design and Analysis

   Suppose that the splits at every level of quicksort are in the proportion $1-\alpha$ to $\alpha$, where $0 < \alpha \leq 1/2$ is a constant. show that the minimum depth of a leaf in the recursion tree is approximately $-\lg n/\lg \alpha$ and the maximum depth is approximately $-\lg n/\lg(1-\alpha)$

4. CS5306 Advanced Operating Systems

   What are the two rules in the Lamports happens-before relationship? Can happens-before relationship provide a total order of the events in a distributed system? If so, why? If not, how to provide a total order? A process can fork a child process. How to modify the happens-before relationship to include this situation?

5. CS5310 Network and Communication Systems

   Answer the following questions based on the TCP/IP model:

   (a) What are the main differences between TCP and UDP based on congestion control?
   (b) What is the purpose of the transport layer and why it is referred to as End-to-End?
   (c) Explain briefly how IP packets are forwarded by routers.

6. CS5332 Database Theory and Design

   Draw an extended ER diagram for the following business rules: You may create more attributes to complete the problem. You must normalize the entities, and specify pks and fks:

   Design a database for Ray. He is interested in movies and wants to keep information on movies, actors, and directors in a database. The only user is Ray and he needs to produce the following reports:

   (a) For each director, list his or her number, name, and the year he or she was born. If the director is deceased, list the year of death.
   (b) For each movie, list its number, title, the year the movie was made, and its type (for example, Comedy, Drama, Science Fiction, and so on).
   (c) For each movie, list its number, title, the number and name of its director, the critics rating, the MPAA rating (G, PG, PG-13, or R), the number of awards for which the movie was nominated, and the number of awards the movie won. (The critics rate the movie with a number of "stars." Four stars is the top rating possible. Zero stars is the worst possible rating.)
   (d) For each actor, list his or her number, name, birthplace, and the year he or she was born. If the actor is deceased, list the year of death.
   (e) For each movie, list its number and title, along with the number and name of the actors who appeared in it.
   (f) For each actor, list his or her number and name, along with the number and name of the movies in which the actor starred.

7. CS5332 Database Theory and Design

*(20 marks total)*

The customer database consists of the relations defined in the four tables shown below:

Customer(<u>custNo</u>, custName, state, phone)
Item(<u>itemNo</u>, ItemName, itemPrice, qtyOnHand)
Invoice(<u>invNo</u>, inDate, *custNo*)
Invitem(<u>*invNo, itemNo*</u>, qty)

Primary keys are underlined. Foreign keys are shown in italic font. Primary keys that are also foreign keys are shown in underline italic font. For example, the invNo and itemNo attributes in Invitem relation is the primary key for Invitem and is also foreign keys of tables Invoice and Item. Below are sample data from each of the above relation.

```
Customer table
--------------
custNo   custName    state     phone
211       Garcia      TX       732-555-1000
212       Parkin      NY       212-555-2000
225       Ellen       NJ       973-555-3333
239       Bayer       FL       401-555-7777

Item table
----------
itemNo     itemName   itemPrice     qtyOnHand
 1          Screw      5.00           50
 2          Nut        2.25           110
 3          Bolt       3.99           75
 4          Hammer     10.00          125
 5          Washer     6.00           300
 6          Nail       0.99           400

Invoice table
-------------
invNo      invDate        custNo
1001        05-Sep-07      212
1002        17-Sep-07      225
1003        17-Sep-07      239
1004        18-Sep-07      211
1005        21-Sep-07      212

invItem table
-------------
invNo        itemNo           qty
1001          1                5
1001          3                5
1001          5                9
1002          1                2
1002          2                3
1003          1                7
1003          2                1
1004          4                5
1005          4                10
```

Write SQL to answer the following queries:

(a) *(5 marks)* Find items which have the itemPrice between $5 and $10?

(b) *(5 marks)* Count the number of items ordered in each invoice.

(c) *(5 marks)*

(d) *(5 marks)* Find items that are cheaper than nut. You must display the itemName and the associated itemPrice. Beware that nut can be written in both upper and lower case.

(e) *(5 marks)* Find the invoices in which three or more items are ordered.

(f) *(5 marks)* List the invoices, customer names and the items purchased by them. The result should display invoices, customer names and item names together ordered by the customer name.

**Next:** About this document

**Department of Computer Science**
**Graduate Comprehensive Exam**
**Spring 1997**

- **Answer the questions on the paper supplied. Be sure to answer questions 1, 2, and *either* 3 *or* 4, depending upon how you have registered.**
- **Start each question on a new page.**
- **Do NOT write on the back of the paper.**
- **Write your SSN in the top right corner of each page of your answer. Do NOT put your name anywhere on the answers.**
- **Put the number of the question being answered in the top left corner of each page of your answer.**
- **If the answer to a question is written on more than one page, number the pages consecutively.**

1. (CS5391) Discuss the activities (steps or actions) that are performed during the requirements engineering phase of the software life cycle. Be sure to address the outputs from each activity.
2. (CS5346) Answer each of the following:

    1. Explain "admissible heuristic." Prove the optimality of A* algorithm. Show by examples the completeness of A* algorithm. Also, explain the complexity of A* algorithm.
    2. Explain with examples the Minmax search procedure. How are utility functions, payoff functions and evaluation functions used in Minmax search?

Answer **exactly one** of the following (CS5338 *or* CS5318):

3. (CS5338) Consider finite-state automata, pushdown automata, and Turing machines.

    1. Define each of these machines, including important variations.
    2. Discuss their significance, applications, and limitations.
    3. Using the least powerful machine category for each problem, construct a machine that recognizes each of tex , tex2h , and tex2html .

4. (CS5318) Explain in detail and with examples the following:

    1. BNF and EBNF grammars
    2. Attribute grammars
    3. Translational semantics
    4. Axiomatic semantics

---

- About this document ...

---

*Carol Hazlewood*
*Mon Nov 23 11:50:55 CST 1998*

**Next:** About this document

**Department of Computer Science**
**Graduate Comprehensive Exam**
**Spring 1998**

- **Answer the questions on the paper supplied. Be sure to answer question 1, question 2, and question 3.**
- **Start each question on a new page.**
- **Do NOT write on the back of the paper.**
- **Write your SSN in the top right corner of each page of your answer. Do NOT put your name anywhere on the answers.**
- **Put the number of the question being answered in the top left corner of each page of your answer.**
- **If the answer to a question is written on more than one page, number the pages consecutively.**

1. CS5332 Database Theory and Design

   Discuss the significance of theory of functional dependencies. Discuss the relationships between 3NF and BCNF. Why are 3NF and BCNF are preferred over 2NF?

2. CS5346 Advanced Artificial Intelligence

   1. Show by examples how artificial intelligence techniques are different from conventional techniques.
   2. Explain intelligent agents and their characteristics.
   3. Compare at least four search strategies based on the four criteria.
   4. Explain, with examples, simplified memory - Bounded A* search.

3. CS5391 Survey of Software Engineering

   Contrast structural analysis and design methods and object- oriented analysis and design methods. Focus your discussion on the different views (models) of the problem domain and the ability of each of these methods to express these views.

---

- About this document ...

---

*Carol Hazlewood*
*Mon Nov 23 14:36:57 CST 1998*

**Next:** About this document

**Department of Computer Science**
**Graduate Comprehensive Exam**
**Spring 1999**

- **Answer the questions on the paper supplied.**
- **Answer question 1. Answer question 2. Answer *one of* questions 3 *or* 4, depending on how you registered.**
- **Start each question on a new page.**
- **Write on only one side of the paper.**
- **Write your SSN in the top right corner of each page of your answer. Do NOT put your name anywhere on the answers.**
- **Put the number of the question being answered in the top left corner of each page of your answer.**
- **If the answer to a question is written on more than one page, number the pages consecutively.**

1. CS5346 Advanced Artificial Intelligence

    1. Explain forward and backward chaining in detail with examples.
    2. Explain in detail with examples admissibility, optimality, and complexity of the A* algorithm.
    3. It is known that Chris is the mother of Mary. Prove that Chris is not the father of Bill. You will need to write at least two rules and represent the given facts in predicate logic. Use the resolution principle in your proof.

2. CS5329 Algorithm Design and Analysis

    1. Design Techniques.
        1. Discuss how dynamic programming and greedy techniques work.
        2. Describe characteristics of problems that make them candidates for solution by each technique.
        3. Give examples of problems that can be solved by each technique and explain why that technique is advantageous.
    2. Design a divide-and-conquer algorithm to find the minimal and maximal elements in a set. Write a recurrence relation for your algorithm and show that your algorithm uses at most 3n/2 comparisons. Assume n is a power of two.

3. CS5318 Design of Programming Languages
    1. Discuss in detail and with examples criteria for programming language evaluation.
    2. Discuss in detail and with examples the role of abstraction in programming languages.

3. Complete an algebraic specification for the following abstract *list* data type. Use it to show first(tail(cons(x,L)))) = if isempty(L) then error else first(L).

tabular32

4. CS5338 Formal Languages

   1. Show that the union of two context-free languages is context-free.
   2. Show that the complement of a regular language is regular.
   3. Construct a Turing Machine to compute the sum of two non-negative integers. A machine starting in the start state with input tex2html_ should reach a halt state with output tex2html , where the underline indicates the postion of the read/write head. Describe the operation of your machine.

---

- About this document ...

**Next:** About this document

*Carol Hazlewood*
*Tue Apr 13 15:41:17 CDT 1999*

**Next:** About this document

**Department of Computer Science**
**Graduate Comprehensive Exam**
**in**
**Computer Science**
**Spring 2000**

- **Answer the questions on the paper supplied.**
- **Answer question 1. Answer question 2. Answer *either* question 3 *or* question 4, but not both.**
- **Start each question on a new page.**
- **Write on only one side of the paper.**
- **Write your SSN in the top right corner of each page of your answer. Do NOT put your name anywhere on the answers.**
- **Put the number of the question being answered in the top left corner of each page of your answer.**
- **If the answer to a question is written on more than one page, number the pages consecutively.**

1. CS5329 Algorithm Design and Analysis
    1. Solve the following recurrences:

       tabular25

    2. Design Techniques.
       Discuss the divide-and-conquer and dynamic programming techniques. Describe each technique, list characteristics of problems that make them amenable to solution by each technique, and show how the techniques are used in examples of problems that can be solved by each technique.

2. CS5346 Advanced Artificial Intelligence
    1. Explain "admissible heuristic". Prove the optimality of A* algorithm. Show by examples the completeness of A* algorithm. Also, explain the complexity of A* algorithm.
    2. Explain with examples the Minmax search procedure. How are utility functions, payoff functions and evaluation functions used in Minmax search?

3. CS5310 Network and Communication Systems
   Describe the essential elements required for a client and a server to perform a session of TCP/IP based communication. For both connectionless and connection-oriented communications, describe how the client and the server will obtain those required essential elements.

4. CS5332 Data Base Theory and Design
    1. Discuss the significance of theory of functional dependencies.
    2. Discuss with examples the relationships between 3NF and BCNF. Why are 3NF and BCNF are preferred over 2NF?

- About this document ...

---

**Next:** About this document

*Carol Hazlewood*
*Thu Jun 16 14:52:21 CDT 2000*

# Department of Computer Science
# Graduate Comprehensive Exam
# in
# Computer Science
# Spring 2001

**Directions:**

- **Answer the questions on the paper supplied.**
- **Answer question 1. Answer question 2. Answer question 3.**
- **Start each question on a new page.**
- **Write on *only one side* of the paper.**
- **Write your SSN in the top right corner of each page of your answer. Do NOT put your name anywhere on the answers.**
- **Put the number of the question being answered in the top left corner of each page of your answer.**
- **If the answer to a question is written on more than one page, number the pages consecutively.**

1. CS5329 Algorithm Design and Analysis

   a. State and solve a recurrence relation for the running time of the following algorithm:

```
// given an array A indexed from l to r, l <= r,
// and containing elements that can be compared,
// rec-max(A,l,r) returns the maximum element in A[l..r].
rec-max(A,l,r)
    if (l == r) return A[l];
    m = rec-max(A,l+1,r);
    if (A[l] >= m) return A[l] else return m;
```

   b. Discuss *one* of the following algorithms: mergesort, heapsort, *or* quicksort. A complete answer will include a statement of the problem solved by the algorithm, any special characteristics of the algorithm, discussion of space and time requirements, an example of the way the algorithm works, and a discussion of the design technique(s) (for example, divide-and-conquer or dynamic programming) embodied by the algorithm.

2. CS 5391 Survey of Software Engineering

   What affect does the choice of life cycle model have on the design process? Describe how the design process works within the various life cycle models (linear sequential, prototyping, incremental, spiral). How does the choice of life cycle model affect the software testing process? What is the relationship between life cycle models and the capability maturity model?

3. CS 5346: Advanced Artificial Intelligence

a. Explain forward chaining and backward chaining in detail with examples.

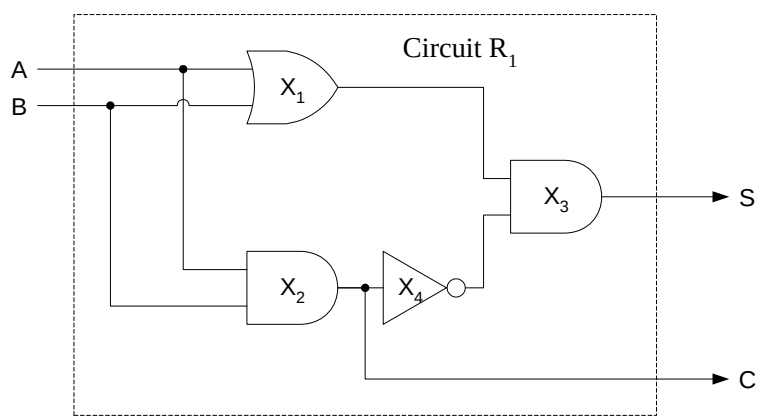
b. The following general rules are followed in digital circuits. Rules 4-6 cover either 'ON' or 'OFF' states of the gates. Add three more rules to cover both 'ON' and 'OFF' states.

1. If two terminals are connected, then they have the same signal:
   $\forall\, t_1, t_2\ Connected(t_1, t_2) \Rightarrow Signal(t_1) = Signal(t_2)$
2. The signal at every terminal is either on or off (but not both):
   $\forall\, t\ Signal(t) = On - Signal(t) = Off$
   $On \neq Off$
3. Connected is a commutative predicate:
   $\forall\, t_1, t_2\ Connected(t_1, t_2) \Leftrightarrow Connected(t_2, t1)$
4. An OR gate's output is on if and only if any of its inputs are on:
   $\forall\, g\ Type(g) = OR \Rightarrow$
   $Signal(Out(1, g)) = On \Leftrightarrow \exists n\ Signal\ (In(n, g)) = On$
5. An AND gate's output is off if and only if any of its inputs are off:
   $\forall\, g\ Type(g) = AND \Rightarrow$
   $Signal(Out(1, g)) = Off \Leftrightarrow \exists n\ Signal\ (In(n, g)) = Off$
6. An XOR gate's output is on if and only if its inputs are different:
   $\forall\, g\ Type(g) = XOR \Rightarrow$
   $Signal(Out(1, g)) = On \Leftrightarrow Signal\ (In(1, g)) \neq Signal\ (In(2, g))$
7. A NOT gate's output is different from its input:
   $\forall\, g\ (Type(g) = NOT) \Rightarrow Signal\ (Out(1, g)) \neq Signal\ (In(1, g))$

The following is the description of the gate types of the circuit $R_1$ given below:

$Type\ (X_1) = OR$ $\qquad$ $Type\ (X_2) = AND$

$Type\ (X_3) = AND$ $\qquad$ $Type\ (X_4) = NOT$




Represent the connections of the circuit and determine what combinations of Sig ($In(A, R_1)$), Sig ($In(B, R_1)$) would cause the following output:
($OUT(S, R_1) = OFF$) . ($OUT(C, R_1) = ON$)

**Department of Computer Science**
**Graduate Comprehensive Exam**
**in**
**Computer Science**
**Spring 2002**

- **Answer the questions on the paper supplied.**
- **Answer question 1. Answer *either* question 2 *or* question 3. Answer *either* question 4 *or* question 5.**
- **Start each question on a new page.**
- **Write on only one side of the paper.**
- **Write your SSN in the top right corner of each page of your answer. Do NOT put your name anywhere on the answers.**
- **Put the number of the question being answered in the top left corner of each page of your answer.**
- **If the answer to a question is written on more than one page, number the pages consecutively.**

1. CS5346 Advanced Artificial Intelligence

   (a) Consider the following sentences:
   - Steve only likes easy courses.
   - Science courses are hard.
   - All the courses in the theater department are easy.
   - TH301 is a theater course.
   - Kate likes all courses Steve likes.

   i. Translate these sentences into formulas in predicate logic.
   ii. Convert the formulas into clause form.
   iii. Use resolution to answer the question: "What course does kate like?"
   iv. Prove that Steve likes TH301 using backward chaining.

   (b) Given the following sentences:
   - If x is on top of y, y supports x.
   - If x is above y and they are touching each other, x is on top of y.
   - A cup is above a book.
   - A pen is touching a cup.
   - A cup is touching a book.

   i. Translate these sentences into formulas in predicate logic.
   ii. Convert the formulas into clause form.
   iii. Use resolution to answer the question: "Does a book support a pen?"
   iv. Prove that a book supports a cup using backward chaining.

   (c) Describe the Minimax search using an example. Describe how alpha-beta pruning improves the minimax search using an example.

2. CS5318 Design of Programming Languages

   (a) Discuss in detail and with examples the criteria of writability and readability in evaluating programming languages.
   (b) Discuss in detail and with examples the use of context-free grammars to specify associativity and precedence in programming languages.

3. CS5338 Formal Languages

   (a) Give a precise definition of a Turing Machine. Discuss briefly how a Turing Machine works.
   (b) Construct a minimal-state finite-state automaton to recognize all strings of 0's and 1's that do not contain three consecutive 0's.
   (c) Show that the intersection of two regular languages is regular.

4. CS5310 Network and Communication Systems

   Briefly describe the seven layer ISO's OSI model (For each layer briefly describe its services and design issues). Explain why the top four layers are called "end-to-end" layers. You should support your explanation with an application example.

5. Data Base Theory and Design

   Discuss the significance of theory of functional dependencies. Discuss the relationships between 3NF and BCNF. Why are 3NF and BCNF are preferred over 2NF?

**Department of Computer Science**
**Graduate Comprehensive Exam**
**in**
**Computer Science**
**Spring 2003**

- **Answer the questions on the paper supplied.**
- **Answer question 1. Answer question 2. Answer *either* question 3 *or* question 4.**
- **Start each question on a new page.**
- **Write on only one side of the paper.**
- **Write your SSN in the top right corner of each page of your answer. Do NOT put your name anywhere on the answers.**
- **Put the number of the question being answered in the top left corner of each answer page.**
- **If the answer to a question is written on more than one page, number the pages consecutively.**

1. CS5329 Algorithm Design and Analysis

   (a) Discuss in detail and with examples the greedy algorithm design technique. A complete answer will include a discussion of characteristics of problems for which a greedy strategy yields an efficient solution and a comparison with other design techniques such as divide-and-conquer and dynamic programming.

   (b) Setup a recurrence relation for the worst-case time complexity of the following algorithm. Show the steps in solving the recurrence. Assume the original call is DOWNHEAP(A,1) on a heap A of size n indexed from 1 to n. Also assume that the function FIND-LARGEST runs in constant time and a subtree of a heap of size n is no larger than 2n/3.

```
// given a binary treee represented by an array A, indexed from 1 to n, and an index i,
// downheap(A, i) compares node i with its two children, moves the largest of the three
// elements to position i, and recurses down the tree until no elements are moved.

DOWNHEAP(A,i)
  ℓ = FIND-LARGEST(A,i)
  if (ℓ !=i)
    then
      swap(A[i],A[ℓ]);
      DOWNHEAP(A,ℓ);
```

2. CS5391 Survey of Software Engineering

   There are five levels to the capability maturity model (CMM). Discuss each of these levels, including the key practices that are in place at each level and which practices are needed to advance to the next level.

3. CS5310 Advanced Networks

   Describe the essential elements required for a client and a server to perform a session of TCP/IP based communication. For both connectionless and connection-oriented communications, describe how the client and the server will obtain those required essential elements.

4. CS5332 Advanced Database Systems

   Discuss the significance of theory of functional dependencies. Discuss the relationships between 3NF and BCNF. Why are 3NF and BCNF are preferred over 2NF?

**Department of Computer Science**
**Graduate Comprehensive Exam**
**in**
**Computer Science**
**Spring 2004**

- **Answer the questions on the paper supplied.**
- **Answer question 1. Answer** *either* **question 2** *or* **question 3. Answer** *either* **question 4** *or* **question 5** *or* **question 6** *or* **question 7. You should answer a total of three questions.**
- **Start each question on a new page.**
- **Write on only one side of the paper.**
- **Write your Texas State ID in the top right corner of each page of your answer. Do NOT put your name anywhere on the answers.**
- **Put the number of the question being answered in the top left corner of each answer page.**
- **If the answer to a question is written on more than one page, number the pages consecutively.**

1. CS5346 Advanced Artificial Intelligence

   (a) Prove that if the heuristic function $h$ obeys the triangle inequality then the values of $f(n)$ along any path are non-decreasing.

   (b) Explain, with examples, the differences in the perfect decisions and imperfect decisions methodologies employed in two person games.

   (c) It is known that Chris is the son of Nancy. Prove that Chris is not the daughter of Mike. You will need to write at least two implications and represent the given facts in predicate logic. Use refutation principle in your proof.

2. CS5318 Design of Programming Languages

   (a) Regularity is one of the criteria used to evaluate programming languages. Regularity is often subdivided into areas of generality, uniformity and orthogonality. Discuss in detail and with examples these three sub-criteria.

   (b) Discuss attribute grammars. A complete answer will include a comparison of the power or expressivity of attribute grammars and context-free grammars, an example of an attribute grammar, and an example of a decorated parse tree.

3. CS5338 Formal Languages

   (a) Show that the difference of two regular languages is regular.

   (b) Construct a deterministic finite-state automaton and a Turing Machine to recognize the language $a^*b^*$.

4. CS5306 Advanced Operating Systems

   Describe three mutual exclusion methods in distributed systems, point out their advantages and disadvantages, and compare the number of messages communicated among processors.

5. CS5310 Advanced Networks

   Describe the essential elements required for a client and a server to perform a session of TCP/IP based communication. For both connectionless and connection-oriented communications, describe how the client and the server will obtain those required essential elements.

6. CS5332 Advanced Database Systems

   Consider the following relational schema :

   Suppliers(sid, sname, address)

   Parts(pid, pname, color)

   Catalog(sid, pid, cost)

   Assume that the ids are integers, that cost is a real number, that all other attributes are strings, that the sid field in Catalog relation is a foreign key containing a supplier id, and that the pid in Catalog relation field is a foreign key containing a part id.

   Write SQL statements to answer each of the following queries:

   (a) Find the names of suppliers who supply some red part.
   (b) Find the sids of suppliers who supply some red or green part.
   (c) Find the sids of suppliers who supply some red part and some green part.
   (d) Find the sids of suppliers who supply every part.
   (e) Find pairs of sids such that the supplier with the first sid charges more for some part than the supplier with the second sid.
   (f) Find the pids of parts that are supplied by at least two different suppliers.

7. CS5332 Advanced Database Systems

   Given a university environment, develop an E/R diagram (model) for the following possible relations: Student, Professor, Employee, Department, Course (in catalog; no class index), Class (in schedule; with class index number), and Classroom.

   Notes: You need to create appropriate attributes, primary keys, and optional foreign keys for each relation. You may also need to create additional relations and must normalize all relations whenever needed. Finally, specify all required relationships among entities.

**Department of Computer Science**
**Graduate Comprehensive Exam**
**in**
**Computer Science**
**Spring 2005**

- **Answer the questions on the paper supplied.**
- **Answer question 1 or question 2. Answer question 3. Answer question 4. You should answer a total of three questions.**
- **Start each question on a new page. Write on only one side of the paper.**
- **Write your SIX-DIGIT Texas State ID in the top right corner of each page of your answer. Do NOT put your name anywhere on the answers.**
- **Put the number of the question being answered in the top left corner of each answer page.**
- **If the answer to a question is written on more than one page, number the pages consecutively.**

1. CS5391 Survey of Software Engineering. Explain in detail why adding people to a project that is behind schedule will make the situation worse. Explain what underlying issues are involved.

2. CS5391 Survey of Software Engineering. Describe how software testing is handled by each of the following software life cycle models: linear-sequential, incremental, spiral. In your descriptions, indicate how test planning is performed, how often testing occurs, and when testing occurs (with respect to other development phases such as requirements specification, design, implementation, and maintenance).

3. CS5329 Algorithm Design and Analysis

   (a) Recurrences.

      i. Setup a recurrence relation for the worst-case time complexity of the following algorithm. This algorithm outputs all permutations of LIST[K..M]. Assume the original call is PERM(LIST, 0,N-1). Also assume that the function SWAP(X,Y) runs in constant time.

```
PERM(list,k,m)
  if (k==m) then
    for i = 0 to  m write(list[i]);
  else
    SWAP(list[k],list[i]);
    PERM(list, k+1,m);
    SWAP(list[k],list[i]);
```

      ii. Can the recurrence relation from part 2(a)i be solved using the Master Method?

      iii. Show the steps in solving the following recurrence:

$$T(n) = \begin{cases} 1 & n = 1 \\ 4T(n/2) + n^3 & n > 1 \end{cases}$$

   (b) Design Techniques. Discuss the Greedy Strategy for algorithm design. A complete answer will include a comparison with Divide and Conquer and Dynamic Programming strategies, a characterization of problems that can be solved efficiently using dynamic programming, a discussion of the steps in algorithm development and an example.

4. CS5346 Advanced Aritficial Intelligence

(a) Consider the following Wumpus World

| | | | |
|---|---|---|---|
| Stench<br>Breeze | **WUMPUS** | Stench<br>Glitter<br>**GOLD** | Breeze |
| **PIT #2** | Stench<br>Breeze | Breeze | **PIT #1** |
| Breeze | | | Breeze |
| **START** | | | |

Write the facts and rules to determine the positions of the Wumpus and the PIT #1. Show all steps of the derivation in determining the positions of the Wumpus and the PIT #1.

(b) A database of a county banks loan department is represented as positive and negative instances in the following table:

| Individual | RATING | BAL | APP | INC | OK |
|---|---|---|---|---|---|
| 1 | 1 | 0 | 0 | 1 | 0 |
| 2 | 0 | 0 | 1 | 0 | 0 |
| 3 | 1 | 1 | 0 | 1 | 1 |
| 4 | 0 | 1 | 1 | 1 | 1 |
| 5 | 0 | 1 | 1 | 0 | 0 |
| 6 | 1 | 1 | 1 | 0 | 1 |
| 7 | 1 | 1 | 1 | 1 | 1 |
| 8 | 1 | 0 | 1 | 0 | 0 |
| 9 | 1 | 1 | 0 | 0 | 0 |

Derive a set of rules that covers all and only the positive instances.

**Department of Computer Science**
**Graduate Comprehensive Exam**
**in**
**Computer Science**
**Spring 2006**

- **Answer the questions on the paper supplied.**
- **Answer question 1. Answer question 2 or question 3. Answer question 4. You should answer a total of three questions.**
- **Start each question on a new page. Write on only one side of the paper.**
- **Write your SIX-DIGIT Texas State ID in the top right corner of each page of your answer. Do NOT put your name anywhere on the answers.**
- **Put the number of the question being answered in the top left corner of each answer page.**
- **If the answer to a question is written on more than one page, number the pages consecutively.**

1. CS5391 Survey of Software Engineering.

   (a) Explain the differences between functional and non-functional requirements. Describe how testing methodologies differ for functional and non-functional requirements.

   (b) Describe the following software process models and compare and contrast them being specific concerning the type of development they are best suited for and how the choice of model may effect the time and cost of development.

      i. waterfall model
      ii. incremental model
      iii. spiral model

2. CS5318 Design of Programming Languages.

   (a) Discuss two common computational paradigms. A complete answer will include computational models, data and control structures, and an example language.

   (b) Discuss in detail and with examples the roles of context-free and attribute grammars in programming language design.

3. CS5338 Formal Languages.

   Answer all the parts of this question

   (a) Prove or disprove the following statement:
   "The language $L = \{w \in \{a, b\}^* : \ w = (ab)^m, m \geq 0\}$, is a regular language"

   (b) Prove or disprove the following statement:
   "The language $L = \{w \in \{a, b\}^* : w = a^2 b^m a^m, m \geq 0\}$, is not a regular language"

   (c) Show (no need for a formal proof) that the following statement is true:
   "The language $L = \{w \in \{a, b, c\}^*, \text{ and } w_1 \in \{a, b\}^* : w = w_1 c w_1\}$,
   is not a context free language"

   (d) Show a context-free grammar and an accepting PDA for the following language:
   $L = \{w \in \{a, b, c\}^* : w = a^n b^{n+m} c^m; n, m \geq 0\}$

   (e) Prove by induction that every string that belongs to the language:
   $L = \{a^n c^m b^n : n \geq 0, m \geq 1\}$,
   is produced by the CFL with production rules:
   $\{S \rightarrow aSb|T; T \rightarrow cT|c\}$.

4. CS5346 Advanced Artificial Intelligence.

   (a) Trace the MIN MAX algorithm (shown in Fig.1) on the example shown in Fig. 2. MAX makes the first move at the root node A. Evaluation function values are computed from MAX's point of view.

   (b) Explain in detail with an example how Alpha-Beta Pruning can enhance the efficiency of search used in the algorithms of two player games

**Algorithm: MINIMAX(Position, Depth, Player)**

1. If DEEP-ENOUGH(*Position*, *Depth*), then return the structure

   VALUE = STATIC(*Position*, *Player*);
   PATH = nil

   This indicates that there is no path from this node and that its value is that determined by the static evaluation function.

2. Otherwise, generate one more ply of the tree by calling the function MOVE-GEN(Position, Player) and setting SUCCESSORS to the list it returns.

3. If SUCCESSORS is empty, then there are no moves to be made, so return the same structure that would have been returned if DEEP-ENOUGH had returned true.

4. If SUCCESSORS is not empty, then examine each element in turn and keep track of the best one. This is done as follows.

   Initialize BEST-SCORE to the minimum value that STATIC can return. It will be updated to reflect the best score that can be achieved by an element of SUCCESSORS.

   For each element SUCC of SUCCESSORS, do the following:

   (a) Set RESULT-SUCC to
   MINIMAX(SUCC, *Depth* + 1, OPPOSITE(*Player*))[3]
   This recursive call to MINIMAX will actually carry out the exploration of SUCC.

   (b) Set NEW-VALUE to – VALUE(RESULT-SUCC). This will cause it to reflect the merits of the position from the opposite perspective from that of the next lower level.

   (c) If NEW-VALUE > BEST-SCORE, then we have found a successor that is better than any that have been examined so far. Record this by doing the following:

      i. Set BEST-SCORE to NEW-VALUE.

      ii. The best known path is now from CURRENT to SUCC and then on to the appropriate path down from SUCC as determined by the recursive call to MINIMAX. So set BEST-PATH to the result of attaching SUCC to the front of PATH(RESULT-SUCC).

5. Now that all the successors have been examined, we know the value of Position as well as which path to take from it. So return the structure

   VALUE = BEST-SCORE
   PATH = BEST-PATH

Fig 1

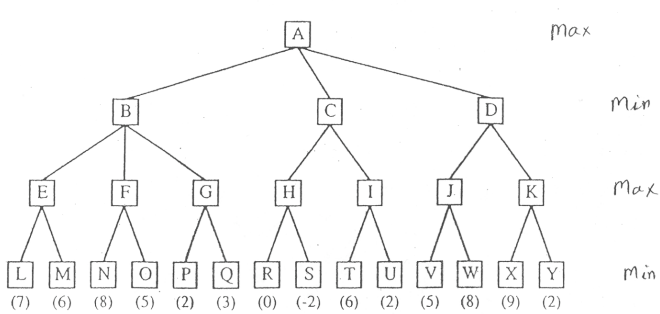



Fig.2

**Department of Computer Science**
**Graduate Comprehensive Exam**
**in**
**Computer Science**
**Spring 2007**

- **Answer the questions on the paper supplied.**
- **Answer question 1 or question 2. Answer question 3. Answer question 4 or question 5 or question 6 or question 7. You should answer a total of three questions.**
- **Start each question on a new page. Write on only one side of the paper.**
- **Write your SIX-DIGIT Texas State ID in the top right corner of each page of your answer. Do NOT put your name anywhere on the answers.**
- **Put the semester and year of the exam in the appropriate blank at the top of each answer page.**
- **Put the number of the question being answered in the appropriate blank on each answer page.**
- **If the answer to a question is written on more than one page, number the pages consecutively.**

1. CS5338 Formal Languages

   Answer all the parts of this question

   (a) Prove or disprove the following statement: "The language

   $$L = \{w \in \{a,b,c\}^* : \text{ every c has an a immediately to its left and a b immediately to its right}\}$$

   is a regular language"

   (b) Prove or disprove the following statement: "The language

   $$L = \{w \in \{a,b\}^*;\ w = a^n b^j :\ |n-j| = 2\},$$

   is not a regular language"

   (c) Show, using a "sketch" of the pumping lemma, that the following statement is true: "The language

   $$L = \{w : w = yxx^R y \text{ and } x, y \in \{a,b\}^*\}$$

   is not a context free language"

   (d) Show a context-free grammar and an accepting PDA for the following language:

   $$L = \{a^n b^m c^k : n = m + k \text{ or } m = n + k\}$$

   (e) Prove by induction that every string that belongs to the language:

   $$L = \{a^n c^{m+1} b^m : n \geq 0, m \geq 1\},$$

   is produced by the CFL with production rules: {
   $S \rightarrow aS|T;$
   $T \rightarrow cTb|c$
   }

2. CS5318 Programming Language Design

   (a) Discuss in detail and with examples the roles of context-free and attribute grammars in programming language design.

   (b) Discuss the significance of abstraction in programming languages.

3. CS5391 Survey of Software Engineering

   Describe the Capability Maturity Model (CMM) including the levels and the key process areas which much be completed for each level to be attained. What organization developed the CMM? How should CMM be used by a software development organization?

4. CS5306 Advanced Operating Systems

   Suppose that four processes initially have the following local clock values:

| P1 | P2 | P3 | P4 |
|---|---|---|---|
| 8 | 17 | 2 | 33 |

   These values are to be advanced by the minimum necessary amount 1 for each new event (including the first). The only events which each process counts are the sending and receiving of messages, each of which is one distinct event.

   Suppose the following sequence of messages occurs, and that each message is sent only after the previous one is received. Give the time stamp for each message.

| **MESSAGE** | **TIME STAMP** |
|---|---|
| P2 sends a message to P4 | _____ |
| P4 sends a message to P2 | _____ |
| P2 sends a message to P1 | _____ |
| P1 sends a message to P3 | _____ |
| P3 sends a message to P2 | _____ |

5. CS5310 Network and Communication Systems

   What is modulation? Why modulations are needed? What are popular modulation techinques? Please provide three examples of real applications where modulations are essential. Please provide necessary details to support your argument.

6. CS5332 Database Theory and Design

   Design an E/R diagram with normalized tables (entities), attributes, PKs, FKs and relationships for the following real world problem. (All base tables/relations must be in 3NF or BCNF.)

   Read the following case study, which describes the data requirements for a video rental company. The video rental company has several branches throughout the USA. The data held on each branch is the branch address made up of a street, city, state, and zip code, and the telephone number. Each branch is given a branch number, which is unique throughout the company. Each branch is allocated staff, which includes a Manager. The Manager is responsible for the day-to-day running of a given branch. The data held on a member of staff is his or her name, position, and salary. Each member of a staff is given a staff number, which is unique throughout the company. Each branch has a stock of videos. The data held on a video is the catalog number, video number, title, category, daily rental, cost, status, and the names of the main actors and the director. The catalog number uniquely identifies each video. However, in most cases, there are several copies of each video at a branch, and the individual copies are identified using the video number. A video is given a category such as Action, Adult, Children, Drama, Horror, or Sci-Fi. The status Indicates whether a specific copy of a video is available for rent. Before hiring a video from the company, a customer must first register as a member of a local branch. The data held on a member is the first and last name, address, and the date that the member registered at a branch. Each member is given a member number, which is unique throughout the company. Once registered, a member is free to rent videos, up to a maximum of ten at any one time. The data held on each video rented is the rental number, the full name and number of the member, the video number, title, and daily rental, and the dates the video is rented out and returned. The rental number is unique throughout the company.

7. CS5332 Database Theory and Design

*(20 marks total)* Consider the following typed relational schema describing books and borrowers in a library:

Book(<u>isbn:integer</u>, title:string, publisher:string, year:integer)
Author(<u>isbn:integer</u>, <u>name:string</u>, rank:integer)
Borrower(<u>bid:integer</u>, name:string, address:string)
Borrowings(<u>isbn:integer</u>, <u>borrower:integer</u>, whenTaken:date, whenReturned:date)

Assume the following:

- the fields that are underlined are the primary keys of the relations
- there is a referential integrity constraint between "borrower" in relation Borrowings and "bid" in relation Borrower
- there are also referential integrity constraints between "isbn" in relation Book and "isbn" in relations Author and Borrowings
- dates are represented as a count of the number of days since Jan 1 2004
- there is a system function `today` that returns today's `date`
- the relational operators $<$ and $>$ allow you test *before* and *after* on `date`s
- all loans are for a period of 14 days
- the `whenReturned` field is `null` until the book is returned
- the `rank` field specifies whether the author is the first, second, ... author

Implement solutions to the following:

(a) *(4 marks)* Write an SQL query to show all the books (`title` only) written by a particular author called "Ullman".

(b) *(4 marks)* Write an SQL query to show which books (`title` only) have multiple authors.

(c) *(4 marks)* Write an SQL query to show which books (`title` only) have never been borrowed.

(d) *(4 marks)* Write an SQL query to show which books (`title` only) are currently out on loan.

(e) *(4 marks)* Write an SQL query to list overdue books (`title`) and the borrower's name (Hints: you need to join relations Book, Borrowers and Borrowings).

**Next:** About this document

**Department of Computer Science**
**Graduate Comprehensive Exam**
**Summer 1997**

- **Answer the questions on the paper supplied. Be sure to answer question 1, question 2 *or* 3, *and* question 4 *or* 5 *or* 6, depending upon how you have registered.**
- **Start each question on a new page.**
- **Do NOT write on the back of the paper.**
- **Write your SSN in the top right corner of each page of your answer. Do NOT put your name anywhere on the answers.**
- **Put the number of the question being answered in the top left corner of each page of your answer.**
- **If the answer to a question is written on more than one page, number the pages consecutively.**

1. (CS5329) Design an algorithm to determine whether there are two elements of $S$ whose sum is exactly $x$. The algorithm should run in time tex2html_ . The input is a set $S$ containing $n$ real numbers, and a real number $x$. Prove the correctness and the complexity of the algorithm.

   Answer **exactly one** of the following (CS5338 *or* CS5318):
2. (CS5338) Consider the regular, context-free, recursive, and recursively enumerable languages.

   1. Define each language and identify its associated machine model.
   2. These languages can be nested; show the nesting.
   3. For each nesting level, give an example of a language that belongs to the outer level but not to the one immediately inside.
   4. Tabulate their closure properties under operations of union, intersection, complementation, concatenation, and Kleene star.
   5. Define a grammar for each of tex , tex2h and tex2htm .

3. (CS5318) Explain in detail and with examples the following:

   1. Programming language design criteria
   2. Computational paradigms
   3. Abstractions in programming languages

   Answer **exactly one** of the following (CS5306 *or* CS5310 *or* CS5332):
4. (CS5306) Explain the principles of virtual memory systems. Give the advantages of virtual memeory systems and explain why many architectures have multilevel page tables (directories.)

Explain how paging is carried out in at least two architectures, such as Intel x86, Motorola 68x, etc. Which parts of the process are typically carried out in hardware and which are performed by the OS?

5. (CS5310) Describe the essential elements required for a client and a server to perform a session of TCP/IP based communication. For both connectionless and connection-oriented communications, describe how the client and the server will obtain those required essential elements.
6. (CS5332) Discuss the significance of the theory of functional dependencies. Discuss the relationships between 3NF and BCNF. Why are 3NF and BCNF preferred over 2NF?

---


- About this document ...

**Next:** About this document


*Carol Hazlewood*
*Mon Nov 23 12:18:09 CST 1998*

**Department of Computer Science**
**Graduate Comprehensive Exam**
**Summer 1998**

- **Answer the questions on the paper supplied.**
- **Answer question 1. Answer question 2. Answer *one of* questions 3 *or* 4, *or* 5, depending upon how you have registered.**
- **Start each question on a new page.**
- **Write on only one side of the paper.**
- **Write your SSN in the top right corner of each page of your answer. Do NOT put your name anywhere on the answers.**
- **Put the number of the question being answered in the top left corner of each page of your answer.**
- **If the answer to a question is written on more than one page, number the pages consecutively.**

1. CS5329 Algorithm Design and Analysis

   State the Sorting Problem.

   For three of heapsort, quicksort, merge sort, insertion sort, counting sort, radix sort, and bucket sort, discuss average-case time complexity, worst-case time complexity, space complexity, stability, and characteristics of problems that make the algorithm a good or bad choice.

   Show the steps in sorting the array 4,3,5,1,2 using two of the sorting algorithms listed above.

   Explain how heapsort can be optimal but slower than counting sort.

2. CS5391 Survey of Software Engineering

   There are five levels to the capability maturity model (CMM). Discuss each of these levels, including the key practices that are in place at each level and which practices are needed to advance to the next level.

3. CS5306 Advanced Operating Systems

   Using the traditional Unix design as a model, explain the problem in upgrading from a single cpu machine to a multiprocessor machine; e.g. if we try to use the same unix kernel, what are the problems that may occur in running kernel code. Be sure to discuss: base level code, preemptablility of code, interrupt handlers, short term and long term mutual exclusion, etc. as you explain how traditional Unix handles race conditions.

Explain what a Master/Slave solution to the problem above is. What minimal coding must be added to the kernel? What are the advantages and disadvanages of this solution? What are other simple solutions?

4. 5310 Network and Communication Systems

   Describe the Fourier series. Explain in detail its significance with respect to data communications.

5. CS5332 Database Theory and Design

   Discuss the relationships between relational algebra and SQL. In each case, please list their essential features first. Give an example to demonstrate the similarities and differences between these two query languages.

---

- About this document ...

*Carol Hazlewood*
*Mon Nov 23 14:39:09 CST 1998*

**Next:** About this document ...

# Department of Computer Science
# Graduate Comprehensive Exam
# Summer 1999

- Answer the questions on the paper supplied.
- Answer question 1. Answer *one of* questions 2 *or* 3, depending on how you registered. Answer *one of* questions 4 *or* 5, *or* 6 depending on how you registered.
- Start each question on a new page.
- Write on **only one side** of the paper.
- Write your SSN in the top right corner of each page of your answer. Do **NOT** put your name anywhere on the answers.
- Put the number of the question being answered in the top left corner of each page of your answer.
- If the answer to a question is written on more than one page, number the pages consecutively.

1.

CS5346 Advanced Artificial Intelligence

(a)
Explain in detail with examples, at least two informed search techniques.

(b)
Explain the use of Forward and Backward Chaining in Expert Systems in detail with examples.

(c)
Explain in detail with examples admissibility, optimality, and complexity of the $A^*$ algorithm.

2.

CS5318 Design of Programming Languages

(a)

Discuss in detail and with examples the role of abstraction in programming languages.

(b)

Discuss in detail and with examples the use of context-free grammars to specify operator precedence and associativity.

3.

CS5338 Formal Languages

(a)

Give a precise definition of a finite-state automaton, a pushdown automaton, and a Turing machine.

(b)

Construct a finite-state automaton, a pushdown automaton, *and* a Turing Machine to recognize the language $\{a^n \mid n >= 0\}$. Discuss how your machines work.

4.

CS5306 Advanced Operating Systems

Explain the principles of virtual memory systems. Give the advantages of virtual memory systems and explain why many architectures have multilevel page tables (directories). Explain how paging is carried out in at least two architectures, such as Intel x86, Motorola 68x, etc. Which parts of the process are typically carried out in hardware and which are performed by the OS?

5.

CS5310 Network and Communication Systems

Discuss the seven layers of the OSI reference model.

6.

CS5332 Data Base Theory and Design

Discuss the significance of the theory of functional dependencies. Discuss the relationships between 3NF and BCNF. Why are 3NF and BCNF are preferred over 2NF?

---

- About this document ...

---

**Next:** About this document ...

*Carol Hazlewood*
*2000-06-15*

# Department of Computer Science
# Graduate Comprehensive Exam
# in
# Computer Science
# Summer 2000

- **Answer the questions on the paper supplied.**
- **Answer question 1. Answer *either* question 2 or question 3. Answer *either* question 4 *or* question 5, *or* question 6.**
- **Start each question on a new page.**
- **Write on only one side of the paper.**
- **Write your SSN in the top right corner of each page of your answer. Do NOT put your name anywhere on the answers.**
- **Put the number of the question being answered in the top left corner of each page of your answer.**
- **If the answer to a question is written on more than one page, number the pages consecutively.**

1. CS5329 Algorithm Design and Analysis
   (a) Set up and solve a recurrence for the time complexity of MergeSort.
   (b) Solve the following recurrence, where $a \geq 1$ is a constant:
   $$\begin{aligned} T(a) &= 1 \\ T(n) &= T(n-a) + T(a) + n \end{aligned}$$
   (c) Design Techniques.
   Discuss dynamic programming and greedy techniques. Describe each technique, list characteristics of problems that make them amenable to solution by each technique, and show how the techniques are used in examples of problems that can be solved by each technique.

2. CS5318 Design of Programming Languages
   (a) One of the criteria used to evaluate a programming language is its cost. Discuss in detail and with examples three costs associated with programming languages.
   (b) discuss in detail and with examples the role of abstraction in programming language design.

3. CS5338 Formal Languages
   (a) Give a precise definition of a finite-state automaton, a pushdown automaton, and a Turing machine.
   (b) Construct a finite-state automaton, a pushdown automaton, and a Turing Machine to recognize the language $L = \{a^*b^*\}$. Discuss how your machines work.

4. CS5306 Advanced Operating Systems
   Explain the principles of virtual memory systems. Give the advantages of virtual memory systems and explain why many architectures have multilevel page tables (directories). Explain how paging is carried out in at least two architectures, such as Intel x86, Motorola 68x, etc. Which parts of the process are typically carried out in hardware and which are performed by the OS?

5. CS5310 Network and Communication Systems
   Describe the Fourier series. Explain in detail its significance with respect to data communications.

6. CS5332 Data Base Theory and Design
   (a) Discuss the significance of theory of functional dependencies.
   (b) Discuss with examples the relationships between 3NF and BCNF. Why are 3NF and BCNF are preferred over 2NF?

# Department of Computer Science
# Graduate Comprehensive Exam
# in
# Computer Science
# Summer 2001

- **Answer the questions on the paper supplied.**
- **Answer question 1. Answer *either* question 2 *or* question 3. Answer *either* question 4 *or* question 5.**
- **Start each question on a new page.**
- **Write on only one side of the paper.**
- **Write your SSN in the top right corner of each page of your answer. Do NOT put your name anywhere on the answers.**
- **Put the number of the question being answered in the top left corner of each page of your answer.**
- **If the answer to a question is written on more than one page, number the pages consecutively.**

1. CS5346 Advanced Artificial Intelligence

   (a) Consider the following sentences:
   - Steve only likes easy courses.
   - Science courses are hard.
   - All the courses in the theater department are easy.
   - TH301 is a theater course.
   - Kate likes all courses Steve likes.

   i. Translate these sentences into formulas in predicate logic.
   ii. Convert the formulas into clause form.
   iii. Use resolution to answer the question: "What course does kate like?"
   iv. Prove that Steve likes TH301 using backward chaining.

   (b) Describe the Minimax search using an example. Describe how alpha-beta pruning improves the minimax search using an example.

2. CS5318 Design of Programming Languages

   (a) One of the criteria used to evaluate a programming language is its cost. Discuss in detail and with examples three costs associated with programming languages.

   (b) Discuss in detail and with examples the use of context-free grammars to specify operator precedence and associativity.

3. CS5338 Formal Languages

   (a) Give a precise definition of a Turing Machine. Discuss how a Turing machine works.

   (b) Construct a minimal-state finite-state automaton to recognize the langaage $L = \{a^*ba^*\}$. Discuss why $L$ can be recognized by a finite-state automaton but the language $K = \{a^nba^n\}$ cannot.

4. CS5310 Network and Communication Systems

   Briefly describe the seven layer ISO's OSI model (For each layer briefly describe its services and design issues). Explain why the top four layers are called "end-to-end" layers. You should support your explanation with an application example.

5. CS5332 Database Theory and Design

   Discuss the relationships between relational algebra and SQL. In each case, please list their essential features first. Give an example to demonstrate the similarities and differences between these two query languages.

**Department of Computer Science**
**Graduate Comprehensive Exam**
**in**
**Computer Science**
**Summer 2002**

- **Answer the questions on the paper supplied.**
- **Answer question 1. Answer question 2. Answer question 3.**
- **Start each question on a new page.**
- **Write on only one side of the paper.**
- **Write your SSN in the top right corner of each page of your answer. Do NOT put your name anywhere on the answers.**
- **Put the number of the question being answered in the top left corner of each page of your answer.**
- **If the answer to a question is written on more than one page, number the pages consecutively.**

1. CS5329 Algorithm Design and Analysis

   (a) Discuss in detail and with examples the divide-and-conquer algorithm design technique. A complete answer will include a discussion of characteristics of problems for which a divide-and-conquer strategy yields an efficient solution and a comparison with other design techniques.

   (b) Set up a recurrence relation for the time complexity of the following algorithm. Show the steps in solving the recurrence. Assume the original call is
   search(X, 1, n, a).

   // given an array X, indexed from $\ell$ to r, of elements that can be compared, and a key $a$,
   // search(X,$\ell$,r,a) returns the location of the first occurrence of key $a$ in X[$\ell$..r],
   //if $a$ is in the array, and 0 otherwise.

   ```
   search(X, ℓ, r, a)
      if (ℓ > r) return 0
      if (a == X[ℓ])
         then return ℓ;
         else return search(X, ℓ+1, r,a);
   ```

2. CS5391 Survey of Software Engineering

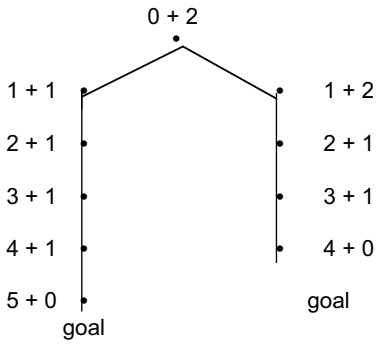
   There are five levels to the capability maturity model (CMM). Discuss each of these levels, including the key practices that are in place at each level and which practices are needed to advance to the next level.

3. CS5346 Advanced Artificial Intelligence

   (a) Explain "admissible heuristic". Prove the optimality of A* algorithm. Also, explain the complexity of A* algorithm.

   (b) Verify the soundness of the resolution rule by constructing a truth table.

   (c) Trace the IDA* algorithm for the following tree:




IDA*

1. Set c = 1; this is the current depth cutoff
2. Set L to be a list of the initial nodes in the problem.
3. Let n be the first node on L. If L is empty, increment c and return to step 2
4. If n is a goal node, stop and return it and the path from the initial node to n.
5. Otherwise, remove n from L. Add to the front of L every child n' of n for which f(n')$\leq$c. Return to step 3.

**Department of Computer Science**
**Graduate Comprehensive Exam**
**in**
**Computer Science**
**Summer 2003**

**Directions:**

- **Answer the questions on the paper supplied.**
- **Answer question 1. Answer question 2. Answer question 3.**
- **Start each question on a new page.**
- **Write on *only one side* of the paper.**
- **Write your SSN in the top right corner of each page of your answer. Do NOT put your name anywhere on the answers.**
- **Put the number of the question being answered in the top left corner of each page of your answer.**
- **If the answer to a question is written on more than one page, number the pages consecutively.**

1. **Comprehensive problem for CS5329 Algorithm Design and Analysis**

A. Discuss in detail and with examples the greedy algorithm design strategy. A complete answer will include a discussion of the characteristics of problems for which the greedy strategy yields correct and efficient solutions and a comparison with other design techniques.

B. Set up a recurrence relation for the time complexity of the following algorithm. Show the steps in solving the recurrence. Assume the original call is

search(X,1,n,a).

```
// given an array X,  indexed from p to r,  of elements that can be compared,
// and a key a, search(X, p, r, a) returns the location of the last occurrence
// of key a in array X[p..r], if a is in the array, and 0 otherwise.

int search (X, p, r, a)

    if (p > r ) return 0;
    if (a == X[r]) then
        return r;
    else
        return search (X, p, r-1, a);
```

**2. Comprehensive problem for CS 5391 Survey of Software Engineering**

Requirements engineering (elicitation, analysis, specification) occurs at the beginning of every software life cycle model (to varying degrees). Describe how requirements engineering is performed within these life cycle models - linear sequential, incremental, and spiral. Include in your description when it occurs, how often, and how a complete requirements specification is obtained. Also, address how well each of the above life cycle models accommodates mid-development requirements change.

**3. Comprehensive problem for CS 5346 Artificial Intelligence**

Consider the following facts:

The members of the Elm St. Bridge Club are Joe, Sally, Bill, and Ellen.
Joe is married to Sally.
Bill is Ellen's brother.
The spouse of every married person in the club is also in the club.
The last meeting of the club was at Joe's house.

A) Represent these facts in predicate calculus.

B) Convert the formulas into clause form.

C) Using resolution, can you prove that "the last meeting of the club was at Sally's house". If you cannot, explain why the proof is not possible and indicate which additional facts are necessary to complete the proof.

D) Using resolution, prove that "Ellen is not married". If you cannot, explain why the proof is not possible and indicate which additional facts are necessary to complete the proof.

**Department of Computer Science**
**Graduate Comprehensive Exam**
**in**
**Computer Science**
**Summer 2004**

- **Answer the questions on the paper supplied.**
- **Answer question 1. Answer question 2. Answer question 3 .**
- **Start each question on a new page.**
- **Write on only one side of the paper.**
- **Write your Texas State ID in the top right corner of each page of your answer. Do NOT put your name anywhere on the answers.**
- **Put the number of the question being answered in the top left corner of each answer page.**
- **If the answer to a question is written on more than one page, number the pages consecutively.**

1. CS5329 Algorithm Design and Analysis

   (a) Recurrences.

   i. Setup a recurrence relation for the worst-case time complexity of the following algorithm. This algorithm solves the Towers of Hanoi problem. The call TOWER(N,X,Y,Z) moves $n$ disks of graduated sizes from tower $x$ to tower $y$ using tower $z$ for intermediate storage, where no disk is placed on top of a smaller one. Assume the original call is TOWER(N,1,2,3). Also assume that the function MOVE-ONE(X,Y) runs in constant time and moves one disk from the top of tower x to the top of tower y.

```
TOWER(n,x,y,z)
  if (n > 0) then
    TOWER(n-1,x,z,y);
    MOVE-ONE(x,y);
    TOWER(n-1,z,y,x)
```

   ii. Can the recurrence relation from part 1(a)i be solved using the Master Method?

   iii. Show the steps in solving the following recurrence:

$$T(n) = \begin{cases} 1 & n = 1 \\ 3T(n/3) + n^2 & n > 1 \end{cases}$$

   (b) Design Techniques. Discuss the algorithm design technique of Dynamic Programming. A complete answer will include a comparison with Divide and Conquer and Greedy strategies, a characterization of problems that can be solved efficiently using dynamic programming, a discussion of the steps in algorithm development and an example.

2. CS5391 Survey of Software Engineering

   Describe how software testing is handled by each of the following software life cycle models: linear-sequential, incremental, spiral. In your descriptions, indicate how test planning is performed, how often testing occurs, and when testing occurs (with respect to other development phases such as requirements specification, design, implementation, and maintenance).

3. CS5346 Advanced Artificial Intelligence

(a) By generating truth tables, determine for each of the following sentences if it is valid, unsatisfiable or satisfiable:

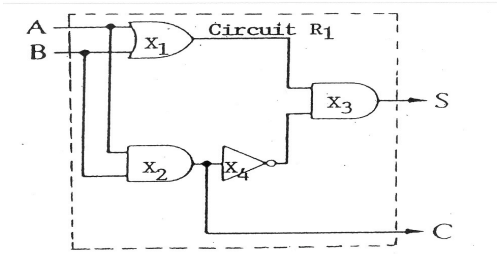
$$(\text{Smoke} \Rightarrow \text{Fire}) \Rightarrow (\neg\, \text{Smoke} \Rightarrow \neg\, \text{Fire})$$
$$((\text{Smoke} \wedge \text{Heat}) \Rightarrow \text{Fire}) \Leftrightarrow ((\text{Smoke} \Rightarrow \text{Fire}) \vee (\text{Heat} \Rightarrow \text{Fire}))$$

(b) The following general rules are followed in digital circuits. Rules 4-6 cover either ON or OFF states of the gates. Add three more rules to cover both ON and OFF states.

1. If two terminals are connected, then they have the same signal:
   $\forall\, t_1, t_2\ Connected(t_1, t_2) \Rightarrow Signal(t_1) = Signal(t_2)$
2. The signal at every terminal is either on or off (but not both):
   $\forall\, t\ Signal(t) = On - Signal(t) = Off$
   $On \neq Off$
3. Connected is a commutative predicate:
   $\forall\, t_1, t_2\ Connected(t_1, t_2) \Leftrightarrow Connected(t_2, t1)$
4. An OR gate's output is on if and only if any of its inputs are on:
   $\forall\, g\ Type(g) = OR \Rightarrow$
   $Signal(Out(1, g)) = On \Leftrightarrow \exists n\ Signal\ (In(n, g)) = On$
5. An AND gate's output is off if and only if any of its inputs are off:
   $\forall\, g\ Type(g) = AND \Rightarrow$
   $Signal(Out(1, g)) = Off \Leftrightarrow \exists n\ Signal\ (In(n, g)) = Off$
6. An XOR gate's output is on if and only if its inputs are different:
   $\forall\, g\ Type(g) = XOR \Rightarrow$
   $Signal(Out(1, g)) = On \Leftrightarrow Signal\ (In(1, g)) \neq Signal\ (In(2, g))$
7. A NOT gate's output is different from its input:
   $\forall\, g\ (Type(g) = NOT) \Rightarrow Signal\ (Out(1, g)) \neq Signal\ (In(1, g))$

The following is the description of the gate types of the circuit R1 given below:

| | |
|---|---|
| Type (X1) = OR | Type (X2) = AND |
| Type (X3) = AND | Type (X4) = NOT |




Represent the connections of the circuit and determine what combinations of Sig (In(A, R1)), Sig (In(B, R1)) would cause the following output
(OUT(S, R1) = ON) $\wedge$ (OUT(C, R1) = OFF)

**Department of Computer Science**
**Graduate Comprehensive Exam**
**in**
**Computer Science**
**Spring 2004**

- **Answer the questions on the paper supplied.**
- **Answer question 1. Answer** *either* **question 2** *or* **question 3. Answer** *either* **question 4** *or* **question 5** *or* **question 6** *or* **question 7. You should answer a total of three questions.**
- **Start each question on a new page.**
- **Write on only one side of the paper.**
- **Write your Texas State ID in the top right corner of each page of your answer. Do NOT put your name anywhere on the answers.**
- **Put the number of the question being answered in the top left corner of each answer page.**
- **If the answer to a question is written on more than one page, number the pages consecutively.**

1. CS5346 Advanced Artificial Intelligence

   (a) Prove that if the heuristic function $h$ obeys the triangle inequality then the values of $f(n)$ along any path are non-decreasing.

   (b) Explain, with examples, the differences in the perfect decisions and imperfect decisions methodologies employed in two person games.

   (c) It is known that Chris is the son of Nancy. Prove that Chris is not the daughter of Mike. You will need to write at least two implications and represent the given facts in predicate logic. Use refutation principle in your proof.

2. CS5318 Design of Programming Languages

   (a) Regularity is one of the criteria used to evaluate programming languages. Regularity is often subdivided into areas of generality, uniformity and orthogonality. Discuss in detail and with examples these three sub-criteria.

   (b) Discuss attribute grammars. A complete answer will include a comparison of the power or expressivity of attribute grammars and context-free grammars, an example of an attribute grammar, and an example of a decorated parse tree.

3. CS5338 Formal Languages

   (a) Show that the difference of two regular languages is regular.

   (b) Construct a deterministic finite-state automaton and a Turing Machine to recognize the language $a^*b^*$.

4. CS5306 Advanced Operating Systems

   Describe three mutual exclusion methods in distributed systems, point out their advantages and disadvantages, and compare the number of messages communicated among processors.

5. CS5310 Advanced Networks

   Describe the essential elements required for a client and a server to perform a session of TCP/IP based communication. For both connectionless and connection-oriented communications, describe how the client and the server will obtain those required essential elements.

6. CS5332 Advanced Database Systems

   Consider the following relational schema :

   Suppliers(sid, sname, address)

   Parts(pid, pname, color)

   Catalog(sid, pid, cost)

   Assume that the ids are integers, that cost is a real number, that all other attributes are strings, that the sid field in Catalog relation is a foreign key containing a supplier id, and that the pid in Catalog relation field is a foreign key containing a part id.

   Write SQL statements to answer each of the following queries:

   (a) Find the names of suppliers who supply some red part.
   (b) Find the sids of suppliers who supply some red or green part.
   (c) Find the sids of suppliers who supply some red part and some green part.
   (d) Find the sids of suppliers who supply every part.
   (e) Find pairs of sids such that the supplier with the first sid charges more for some part than the supplier with the second sid.
   (f) Find the pids of parts that are supplied by at least two different suppliers.

7. CS5332 Advanced Database Systems

   Given a university environment, develop an E/R diagram (model) for the following possible relations: Student, Professor, Employee, Department, Course (in catalog; no class index), Class (in schedule; with class index number), and Classroom.

   Notes: You need to create appropriate attributes, primary keys, and optional foreign keys for each relation. You may also need to create additional relations and must normalize all relations whenever needed. Finally, specify all required relationships among entities.

**Department of Computer Science**
**Graduate Comprehensive Exam**
**in**
**Computer Science**
**Summer 2004**

- **Answer the questions on the paper supplied.**
- **Answer question 1. Answer question 2. Answer question 3 .**
- **Start each question on a new page.**
- **Write on only one side of the paper.**
- **Write your Texas State ID in the top right corner of each page of your answer. Do NOT put your name anywhere on the answers.**
- **Put the number of the question being answered in the top left corner of each answer page.**
- **If the answer to a question is written on more than one page, number the pages consecutively.**

1. CS5329 Algorithm Design and Analysis

   (a) Recurrences.

   i. Setup a recurrence relation for the worst-case time complexity of the following algorithm. This algorithm solves the Towers of Hanoi problem. The call TOWER(N,X,Y,Z) moves $n$ disks of graduated sizes from tower $x$ to tower $y$ using tower $z$ for intermediate storage, where no disk is placed on top of a smaller one. Assume the original call is TOWER(N,1,2,3). Also assume that the function MOVE-ONE(X,Y) runs in constant time and moves one disk from the top of tower x to the top of tower y.

```
TOWER(n,x,y,z)
  if (n > 0) then
    TOWER(n-1,x,z,y);
    MOVE-ONE(x,y);
    TOWER(n-1,z,y,x)
```

   ii. Can the recurrence relation from part 1(a)i be solved using the Master Method?

   iii. Show the steps in solving the following recurrence:

$$T(n) = \begin{cases} 1 & n = 1 \\ 3T(n/3) + n^2 & n > 1 \end{cases}$$

   (b) Design Techniques. Discuss the algorithm design technique of Dynamic Programming. A complete answer will include a comparison with Divide and Conquer and Greedy strategies, a characterization of problems that can be solved efficiently using dynamic programming, a discussion of the steps in algorithm development and an example.

2. CS5391 Survey of Software Engineering

   Describe how software testing is handled by each of the following software life cycle models: linear-sequential, incremental, spiral. In your descriptions, indicate how test planning is performed, how often testing occurs, and when testing occurs (with respect to other development phases such as requirements specification, design, implementation, and maintenance).

3. CS5346 Advanced Artificial Intelligence

(a) By generating truth tables, determine for each of the following sentences if it is valid, unsatisfiable or satisfiable:

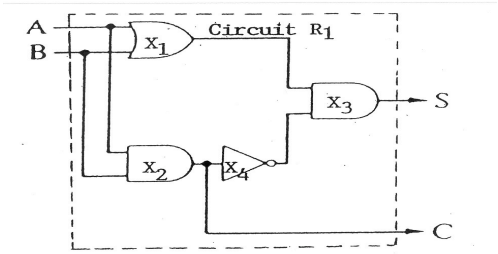
$$(\text{Smoke} \Rightarrow \text{Fire}) \Rightarrow (\neg\, \text{Smoke} \Rightarrow \neg\, \text{Fire})$$
$$((\text{Smoke} \wedge \text{Heat}) \Rightarrow \text{Fire}) \Leftrightarrow ((\text{Smoke} \Rightarrow \text{Fire}) \vee (\text{Heat} \Rightarrow \text{Fire}))$$

(b) The following general rules are followed in digital circuits. Rules 4-6 cover either ON or OFF states of the gates. Add three more rules to cover both ON and OFF states.

1. If two terminals are connected, then they have the same signal:
   $\forall\, t_1, t_2\ Connected(t_1, t_2) \Rightarrow Signal(t_1) = Signal(t_2)$
2. The signal at every terminal is either on or off (but not both):
   $\forall\, t\ Signal(t) = On - Signal(t) = Off$
   $On \neq Off$
3. Connected is a commutative predicate:
   $\forall\, t_1, t_2\ Connected(t_1, t_2) \Leftrightarrow Connected(t_2, t1)$
4. An OR gate's output is on if and only if any of its inputs are on:
   $\forall\, g\ Type(g) = OR \Rightarrow$
   $Signal(Out(1, g)) = On \Leftrightarrow \exists n\ Signal\ (In(n, g)) = On$
5. An AND gate's output is off if and only if any of its inputs are off:
   $\forall\, g\ Type(g) = AND \Rightarrow$
   $Signal(Out(1, g)) = Off \Leftrightarrow \exists n\ Signal\ (In(n, g)) = Off$
6. An XOR gate's output is on if and only if its inputs are different:
   $\forall\, g\ Type(g) = XOR \Rightarrow$
   $Signal(Out(1, g)) = On \Leftrightarrow Signal\ (In(1, g)) \neq Signal\ (In(2, g))$
7. A NOT gate's output is different from its input:
   $\forall\, g\ (Type(g) = NOT) \Rightarrow Signal\ (Out(1, g)) \neq Signal\ (In(1, g))$

The following is the description of the gate types of the circuit R1 given below:

| | |
|---|---|
| Type (X1) = OR | Type (X2) = AND |
| Type (X3) = AND | Type (X4) = NOT |




Represent the connections of the circuit and determine what combinations of Sig (In(A, R1)), Sig (In(B, R1)) would cause the following output
(OUT(S, R1) = ON) $\wedge$ (OUT(C, R1) = OFF)

**Department of Computer Science**
**Graduate Comprehensive Exam**
**in**
**Computer Science**
**Fall 2004**

- **Answer the questions on the paper supplied.**
- **Answer** *one of* **question 1 or question 2.. Answer** *one of* **question 3 or question 4. Answer** *one of* **question question 5, question 6, question 7, or question 8. You should answer a total of three questions.**
- **Start each question on a new page. Write on only one side of the paper.**
- **Write your Texas State ID in the top right corner of each page of your answer. Do NOT put your name anywhere on the answers.**
- **Put the number of the question being answered in the top left corner of each answer page.**
- **If the answer to a question is written on more than one page, number the pages consecutively.**

1. CS5391 Survey of Software Engineering

   (a) As a manager you ask a software engineer to add features to a software product. The engineer responds that adding these features makes the program incorrect since the specification for these features directly contradicts the specification for this product, and customers will not purchase an incorrect program. You agree with your engineer, however, you want to convince him that the effect of these specification discrepancies is not that serious and they shall appear only in few circumstances. How would you build you argument?

   (b) What percentage of the total cost of a software product goes to the maintenance stage? Explain the difference between corrective, adaptive, and perfective maintenances and show the percentage of the total maintenance cost that goes to each of these maintenances.

   (c) Suppose that your software product design is based on the pipe-and-filter architectural style. Even though the development is going according to a schedule, your manager suggests that a product may be completed faster by adding more software engineers to parallelize the development of this product. Do you agree with this suggestion? State why.

2. CS5391 Survey of Software Engineering

   Define cyclomatic complexity for a flow graph. How is cyclomatic complexity used in deriving test cases for source code?

   Generate the flow graph and compute the cyclomatic complexity for the following:

```
bool binarySearch(int *data,int &index,int lastPos, int target)
{
   int first, last, middle;
   first = 0;
   last = lastPos;
   while(last>first){
       middle=(last+first)/2;
       if(data[middle] == target)
           first=last+1;
       else if(data[middle]<target)
           first=middle+1;
       else
           last=middle-1;
   }
   if(data[middle] == target){
        index = middle;
        return true;
   }
   return false;
}
```

   SHOW ALL WORK!!

3. CS5318 Design of Programming Languages

   (a) Discuss three computational paradigms. A complete answer will include the model for the paradigm, one or more example languages, and characteristics of the paradigm.

   (b) Discuss attribute grammars. A complete answer will include a comparison of the power or expressivity of attribute grammars and context-free grammars, an example of an attribute grammar, and an example of a decorated parse tree.

4. CS5338 Formal Languages

   (a) Find a regular expression for each of the following languages.

      i. $\{w \in \{a, b\}* : w$ has $abab$ as a substring$\}$

      ii. $\{w \in \{a, b\}* : w$ has neither $aa$ nor $bb$ as a substring$\}$

   (b) Present Turing machines that decide the following languages over $\{a, b\}$:

      i. $L = \{a\}$

      ii. $L = \{a\}*$

      iii. $L = \{ab\}*$

5. CS5332 Advanced Database Systems

   *(20 marks total)* Consider the following typed relational schema describing books and borrowers in a library:

   Book(<u>isbn:integer</u>, title:string, publisher:string, year:integer)
   Author(<u>isbn:integer</u>, <u>name:string</u>, rank:integer)
   Borrower(<u>bid:integer</u>, name:string, address:string)
   Borrowings(<u>isbn:integer</u>, <u>borrower:integer</u>, whenTaken:date, whenReturned:date)

   Assume the following:

   - the fields that are underlined are the primary keys of the relations
   - there is a referential integrity constraint between "borrower" in relation Borrowings and "bid" in relation Borrower
   - there are also referential integrity constraints between "isbn" in relation Book and "isbn" in relations Author and Borrowings
   - dates are represented as a count of the number of days since Jan 1 2004
   - there is a system function `today` that returns today's `date`
   - the relational operators $<$ and $>$ allow you test *before* and *after* on `date`s
   - all loans are for a period of 14 days
   - the `whenReturned` field is `null` until the book is returned
   - the `rank` field specifies whether the author is the first, second, ... author

   Implement solutions to the following:

   (a) *(4 marks)* Write an SQL query to show all the books (`title` only) written by a particular author called "Ullman".

   (b) *(4 marks)* Write an SQL query to show which books (`title` only) have multiple authors.

   (c) *(4 marks)* Write an SQL query to show which books (`title` only) have never been borrowed.

   (d) *(4 marks)* Write an SQL query to show which books (`title` only) are currently out on loan.

   (e) *(4 marks)* Write an SQL query to list overdue books (`title`) and the borrower's name (Hints: you need to join relations Book, Borrowers and Borrowings).

6. CS5332 Advanced Database Systems

   Consider the following set of requirements for a university database that is used to keep track of student's transcripts.

   (a) The university keeps track of each student's name, student number, social security number, current address and phone, permanent address and phone, birthdate, sex, class (freshman, sophomore, ..., graduate), major department, minor department (if any), and degree program (B.A., B.S., ..., Ph.D.). Some user applications need to refer to the city, state, and zip code of the student's permanent address and to the student's last name. Both social security number and student number have unique values for each student.

   (b) Each department is described by a name, department code, office number, office phone, and college. Both name and code have unique values for each department.

   (c) Each course has a course name, description, course number, number of semester hours, level and offering department. The value of course number is unique for each course.

   (d) Each section has an instructor, semester, year, course, and section number. The section number distinguishes sections of the same course that are taught during the same semester/year; its values are 1,2,3,..., up to the number of sections taught during each semester.

   (e) A grade report has a student, section, letter grade, and numeric grade (0, 1, 2, 3, or 4).

   Design an ER schema for this application, and draw an ER diagram for that schema. Specify key attributes of each entity type and structural constraints on each relationship type. Note any unspecified requirements, and make appropriate assumptions to make the specification complete.

   **Remarks: ER schema consists of a set of normalized tables up to third normal form. You must place attributes of each normalized table in each entity of the ER diagram and indicate PK and FK for each entity. You must also describe the meaning of the notations you use for relationships. Your result of this problem should be a large ER diagram with description and notation explanation at the bottom of the ER diagram.**

7. CS5306 Advanced Operating Systems

   Describe three mutual exclusion methods in distributed systems, point out their advantages and disadvantages, and compare the number of messages communicated among processors.

8. CS5310 Advanced Networks

   Describe the essential elements required for a client and a server to perform a session of TCP/IP based communication. For both connectionless and connection-oriented communications, describe how the client and the server will obtain those required essential elements.

**Department of Computer Science**
**Graduate Comprehensive Exam**
**in**
**Computer Science**
**Spring 2005**

- **Answer the questions on the paper supplied.**
- **Answer question 1 or question 2. Answer question 3. Answer question 4. You should answer a total of three questions.**
- **Start each question on a new page. Write on only one side of the paper.**
- **Write your SIX-DIGIT Texas State ID in the top right corner of each page of your answer. Do NOT put your name anywhere on the answers.**
- **Put the number of the question being answered in the top left corner of each answer page.**
- **If the answer to a question is written on more than one page, number the pages consecutively.**

1. CS5391 Survey of Software Engineering. Explain in detail why adding people to a project that is behind schedule will make the situation worse. Explain what underlying issues are involved.

2. CS5391 Survey of Software Engineering. Describe how software testing is handled by each of the following software life cycle models: linear-sequential, incremental, spiral. In your descriptions, indicate how test planning is performed, how often testing occurs, and when testing occurs (with respect to other development phases such as requirements specification, design, implementation, and maintenance).

3. CS5329 Algorithm Design and Analysis

   (a) Recurrences.

      i. Setup a recurrence relation for the worst-case time complexity of the following algorithm. This algorithm outputs all permutations of LIST[K..M]. Assume the original call is PERM(LIST, 0,N-1). Also assume that the function SWAP(X,Y) runs in constant time.

```
PERM(list,k,m)
  if (k==m) then
    for i = 0 to  m write(list[i]);
  else
    SWAP(list[k],list[i]);
    PERM(list, k+1,m);
    SWAP(list[k],list[i]);
```

      ii. Can the recurrence relation from part 2(a)i be solved using the Master Method?

      iii. Show the steps in solving the following recurrence:

$$T(n) = \begin{cases} 1 & n = 1 \\ 4T(n/2) + n^3 & n > 1 \end{cases}$$

   (b) Design Techniques. Discuss the Greedy Strategy for algorithm design. A complete answer will include a comparison with Divide and Conquer and Dynamic Programming strategies, a characterization of problems that can be solved efficiently using dynamic programming, a discussion of the steps in algorithm development and an example.

4. CS5346 Advanced Aritficial Intelligence

(a) Consider the following Wumpus World

| | | | |
|---|---|---|---|
| Stench<br>Breeze | **WUMPUS** | Stench<br>Glitter<br>**GOLD** | Breeze |
| **PIT #2** | Stench<br>Breeze | Breeze | **PIT #1** |
| Breeze | | | Breeze |
| **START** | | | |

Write the facts and rules to determine the positions of the Wumpus and the PIT #1. Show all steps of the derivation in determining the positions of the Wumpus and the PIT #1.

(b) A database of a county banks loan department is represented as positive and negative instances in the following table:

| Individual | RATING | BAL | APP | INC | OK |
|---|---|---|---|---|---|
| 1 | 1 | 0 | 0 | 1 | 0 |
| 2 | 0 | 0 | 1 | 0 | 0 |
| 3 | 1 | 1 | 0 | 1 | 1 |
| 4 | 0 | 1 | 1 | 1 | 1 |
| 5 | 0 | 1 | 1 | 0 | 0 |
| 6 | 1 | 1 | 1 | 0 | 1 |
| 7 | 1 | 1 | 1 | 1 | 1 |
| 8 | 1 | 0 | 1 | 0 | 0 |
| 9 | 1 | 1 | 0 | 0 | 0 |

Derive a set of rules that covers all and only the positive instances.

**Department of Computer Science**
**Graduate Comprehensive Exam**
**in**
**Computer Science**
**Fall 2005**

- **Answer the questions on the paper supplied.**
- **Answer question 1. Answer question 2 or question 3. Answer question 4 or question 5 or question 6. You should answer a total of three questions.**
- **Start each question on a new page. Write on only one side of the paper.**
- **Write your SIX-DIGIT Texas State ID in the top right corner of each page of your answer. Do NOT put your name anywhere on the answers.**
- **Put the number of the question being answered in the top left corner of each answer page.**
- **If the answer to a question is written on more than one page, number the pages consecutively.**

1. CS5391 Survey of Software Engineering

   Each of the following testing techniques can be used during software development. Describe the methodology for each technique, its goal and at which phase of software development it is used.

   (a) Program inspection.

   (b) Black–box testing.

   (c) White–box or structural testing.

   (d) Loading or stress testing.

   (e) Regression testing.

   (f) Incremental integration testing.

2. CS5318 Design of Programming Languages

   (a) Programming languages are sometimes classified by the computational paradigms they support. Discuss two computational paradigms. A complete answer will include the model underlying the paradigm, characteristics of languages that support the paradigm, and an example of a language that is most closely associated with the paradigm.

   (b) Discuss, with examples, the roles of context free grammars and attribute grammars in programming language design.

3. CS5338 Formal Languages.

   (a) Give a precise definition of a Turing Machine.

   (b) Describe in a phrase or sentence the language accepted by DFA $M$:

$$M = (\{q_0, q_1, q_2\}, \{0, 1\}, \delta, q_0, \{q_2\})$$

   where $\delta$ is given by the following table:

| $\delta$ | 0 | 1 |
|---|---|---|
| $q_0$ | $q_1$ | $q_0$ |
| $q_1$ | $q_2$ | $q_0$ |
| $q_2$ | $q_2$ | $q_0$ |

   (c) Minimize the following DFA:

$$M = (\{q_0, q_1, q_2\}, \{0, 1\}, \delta, q_0, \{q_1, q_2\})$$

   where $\delta$ is given by the following table:

| $\delta$ | 0 | 1 |
|---|---|---|
| $q_0$ | $q_1$ | $q_0$ |
| $q_1$ | $q_2$ | $q_0$ |
| $q_2$ | $q_2$ | $q_0$ |

   (d) Discuss the closure properties of regular languages. A complete answer will include a list of operations under which the regular languages are closed and examples.

4. CS5306 Advanced Operating Systems

   In a certain system, each process typically uses a critical section many times before another process requires it. Explain why Ricart and Agrawalas multicast-based mutual exclusion algorithm is inefficient for this case, and describe how to improve its performance.

5. CS5310 Network and Communication Systems

   Briefly describe the seven layer ISO's OSI model (For each layer briefly describe its services and design issues). Explain why the top four layers are called "end-to-end" layers. You should support your explanation with an application example.

6. CS5332 Database Theory and Design *(26 marks total)*
   The grade book database consists of the relations defined in the six tables shown below:

   Catalog(<u>cno</u>, ctitle)
   Students(<u>sid</u>, fname, lname, middlenameInitial)
   Courses(<u>term, lineno</u>, cno, A, B, C, D)
   Components(<u>*term, lineno*</u>, <u>compname</u>, maxpoints, weight)
   Enrolls(<u>*sid, term, lineno*</u>)
   Scores(<u>*sid, term, lineno, compname*</u>, points)

   Primary keys are underlined. Foreign keys are shown in italic font. Primary keys that are also foreign keys are shown in underline italic font. For example, the term and lineno attributes in Components relation is part of the primary key for Components and is also a foreign key of the table Courses.

   The Catalog relation contains information about course numbers and the title of the course. For example cs5332 is a course number for 'Database Theory and Design'. The Student relation contains information about the students with sid as the primary key for that relation. In Courses relation, the *term* attribute corresponds to the term (Fall2000 or Spring2005) in which the course was taught; the *lineno* is a unique section number assigned by the registrar of the university. A, B, C and D are attributes that represent numerical values that correspond with each letter grade. The Components relation contains information about the various grading components (such as quizzes, mid-term, final, lab) for a particular course. The combination of *sid*, *term*, *lineno* and *compname* forms the primary key.

   The Enrolls relation records information about which student was enrolled in which course. The Scores relation records the grading component scores (or points) for each student enrolled in a course.

   Write SQL to answer the following queries:

   (a) *(5 marks)* Get the names of students enrolled in the 'Database Theory and Design' class in the 'Spring2005' term?
   (b) *(5 marks)* Get the SID values of students who did not enrolled in any class during the Spring2005 term.
   (c) *(5 marks)* Get the SID values of students who have enrolled in cs5332 or cs5369A.
   (d) *(5 marks)* Get the SID values of students who have enrolled in cs5332 and cs5369A
   (e) *(6 marks)* Get the SID values of students who have enrolled in all the courses in the Catalog (hint: division)

**Department of Computer Science**
**Graduate Comprehensive Exam**
**in**
**Computer Science**
**Spring 2006**

- **Answer the questions on the paper supplied.**
- **Answer question 1. Answer question 2 or question 3. Answer question 4. You should answer a total of three questions.**
- **Start each question on a new page. Write on only one side of the paper.**
- **Write your SIX-DIGIT Texas State ID in the top right corner of each page of your answer. Do NOT put your name anywhere on the answers.**
- **Put the number of the question being answered in the top left corner of each answer page.**
- **If the answer to a question is written on more than one page, number the pages consecutively.**

1. CS5391 Survey of Software Engineering.

   (a) Explain the differences between functional and non-functional requirements. Describe how testing methodologies differ for functional and non-functional requirements.

   (b) Describe the following software process models and compare and contrast them being specific concerning the type of development they are best suited for and how the choice of model may effect the time and cost of development.

      i. waterfall model
      ii. incremental model
      iii. spiral model

2. CS5318 Design of Programming Languages.

   (a) Discuss two common computational paradigms. A complete answer will include computational models, data and control structures, and an example language.

   (b) Discuss in detail and with examples the roles of context-free and attribute grammars in programming language design.

3. CS5338 Formal Languages.

   Answer all the parts of this question

   (a) Prove or disprove the following statement:
   "The language $L = \{w \in \{a, b\}^* : \ w = (ab)^m, m \geq 0\}$, is a regular language"

   (b) Prove or disprove the following statement:
   "The language $L = \{w \in \{a, b\}^* : w = a^2 b^m a^m, m \geq 0\}$, is not a regular language"

   (c) Show (no need for a formal proof) that the following statement is true:
   "The language $L = \{w \in \{a, b, c\}^*, \text{ and } w_1 \in \{a, b\}^* : w = w_1 c w_1\}$,
   is not a context free language"

   (d) Show a context-free grammar and an accepting PDA for the following language:
   $L = \{w \in \{a, b, c\}^* : w = a^n b^{n+m} c^m; n, m \geq 0\}$

   (e) Prove by induction that every string that belongs to the language:
   $L = \{a^n c^m b^n : n \geq 0, m \geq 1\}$,
   is produced by the CFL with production rules:
   $\{S \rightarrow aSb|T; T \rightarrow cT|c\}$.

4. CS5346 Advanced Artificial Intelligence.

   (a) Trace the MIN MAX algorithm (shown in Fig.1) on the example shown in Fig. 2. MAX makes the first move at the root node A. Evaluation function values are computed from MAX's point of view.

   (b) Explain in detail with an example how Alpha-Beta Pruning can enhance the efficiency of search used in the algorithms of two player games

**Algorithm: MINIMAX(Position, Depth, Player)**

1. If DEEP-ENOUGH(*Position*, *Depth*), then return the structure

   VALUE = STATIC(*Position*, *Player*);
   PATH = nil

   This indicates that there is no path from this node and that its value is that determined by the static evaluation function.

2. Otherwise, generate one more ply of the tree by calling the function MOVE-GEN(Position, Player) and setting SUCCESSORS to the list it returns.

3. If SUCCESSORS is empty, then there are no moves to be made, so return the same structure that would have been returned if DEEP-ENOUGH had returned true.

4. If SUCCESSORS is not empty, then examine each element in turn and keep track of the best one. This is done as follows.

   Initialize BEST-SCORE to the minimum value that STATIC can return. It will be updated to reflect the best score that can be achieved by an element of SUCCESSORS.

   For each element SUCC of SUCCESSORS, do the following:

   (a) Set RESULT-SUCC to
   MINIMAX(SUCC, *Depth* + 1, OPPOSITE(*Player*))[3]
   This recursive call to MINIMAX will actually carry out the exploration of SUCC.

   (b) Set NEW-VALUE to – VALUE(RESULT-SUCC). This will cause it to reflect the merits of the position from the opposite perspective from that of the next lower level.

   (c) If NEW-VALUE > BEST-SCORE, then we have found a successor that is better than any that have been examined so far. Record this by doing the following:

      i. Set BEST-SCORE to NEW-VALUE.

      ii. The best known path is now from CURRENT to SUCC and then on to the appropriate path down from SUCC as determined by the recursive call to MINIMAX. So set BEST-PATH to the result of attaching SUCC to the front of PATH(RESULT-SUCC).

5. Now that all the successors have been examined, we know the value of Position as well as which path to take from it. So return the structure

   VALUE = BEST-SCORE
   PATH = BEST-PATH

Fig 1

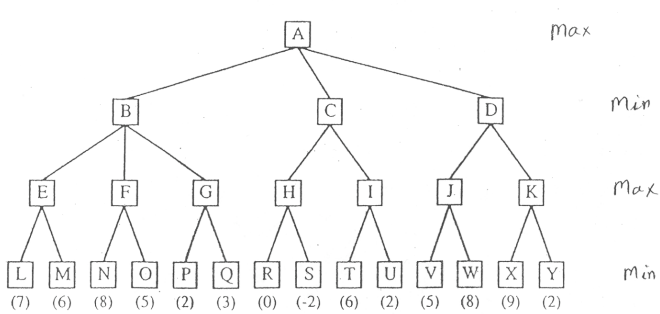



Fig.2

**Department of Computer Science**
**Graduate Comprehensive Exam**
**in**
**Computer Science**
**Fall 2006**

- **Answer the questions on the paper supplied.**
- **Answer question 1 or question 2. Answer question 3. Answer question 3 or question 4 or question 5 or question 6. You should answer a total of three questions.**
- **Start each question on a new page. Write on only one side of the paper.**
- **Write your SIX-DIGIT Texas State ID in the top right corner of each page of your answer. Do NOT put your name anywhere on the answers.**
- **Put the semester and year of the exam in the appropriate blank at the top of each answer page.**
- **Put the number of the question being answered in the appropriate blank on each answer page.**
- **If the answer to a question is written on more than one page, number the pages consecutively.**

1. CS5329 Algorithm Design and Analysis
   (a) Write the QUICK SORT algorithm with its PARTITION function.
   (b) Use a Decision Tree in best case partition to describe the total running time as $\Theta(n \lg n)$.

2. CS5329 Algorithm Design and Analysis
   (a) Recurrences.
      i. Setup a recurrence relation for the worst-case time complexity of the following algorithm. This algorithm moves n rings from spindle a to spindle b using spindle c for temporary storage. Assume that the function MOVE1(A,B) runs in constant time.

```
RINGS(n,a,b,c)
  if (n ≤ 0) then return;
  if (n == 1) then  MOVE1(a,b);
  RINGS(n-1,a,c,b);
  MOVE1(a,b);
  RINGS(n-1,c,b,a);
```

      ii. Can the recurrence relation from part 2(a)i be solved using the Master Method?
      iii. Show the steps in solving the following recurrence:

$$T(n) = \begin{cases} 1 & n = 1 \\ 3T(n/3) + n^2 & n > 1 \end{cases}$$

   (b) Design Techniques. Discuss the methodology, or elements, for algorithm design using dynamic programming. A complete answer will include a description of each step, or element, and the application of each step to a problem that has an efficient dynamic programming solution.

3. CS5346 Advanced Artificial Intelligence

(a) Verify the soundness of the unit resolution rule by constructing a truth table.

(b) In the Block World problem, the following operations are used:

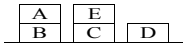
STACK($x$, $y$)
- P: CLEAR($y$) ^ HOLDING($x$)
- D: CLEAR($y$) ^ HOLDING($x$)
- A: ARMEMPTY ^ ON($x$, $y$)

UNSTACK($x$, $y$)
- P: ON($x$, $y$) ^ CLEAR($x$) ^ ARMEMPTY
- D: ON($x$, $y$) ^ ARMEMPTY
- A: HOLDING($x$) ^ CLEAR($y$)

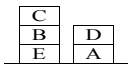
PICKUP($x$)
- P: CLEAR($x$) ^ ONTABLE($x$) ^ ARMEMPTY
- D: ONTABLE($x$) ^ ARMEMPTY
- A: HOLDING($x$)

PUTDOWN($x$)
- P: HOLDING($x$)
- D: HOLDING($x$)
- A: ONTABLE($x$) ^ ARMEMPTY

The following are the start and goal states. Using the Goal Stack Planning method, describe each step which needs to be performed to transform the start state to the goal state.




start: ON(A, B) ^
ON (E, C) ^
ONTABLE(B) ^
ONTABLE(C) ^
ONTABLE(D) ^
ARMEMPTY ^
CLEAR(A) ^
CLEAR(E) ^
CLEAR(D)




goal: On(B, E) ^
ON(C, B) ^
ON(D, A) ^
ONTABLE (E)
ONTABLE (A)
CLEAR (C)
CLEAR (D)
ARMEMPTY

4. CS5306 Advanced Operating Systems

   (a) Why do stateless servers have to include a file offset in each request? Is this also needed for statefull servers?

   (b) A file is replicated on 10 servers.

      i. explain the voting algorithm technique (Giffords algorithm).

      ii. List all combinations of read quorum and write quorum that are permitted by the voting algorithm.

   (c) Describe two election algorithms in distributed systems, point out their advantages and disadvantages.

5. CS5310 Network and Communication Systems

   What is modulation? Why modulations are needed? What are popular modulation techinques? Please provide three examples of real applications where modulations are essential. Please provide necessary details to support your argument.

6. CS5332 Database Theory and Design

   From what you understand about the Texas State University system, draw an Entity-Relationship Diagram (E/R Model) including at least, but not limited to, the following entities:

   CLASS, CLASSROOM
   STUDENT, COURSE, INSTRUCTOR

   CLASS is the class for a particular semester; COURSE is the course described in the catalog.

   You must figure out correct primary keys, correct foreign keys and correct relationships. You must normalize all entities (tables). You must specify other essential attributes for all entities. You must also create other entities (tables) if needed.

7. CS5332 Database Theory and Design The grade book database consists of the relations defined in the six tables shown below:

   Catalog(<u>cno</u>, ctitle)
   Students(<u>sid</u>, fname, lname, middlenameInitial)
   Courses(<u>term, lineno</u>, cno, A, B, C, D)
   Components(<u>*term, lineno*</u>, <u>compname</u>, maxpoints, weight)
   Enrolls(<u>*sid, term, lineno*</u>)
   Scores(<u>*sid, term, lineno, compname*</u>, points)

   Primary keys are underlined. Foreign keys are shown in italic font. Primary keys that are also foreign keys are shown in underline italic font. For example, the term and lineno attributes in Components relation is part of the primary key for Components and is also a foreign key of the table Courses.

   The Catalog relation contains information about course numbers and the title of the course. For example cs5332 is a course number for 'Database Theory and Design'. The Student relation contains information about the students with sid as the primary key for that relation. In Courses relation, the *term* attribute corresponds to the term (Fall2000 or Spring2005) in which the course was taught; the *lineno* is a unique section number assigned by the registrar of the university. A, B, C and D are attributes that represent numerical values that correspond with each letter grade. The Components relation contains information about the various grading components (such as quizzes, mid-term, final, project) for a particular course. The combination of *sid*, *term*, *lineno* and *compname* forms the primary key.

   The Enrolls relation records information about students who were enrolled in a specific course. The Scores relation records the grading component scores (or points) for each student enrolled in a course.

   Write SQL to answer the following queries:

   (a) *(5 marks)* Get the names of students enrolled in the 'Database Theory and Design' class in the 'Spring2005' term?
   (b) *(5 marks)* Get the SID values of students who did not enrolled in any class during the Spring2005 term.
   (c) *(5 marks)* Get the SID values of students who have enrolled in cs5332 or cs5369a.
   (d) *(5 marks)* Get the SID values of students who have enrolled in cs5332 and cs5369a
   (e) *(6 marks)* Get the SID values of students who have enrolled in all the courses in the Catalog (hint: division)

**Department of Computer Science**
**Graduate Comprehensive Exam**
**in**
**Computer Science**
**Spring 2007**

- **Answer the questions on the paper supplied.**
- **Answer question 1 or question 2. Answer question 3. Answer question 4 or question 5 or question 6 or question 7. You should answer a total of three questions.**
- **Start each question on a new page. Write on only one side of the paper.**
- **Write your SIX-DIGIT Texas State ID in the top right corner of each page of your answer. Do NOT put your name anywhere on the answers.**
- **Put the semester and year of the exam in the appropriate blank at the top of each answer page.**
- **Put the number of the question being answered in the appropriate blank on each answer page.**
- **If the answer to a question is written on more than one page, number the pages consecutively.**

1. CS5338 Formal Languages

   Answer all the parts of this question

   (a) Prove or disprove the following statement: "The language

   $$L = \{w \in \{a,b,c\}^* : \text{ every c has an a immediately to its left and a b immediately to its right}\}$$

   is a regular language"

   (b) Prove or disprove the following statement: "The language

   $$L = \{w \in \{a,b\}^*;\ w = a^n b^j :\ |n-j| = 2\},$$

   is not a regular language"

   (c) Show, using a "sketch" of the pumping lemma, that the following statement is true: "The language

   $$L = \{w : w = yxx^R y \text{ and } x, y \in \{a,b\}^*\}$$

   is not a context free language"

   (d) Show a context-free grammar and an accepting PDA for the following language:

   $$L = \{a^n b^m c^k : n = m + k \text{ or } m = n + k\}$$

   (e) Prove by induction that every string that belongs to the language:

   $$L = \{a^n c^{m+1} b^m : n \geq 0, m \geq 1\},$$

   is produced by the CFL with production rules: {
   $S \rightarrow aS|T;$
   $T \rightarrow cTb|c$
   }

2. CS5318 Programming Language Design

   (a) Discuss in detail and with examples the roles of context-free and attribute grammars in programming language design.

   (b) Discuss the significance of abstraction in programming languages.

3. CS5391 Survey of Software Engineering

   Describe the Capability Maturity Model (CMM) including the levels and the key process areas which much be completed for each level to be attained. What organization developed the CMM? How should CMM be used by a software development organization?

4. CS5306 Advanced Operating Systems

   Suppose that four processes initially have the following local clock values:

| P1 | P2 | P3 | P4 |
|---|---|---|---|
| 8 | 17 | 2 | 33 |

   These values are to be advanced by the minimum necessary amount 1 for each new event (including the first). The only events which each process counts are the sending and receiving of messages, each of which is one distinct event.

   Suppose the following sequence of messages occurs, and that each message is sent only after the previous one is received. Give the time stamp for each message.

| **MESSAGE** | **TIME STAMP** |
|---|---|
| P2 sends a message to P4 | _____ |
| P4 sends a message to P2 | _____ |
| P2 sends a message to P1 | _____ |
| P1 sends a message to P3 | _____ |
| P3 sends a message to P2 | _____ |

5. CS5310 Network and Communication Systems

   What is modulation? Why modulations are needed? What are popular modulation techinques? Please provide three examples of real applications where modulations are essential. Please provide necessary details to support your argument.

6. CS5332 Database Theory and Design

   Design an E/R diagram with normalized tables (entities), attributes, PKs, FKs and relationships for the following real world problem. (All base tables/relations must be in 3NF or BCNF.)

   Read the following case study, which describes the data requirements for a video rental company. The video rental company has several branches throughout the USA. The data held on each branch is the branch address made up of a street, city, state, and zip code, and the telephone number. Each branch is given a branch number, which is unique throughout the company. Each branch is allocated staff, which includes a Manager. The Manager is responsible for the day-to-day running of a given branch. The data held on a member of staff is his or her name, position, and salary. Each member of a staff is given a staff number, which is unique throughout the company. Each branch has a stock of videos. The data held on a video is the catalog number, video number, title, category, daily rental, cost, status, and the names of the main actors and the director. The catalog number uniquely identifies each video. However, in most cases, there are several copies of each video at a branch, and the individual copies are identified using the video number. A video is given a category such as Action, Adult, Children, Drama, Horror, or Sci-Fi. The status Indicates whether a specific copy of a video is available for rent. Before hiring a video from the company, a customer must first register as a member of a local branch. The data held on a member is the first and last name, address, and the date that the member registered at a branch. Each member is given a member number, which is unique throughout the company. Once registered, a member is free to rent videos, up to a maximum of ten at any one time. The data held on each video rented is the rental number, the full name and number of the member, the video number, title, and daily rental, and the dates the video is rented out and returned. The rental number is unique throughout the company.

7. CS5332 Database Theory and Design

*(20 marks total)* Consider the following typed relational schema describing books and borrowers in a library:

Book(<u>isbn:integer</u>, title:string, publisher:string, year:integer)
Author(<u>isbn:integer</u>, <u>name:string</u>, rank:integer)
Borrower(<u>bid:integer</u>, name:string, address:string)
Borrowings(<u>isbn:integer</u>, <u>borrower:integer</u>, whenTaken:date, whenReturned:date)

Assume the following:

- the fields that are underlined are the primary keys of the relations
- there is a referential integrity constraint between "borrower" in relation Borrowings and "bid" in relation Borrower
- there are also referential integrity constraints between "isbn" in relation Book and "isbn" in relations Author and Borrowings
- dates are represented as a count of the number of days since Jan 1 2004
- there is a system function `today` that returns today's `date`
- the relational operators $<$ and $>$ allow you test *before* and *after* on `date`s
- all loans are for a period of 14 days
- the `whenReturned` field is `null` until the book is returned
- the `rank` field specifies whether the author is the first, second, ... author

Implement solutions to the following:

(a) *(4 marks)* Write an SQL query to show all the books (`title` only) written by a particular author called "Ullman".

(b) *(4 marks)* Write an SQL query to show which books (`title` only) have multiple authors.

(c) *(4 marks)* Write an SQL query to show which books (`title` only) have never been borrowed.

(d) *(4 marks)* Write an SQL query to show which books (`title` only) are currently out on loan.

(e) *(4 marks)* Write an SQL query to list overdue books (`title`) and the borrower's name (Hints: you need to join relations Book, Borrowers and Borrowings).

**Department of Computer Science**
**Graduate Comprehensive Exam**
**in**
**Computer Science**
**Fall 2007**

- **Answer the questions on the paper supplied.**
- **Answer question 1. Answer question 2 or question 3. Answer question 4 or question 5 or question 6 or question 7. You should answer a total of three questions.**
- **Start each question on a new page. Write on only one side of the paper.**
- **Write your SIX-DIGIT Texas State ID in the top right corner of each page of your answer. Do NOT put your name anywhere on the answers.**
- **Put the semester and year of the exam in the appropriate blank at the top of each answer page.**
- **Put the number of the question being answered in the appropriate blank on each answer page.**
- **If the answer to a question is written on more than one page, number the pages consecutively.**

1. CS5346 Advanced Artificial Intelligence

   (a) Explain in detail with example, the difference between the Uniform Cost Search and A* Search.

   (b) Transform the following English sentences first into a predicate logic form and then either to conjunctive normal form or implicative normal form:

      1. Marcus was a man.
      2. Marcus was a Pompeian.
      3. All Pompeians were Romans.
      4. Caesar was a ruler.
      5. All Romans were either loyal to Caesar or hated him.
      6. Everyone is loyal to someone.
      7. People only try to assassinate rulers they are not loyal to.
      8. Marcus tried to assassinate Caesar.

      Using the refutation rule, prove Marcus hated Caesar.

2. CS5329 Algorithm Design and Analysis

   (a) State and solve a recurrence relation for the running time of the following algorithm. Assume the original call is
   rec-srch(A, 1, n, x).

```
// given an array A indexed from l to r, of elements that can be compared,
// and a key x, rec-srch(A,l,r) returns the location of the first
// occurrence of key x in A[l..r], if x is in the array, and 0 otherwise.
rec-srch(A, ℓ, r, x)
   if (ℓ > r) return 0
   if (x == A[ℓ])
      then return ℓ;
      else return rec-srch(A, ℓ+1, r, x);
```

   (b) Discuss in detail and with examples the greedy algorithm design technique. A complete answer will include a discussion of characteristics of problems for which a greedy strategy yields an efficient solution and a comparison with other design techniques such as divide-and-conquer and dynamic programming.

3. CS5329 Algorithm Design and Analysis

   Suppose that the splits at every level of quicksort are in the proportion $1-\alpha$ to $\alpha$, where $0 < \alpha \leq 1/2$ is a constant. show that the minimum depth of a leaf in the recursion tree is approximately $-\lg n/\lg \alpha$ and the maximum depth is approximately $-\lg n/\lg(1-\alpha)$

4. CS5306 Advanced Operating Systems

   What are the two rules in the Lamports happens-before relationship? Can happens-before relationship provide a total order of the events in a distributed system? If so, why? If not, how to provide a total order? A process can fork a child process. How to modify the happens-before relationship to include this situation?

5. CS5310 Network and Communication Systems

   Answer the following questions based on the TCP/IP model:

   (a) What are the main differences between TCP and UDP based on congestion control?
   (b) What is the purpose of the transport layer and why it is referred to as End-to-End?
   (c) Explain briefly how IP packets are forwarded by routers.

6. CS5332 Database Theory and Design

   Draw an extended ER diagram for the following business rules: You may create more attributes to complete the problem. You must normalize the entities, and specify pks and fks:

   Design a database for Ray. He is interested in movies and wants to keep information on movies, actors, and directors in a database. The only user is Ray and he needs to produce the following reports:

   (a) For each director, list his or her number, name, and the year he or she was born. If the director is deceased, list the year of death.
   (b) For each movie, list its number, title, the year the movie was made, and its type (for example, Comedy, Drama, Science Fiction, and so on).
   (c) For each movie, list its number, title, the number and name of its director, the critics rating, the MPAA rating (G, PG, PG-13, or R), the number of awards for which the movie was nominated, and the number of awards the movie won. (The critics rate the movie with a number of "stars." Four stars is the top rating possible. Zero stars is the worst possible rating.)
   (d) For each actor, list his or her number, name, birthplace, and the year he or she was born. If the actor is deceased, list the year of death.
   (e) For each movie, list its number and title, along with the number and name of the actors who appeared in it.
   (f) For each actor, list his or her number and name, along with the number and name of the movies in which the actor starred.

7. CS5332 Database Theory and Design

*(20 marks total)*

The customer database consists of the relations defined in the four tables shown below:

Customer(<u>custNo</u>, custName, state, phone)
Item(<u>itemNo</u>, ItemName, itemPrice, qtyOnHand)
Invoice(<u>invNo</u>, inDate, *custNo*)
Invitem(<u>*invNo, itemNo*</u>, qty)

Primary keys are underlined. Foreign keys are shown in italic font. Primary keys that are also foreign keys are shown in underline italic font. For example, the invNo and itemNo attributes in Invitem relation is the primary key for Invitem and is also foreign keys of tables Invoice and Item. Below are sample data from each of the above relation.

```
Customer table
--------------
custNo   custName    state     phone
211       Garcia      TX       732-555-1000
212       Parkin      NY       212-555-2000
225       Ellen       NJ       973-555-3333
239       Bayer       FL       401-555-7777

Item table
----------
itemNo     itemName   itemPrice     qtyOnHand
 1          Screw      5.00           50
 2          Nut        2.25           110
 3          Bolt       3.99           75
 4          Hammer     10.00          125
 5          Washer     6.00           300
 6          Nail       0.99           400

Invoice table
-------------
invNo      invDate        custNo
1001        05-Sep-07      212
1002        17-Sep-07      225
1003        17-Sep-07      239
1004        18-Sep-07      211
1005        21-Sep-07      212

invItem table
-------------
invNo        itemNo           qty
1001          1                5
1001          3                5
1001          5                9
1002          1                2
1002          2                3
1003          1                7
1003          2                1
1004          4                5
1005          4                10
```

Write SQL to answer the following queries:

(a) *(5 marks)* Find items which have the itemPrice between $5 and $10?

(b) *(5 marks)* Count the number of items ordered in each invoice.

(c) *(5 marks)*

(d) *(5 marks)* Find items that are cheaper than nut. You must display the itemName and the associated itemPrice. Beware that nut can be written in both upper and lower case.

(e) *(5 marks)* Find the invoices in which three or more items are ordered.

(f) *(5 marks)* List the invoices, customer names and the items purchased by them. The result should display invoices, customer names and item names together ordered by the customer name.

**Department of Computer Science**
**Graduate Comprehensive Exam**
**in**
**Computer Science**
**Spring 2008**

- **Answer the questions on the paper supplied.**
- **Answer question 1 or answer question 2, but if you have registered for the Forensics exam, skip questions 1 and 2 and answer question 3. Answer question 4. Answer question 5 or answer question 6 or answer question 7 or answer question 8. You should answer a total of three questions.**
- **Start each question on a new page. Write on only one side of the paper.**
- **Write your SIX-DIGIT Texas State ID in the top right corner of each page of your answer. Do NOT put your name anywhere on the answers.**
- **Put the semester and year of the exam in the appropriate blank at the top of each answer page.**
- **Put the number of the question being answered in the appropriate blank on each answer page.**
- **If the answer to a question is written on more than one page, number the pages consecutively.**

1. CS5318 Design of Programming Languages

   (a) Using the following grammar:

| | | |
|---|---|---|
| expr | → | expr + term \| term |
| term | → | term * factor \| factor |
| factor | → | (expr) \| number |
| number | → | number digit \| digit |
| digit | → | 0 \| 1 \| 2 \| 3 \| 4 \| 5 \| 6 \| 7 \| 8 \| 9 |

   Draw a parse tree for the arithmetic expression:

$$3 + 4 * 5 + 6 * 7$$

   (b) Capitalization of articles at the beginning of a sentence is viewed as context-sensitivity in the text. However, in the simple English grammar:

| | | | |
|---|---|---|---|
| (1) | sentence | → | noun-phrase verb-phrase |
| (2) | noun-phrase | → | article noun |
| (3) | article | → | a \| the |
| (4) | noun | → | girl \| dog |
| (5) | verb-phrase | → | verb noun-phrase |
| (6) | verb | → | sees \| pets |

   capitalization can be achieved by adding only context free rules. How would you do this?

   (c) Discuss two common computational paradigms. A complete answer will include computational models, data and control structures, and an example language.

2. CS5338 Formal Languages

   Answer all the parts of this question

   (a) Prove or disprove the following statement: "The language $L = \{w \in \{a, b\}^* : w = (ab)^m, m \geq 0\}$, is a regular language"

   (b) Prove or disprove the following statement: "The language $L = \{w \in \{a, b\}^* : w = a^2 b^m a^m, m \geq 0\}$, is not a regular language"

(c) Show that the following statement is true: "The language $L = \{w \in \{a, b, c\}^*$, and $w_1 \in \{a, b\}^* : w = w_1 c w_1\}$, is not a context free language"

(d) Show a context-free grammar and an accepting PDA for the following language: $L = \{w \in \{a, b, c\}^* : w = a^n b^{n+m} c^m; n, m \geq 0\}$

(e) Prove by induction that every string that belongs to the language: $L = \{a^n c^m b^n : n \geq 0, m \geq 1\}$, is produced by the CFL with production rules: $\{S \rightarrow aSb|T; T \rightarrow cT|c\}$.

3. CS5369F Digital Forensics

a. How does one go about making a forensic copy of a hard drive?

b. What is the "slack space" on a hard drive? How may data be hidden in it so that the operating system can't see the data?

4. CS5391 Survey of Software Engineering

1) In his famous paper "No Silver Bullet" Fred Brooks mentions several properties of software. Describe what he meant by the following properties (or your understanding of them) and explain how these properties make it harder for us to develop software systems as compared to mechanical systems (e.g. cars).

(a) Complexity.

(b) Conformity.

(c) Changeability.

(d) Invisibility.

5. CS5310 Network and Communication Systems

(a) Describe the concept of multiplexing.

(b) Describe methods that can achieve multiplexing. Discuss the pros and cons of the methods.

(c) Provide at least two real useful examples of multiplexing in socket programming.

6. CS5306 Advanced Operating Systems

(a) Design a distributed algorithm to synchronize physical clocks in a distributed system in a room.

(b) Design a centralized algorithm to do the same thing.

7. CS5332 Data Base Theory and Design

The customer database consists of the relations defined in the four tables shown below:

Customer(<u>custNo</u>, custName, state, phone)
Item(<u>itemNo</u>, ItemName, itemPrice, qtyOnHand)
Invoice(<u>invNo</u>, inDate, *custNo*)
Invitem(<u>*invNo, itemNo*</u>, qty)

Primary keys are underlined. Foreign keys are shown in italic font. Primary keys that are also foreign keys are shown in underline italic font. For example, the invNo and itemNo attributes in Invitem relation is the primary key for Invitem and is also foreign keys of tables Invoice and Item. Below are sample data from each of the above relation.

```
Customer table
--------------
custNo   custName    state     phone
211       Garcia      TX       732-555-1000
212       Parkin      NY       212-555-2000
225       Ellen       NJ       973-555-3333
239       Bayer       FL       401-555-7777

Item table
----------
itemNo     itemName   itemPrice     qtyOnHand
 1          Screw      5.00           50
 2          Nut        2.25           110
 3          Bolt       3.99           75
 4          Hammer     10.00          125
 5          Washer     6.00           300
 6          Nail       0.99           400

Invoice table
-------------
invNo      invDate        custNo
1001        05-Sep-07      212
1002        17-Sep-07      225
1003        17-Sep-07      239
1004        18-Sep-07      211
1005        21-Sep-07      212

invItem table
-------------
invNo        itemNo           qty
1001          1                5
1001          3                5
1001          5                9
1002          1                2
1002          2                3
1003          1                7
1003          2                1
1004          4                5
1005          4                10
```

Write SQL to answer the following queries:

(a) *(5 marks)* Find items which have the itemPrice between $5 and $10?

(b) *(5 marks)* Count the number of items ordered in each invoice.

(c) *(5 marks)* Find customers who are not from "NY" or "NJ"

(d) *(5 marks)* Find items that are cheaper than nut. You must display the itemName and the associated itemPrice. Beware that nut can be written in both upper and lower case.

(e) *(5 marks)* Find the invoices in which three or more items are ordered.

(f) *(5 marks)* List the invoices, customer names and the items purchased by them. The result should display invoice number, customer names and item names together ordered by the customer name.

8. CS5332 Data Base Theory and Design

| staffNo | dentistName | patNo | patName | appointment date | appointment time | SurgeryNo |
|---|---|---|---|---|---|---|
| S1011 | Tony Smith | P100 | Gillian White | 12-Sep-04 | 10.00 | S15 |
| S1011 | Tony Smith | P105 | Jill Bell | 12-Sep-04 | 12.00 | S15 |
| S1024 | Helen Pearson | P108 | Ian MacKay | 12-Sep-04 | 10.00 | S10 |
| S1024 | Helen Pearson | P108 | Ian MacKay | 14-Sep-04 | 14.00 | S10 |
| S1032 | Robin Plevin | P105 | Jill Bell | 14-Sep-04 | 16.30 | S15 |
| S1032 | Robin Plevin | P110 | John Walker | 15-Sep-04 | 18.00 | S13 |

Figure A. Table displaying sample dentist/patient appointment data.

The table shown in Figure A lists sample dentist/patient appointment data. A patient is given an appointment at a specific time and date with a dentist located at a particular surgery. On each day of patient appointments, a dentist is allocated to a specific surgery for that day.

(a) Identify the functional dependencies represented by the attributes shown in the table of Figure A. State any assumptions you make about the data and the attributes shown in this table.

(b) Identify the primary, alternate, and foreign keys in your 3NF relations. Draw an E/R for this database system.

**Department of Computer Science**
**Graduate Comprehensive Exam**
**in**
**Computer Science**
**Fall 2008**

- **Answer the questions on the paper supplied.**
- **Answer question 1. Answer question 2 or answer question 3. Answer question 4 or answer question 5. You should answer a total of three questions.**
- **Start each question on a new page. Write on only one side of the paper.**
- **Write your SIX-DIGIT Texas State ID in the top right corner of each page of your answer. Do NOT put your name anywhere on the answers.**
- **Put the semester and year of the exam in the appropriate blank at the top of each answer page.**
- **Put the number of the question being answered in the appropriate blank on each answer page.**
- **If the answer to a question is written on more than one page, number the pages consecutively.**

1. CS Advanced Artificial Intelligence

   (a) Explain the following in detail with examples:
   Admissible heuristic
   Complexity of A* algorithm.

   (b) By generating truth tables, determine for each of the following sentences if it is valid, satisfiable or unsatisfiable:
   (i) $(Smoke \Rightarrow Fire) \Rightarrow ((Smoke \cap Heat) \Rightarrow Fire)$
   (ii) $Smoke \cup File \cup \neg Smoke \cup \neg Fire$
   (iii) $(Smoke \Rightarrow Fire) \Rightarrow (Fire \Rightarrow Smoke)$
   (iv) $P \cap (Q \cup R) \Leftrightarrow (P \cap Q) \cup (P \cap R)$

2. CS5318 Design of Programming Languages

   (a) Discuss in detail and using examples the role of abstraction in programming language design.

   (b) Discuss attribute grammars. A complete answer will include a comparison of the power or expressivity of attribute grammars and context-free grammars, an example of an attribute grammar, and an example of a decorated parse tree.

3. CS5338 Formal Languages

   1) Answer all the parts of this question

   (a) Define the term ambiguous context free grammar and:

   i) Explain in details what the procedure to disambiguate a grammar.

   ii) Is it always possible to disambiguate a CFG? If your answer is no, then please provide an example of a language that cannot be disambiguated.

   iii) Show two different right most deviations for the following grammar: $E \rightarrow E{+}E|E{*}E|(E)|id$;

   iv) Disambiguate the grammar given in section (iii).

(b) b) Prove or disprove the following statement: "The language $L = \{w \in \{a,b\}^*; w = a^n b^j : |n-j| = 2\}$, is not a regular language."

(c) Show, using a "sketch" of the pumping lemma, that the following statement is true: "$L = \{y^n x x^R y^n$ where $x \in \{a,b\}^*$ and $y \in \{a,b\}\}$ is NOT a context free language".

(d) Show a context-free grammar and an accepting PDA for the following language: $L = \{w : w = yxx^R$ and $x, y \in \{a,b\}^*\}$

4. CS 5329 Algorithm Design and Analysis

   Given the quick sort total running time as

$$T(n) = \max_{1 \le q \le n-1} (T(q) + T(n-q)) + \theta(n)$$

   prove the total running time for this quick sort is always $T(n) \le cn^2$ for $c \ge c_0$.

5. CS 5329 Algorithm Design and Analysis

   Give definition and an example of a double hashing scheme for building a hash table. Insert entries with keys $1, 21, 4, 13$ and $34$ in your table. Draw final table and show calculations.

**Department of Computer Science**
**Core Graduate Comprehensive Exam**
**Spring 2009**

- **Answer the questions on the paper supplied.**
- **Answer question 1. Answer question 2 . Answer question 3 or 4 or 5 or 6. You should answer a total of three questions.**
- **Start each question on a new page. Write on only one side of the paper.**
- **Write your SIX-DIGIT Texas State ID in the top right corner of each page of your answer. Do NOT put your name anywhere on the answers.**
- **Put the number of the question being answered in the top left corner of each answer page.**
- **If the answer to a question is written on more than one page, number the pages consecutively.**

1. CS5346 Advanced Artificial Intelligence

   (a) Explain Hill Climbing (with and without back tracking) in detail with examples and compare it with the A* algorithm.

   (b) A database of a county bank's loan department is represented as positive and negative instances in the following table:

| Individual | RATING | BAL | APP | INC | OK |
|---|---|---|---|---|---|
| 1 | 1 | 0 | 0 | 1 | 0 |
| 2 | 0 | 0 | 1 | 0 | 0 |
| 3 | 1 | 1 | 0 | 1 | 1 |
| 4 | 0 | 1 | 1 | 1 | 1 |
| 5 | 0 | 1 | 1 | 0 | 0 |
| 6 | 1 | 1 | 1 | 0 | 1 |
| 7 | 1 | 1 | 1 | 1 | 1 |
| 8 | 1 | 0 | 1 | 0 | 0 |
| 9 | 1 | 1 | 0 | 0 | 0 |

   Derive a set of rules which cover all and only the positive instances.

2. CS5329 Algorithm Design and Analysis

   Show that running time of the Merge Sort for n items is $O(n \lg n)$. Support your answer by drawing a recursion tree or a comparison tree.

3. CS5306 Advanced Operating Systems

   (a) Describe wait-die and wound-wait deadlock prevention algorithms.

   (b) Point out the main difference between the two.

4. CS5310 Network and Communication Systems

   (a) What is the difference between congestion control and flow control? How does TCP implement congestion control?

   (b) Consider Intra-domain routing within a large Autonomous System. Explain how do routers build routes and how these routes are used to forward packets.

   (c) Consider a link of 10 Mbps capacity. What is the throughput achieved by a Stop and Wait protocol sending packets that are 1KB in size, given that the round-trip time is 50msec? What percentage of bandwidth capacity is wasted due to this protocol? Assume no packet/Ack loss. Your answer should be in Mbps for the first part of the question.

5. CS5332 Data Base Theory and Design

   Consider the following typed relational schema describing books and borrowers in a library:

   Book(<u>isbn:integer</u>, title:string, publisher:string, year:integer)
   Author(<u>isbn:integer</u>, <u>name:string</u>, rank:integer)
   Borrower(<u>bid:integer</u>, name:string, address:string)
   Borrowings(<u>isbn:integer</u>, <u>borrower:integer</u>, whenTaken:date, whenReturned:date)

   Assume the following:

   - the fields that are underlined are the primary keys of the relations

- there is a referential integrity constraint between ”borrower” in relation Borrowings and ”bid” in relation Borrower
- there are also referential integrity constraints between ”isbn” in relation Book and ”isbn” in relations Author and Borrowings
- dates are represented as a count of the number of days since Jan 1 2009
- there is a system function `today` that returns today's `date`
- the relational operators $<$ and $>$ allow you test *before* and *after* on `dates`
- all loans are for a period of 14 days
- the `whenReturned` field is `null` until the book is returned
- the `rank` field specifies whether the author is the first, second, ... author

Implement solutions to the following:

(a) *(4 marks)* Write an SQL query to show all the books (`title` only) written by a particular author called “Ullman”.

(b) *(4 marks)* Write an SQL query to show which books (`title` only) have multiple authors.

(c) *(4 marks)* Write an SQL query to show which books (`title` only) have never been borrowed.

(d) *(4 marks)* Write an SQL query to show which books (`title` only) are currently out on loan.

(e) *(4 marks)* Write an SQL query to list overdue books (`title`) and the borrower's name (Hints: you need to join relations Book, Borrowers and Borrowings).

6. CS5332 Data Base Theory and Design

Given the following Wellmeadows Hospital Form, provide a final E/R model for this form. You need to provide Pks, Fks, and Relationships as well as all attributes in this E/R model. You must normalize tables up to 3NF. If you need to, you may modify the data or the form for this Form.

| Patient Number | Surgeon License Number | Date of Surgery | Patient Name | Patient Address | Surgeon Name | Surgery | Postoperative Drug Administered | Side Effect of Drug |
|---|---|---|---|---|---|---|---|---|
| 111 | 145 | Jan.1,1985 | John White | 15 New Street New York, N.Y. | Seth Little | Gallstones removal | Penicillin | Rash |
| | 311 | June 12,1985 | | | Michael Diamond | Kidney stones removal | — | — |
| 1234 | 243 | Apr. 5, 1984 | Mary Jones | 10 Main Street Rye, N.Y. | Charles Field | Eye cataract removal | Tetracycline | Fever |
| | 467 | May 10, 1985 | | | Patricia Gold | Thrombosis removal | — | — |
| 2345 | 189 | Jan. 8,1986 | Charles Brown | Dogwood Lane Harrison,N.Y. | David Rosen | Open heart surgery | Cephalosporin | — |
| 4878 | 145 | Nov.5,1985 | Hal Kane | 55 Boston Post Road, Chester, Conn. | Beth Little | Cholecystectomy | Demicillin | — |
| 5123 | 145 | May 10,1985 | Paul Kosher | Blind Brook Mamaroneck, N.Y. | Beth Little | Gallstones removal | — | — |
| 6845 | 243 | Apr.5,1984 | Ann Hood | Hilton Road Larchmont,N.Y. | Charles Field | Eye cornea replacement | Tetracycline | Fever |
| | | Dec.15,1984 | | | | Eye cataract | — | — |

**Department of Computer Science**
**Core Graduate Comprehensive Exam**
**Fall 2009**

- **Answer the questions on the paper supplied.**
- **Answer question 1 or answer question 2. Answer question 3 or answer question 4. Answer question 5 or answer question 6. You should answer a total of three questions.**
- **Start each question on a new page. Write on only one side of the paper.**
- **Write your SIX-DIGIT Texas State ID in the top right corner of each page of your answer. Do NOT put your name anywhere on the answers.**
- **Put the number of the question being answered in the top left corner of each answer page.**
- **If the answer to a question is written on more than one page, number the pages consecutively.**

1. CS5329 Algorithm Design and Analysis

   (a) Use recursion tree in best case partition to describe the total Running time of Quick Sort $T(n) = \theta(nlgn)$.

2. CS5329 Algorithm Design and Analysis

   (a) Write the definition of the "Big-O" notation.

   (b) Prove that the function $f(n) = 98n^2 - 1.2$ is $O(n^2)$ by computing the constants $C$ and $n_o$.

3. CS5391 Survey of Software Engineering

   (a) Draw a representation of the Extended V-Model.

   (b) What does this model illustrate?

4. CS5391 Survey of Software Engineering

   Describe the following process models and point out their advantages and disadvantages. For what kind of projects are they good for?

   (a) evolutionary model

   (b) spiral model

5. CS5318 Design of Programming Languages.

   (a) Computational Paradigms. Discuss one common computational paradigms other than the object-oriented paradigm. A complete answer will include a computational model, data and control structures, and an example language.

   (b) Context Free Grammars.

| | | |
|---|---|---|
| expr | → | expr + term \| term |
| term | → | term * factor \| factor |
| factor | → | (expr) \| number |
| number | → | number digit \| digit |
| digit | → | 0 \| 1 \| 2 \| 3 \| 4 \| 5 \| 6 \| 7 \| 8 \| 9 |

   Using the given grammar, draw a parse tree for the arithmetic expression:

$$3*2+5+6*4$$

   (c) Attribute Grammars.

| | Grammar Rule | Semantic Rules |
|---|---|---|
| 1. | $<bin-num> \rightarrow <bin-dig>_1 \ . \ <bin-dig>_2$ | $val(<bin-num>) = val(<bin-dig>_1)$ $+val(<bin-dig>_2)/2**len(<bin-dig>_2)$ |
| 2. | $<bin-dig>_1 \rightarrow <bin-dig>_2 <bit>$ | $val(<bin-dig>_1) = 2*val(<bin-dig>_2)$ $+val(<bit>)$ $len(<bin-dig>_1) = len(<bin-dig>_2)+1$ |
| 3. | $<bin-dig> \rightarrow <bit>$ | $val(<bin-dig>) = val(<bit>)$ $len(<bin-dig>) = 1$ |
| 4. | $<bit> \rightarrow 0$ $<bit> \rightarrow 1$ | $val(<bit>) = 0$ $val(<bit>) = 1$ |

For the given attribute grammar,

i. Identify the attributes in the grammar.
ii. For each grammar symbol that has an attribute, and tell if its attributes are inherited or synthesized.
iii. Using the given attribute grammar, construct and decorate a parse tree for the string

```
10.101
```

6. CS5338 Formal Languages

(a) Answer all the parts of this question

i. Arrange the following languages according to their "minimal place" in the Chomsky hierarchy. Prove your results using formal methods for evaluating whether a language is regular, context free, or context sensitive

A. $L = \{a^i b^j : i, j \geq 0 \text{ and } i \neq j\}$.
B. $L = \{w \in \{a,b\}^* : w = (a \cup b)^* a^n b^n (a \cup b)^*, n \geq 0\}$
C. $L = \{w \in \{a,b,c,d\}^* : \#_b(w) \geq \#_c(w) \geq \#_d(w) \geq 0\}$

ii. Prove or disprove the following assertion: "The intersection of a regular language and a non-regular language must be regular."

iii. Show a context-free grammar and an accepting PDA for the following language: $L = \{wcu \text{ where } w \neq u^R : w, u \in \{a,b\}^*\}$

**Department of Computer Science**
**Core Graduate Comprehensive Exam**
**Spring 2010**

- **Answer the questions on the paper supplied.**
- **Answer question 1 or 2 . Answer question 3 or 4. Answer question 5 or 6 or 7. You should answer a total of three questions.**
- **Start each question on a new page. Write on only one side of the paper.**
- **Write your SIX-DIGIT Texas State ID in the top right corner of each page of your answer. Do NOT put your name anywhere on the answers.**
- **Put the number of the question being answered in the top left corner of each answer page.**
- **If the answer to a question is written on more than one page, number the pages consecutively.**

1. 5329 Algorithm Design and Analysis

   Provide a pseudocode for the Selection Sort algorithm. Analyze the running time performance of your implementation of the Selection Sort in terms of Big-Oh notation separately for "assignments" and "comparisons" for the best case scenario.

2. 5329 Algorithm Design and Analysis

   (a) Write Quick Sort algorithm and that of its Partition function.

   (b) Use recursion tree in best case partition to describe the total running time of Quick Sort: $T(n) = \theta(n \lg n)$

3. 5391 Survey of Software Engineering

   Discuss the concepts of Coupling, Cohesion and information hiding. Give specific attention to how they affect software maintenance.

4. 5391 Survey of Software Engineering

   Explain the following software process models.

   (a) incremental

   (b) component-based

5. 5310 Network and Communication Systems

   (a) Is Internet a circuit-switched network or packet-switched network? Briefly justify your answer.

   (b) What does the term "high-speed network" mean in the context of computer networks and data communications?

   (c) What is I/O multiplexing? Describe briefly all possible alternatives to I/O multiplexing.

6. 5332 Data Base Theory and Design

   Consider the following typed relational schema describing books and borrowers in a library:

   Book(<u>isbn:integer</u>, title:string, publisher:string, year:integer)
   Author(<u>isbn:integer</u>, <u>name:string</u>, rank:integer)
   Borrower(<u>bid:integer</u>, name:string, address:string)
   Borrowings(<u>isbn:integer</u>, <u>borrower:integer</u>, whenTaken:date, whenReturned:date)

   Assume the following:

   - the fields that are underlined are the primary keys of the relations
   - there is a referential integrity constraint between "borrower" in relation Borrowings and "bid" in relation Borrower
   - there are also referential integrity constraints between "isbn" in relation Book and "isbn" in relations Author and Borrowings
   - dates are represented as a count of the number of days since Jan 1 2009
   - there is a system function `today` that returns today's `date`
   - the relational operators $<$ and $>$ allow you test *before* and *after* on `dates`
   - all loans are for a period of 14 days
   - the `whenReturned` field is `null` until the book is returned
   - the `rank` field specifies whether the author is the first, second, ... author

   Implement solutions to the following:

   (a) *(4 marks)* Write an SQL query to show all the books (`title` only) written by a particular author called "Ullman".

(b) *(4 marks)* Write an SQL query to show which books (`title` only) have multiple authors.

(c) *(4 marks)* Write an SQL query to show which books (`title` only) have never been borrowed.

(d) *(4 marks)* Write an SQL query to show which books (`title` only) are currently out on loan.

(e) *(4 marks)* Write an SQL query to list overdue books (`title`) and the borrower's name

7. 5332 Data Base Theory and Design

   Draw an extended ER model for the following business rules: (The attributes of each table and all tables must be within the ER diagram. You may create more attributes to complete the problem. You must normalize the entities, and specify pk's and fk's for each table within the ER diagram.)

   Business Problem Rules: Design a database for Ray. He is interested in movies and wants to keep information on movies, actors, and directors in a database. The only user is Ray and he needs to produce the following reports:

   (a) For each director, list his or her number, name, and the year he or she was born. If the director is deceased, list the year of death.

   (b) For each movie, list its number, title, the year the movie was made, and its type (for example, Comedy, Drama, Science Fiction and so on).

   (c) For each movie, list its number, title, the number and name of its director, the critic's rating, the MPAA rating (G, PG, PG-13, or R), the number of awards for which the movie was nominated, and the number of awards the move won. (The critics rate the movie with a number of "stars". Four stars is the top rating possible. Zero stars is the worst possible rating.)

   (d) For each actor/actress, list his or her number, name, birthplace, and the year he or she was born. If the actor is deceased, list the year of death.

   (e) For each movie, list its number and title, along with the number and name of the actors who appeared in it.

   (f) For each actor/actress, list his or her number and name, along with the number and name of the movies in which the actor starred.

**Department of Computer Science**
**Core Graduate Comprehensive Exam**
**Fall 2010**

- **Answer the questions on the paper supplied.**
- **Answer question 1, Answer question 2 or answer question 3, Answer question 4 or answer question 5. You should answer a total of three questions.**
- **Start each question on a new page. Write on only one side of the paper.**
- **Write your SIX-DIGIT Texas State ID in the top right corner of each page of your answer. Do NOT put your name anywhere on the answers.**
- **Put the number of the question being answered in the top left corner of each answer page.**
- **If the answer to a question is written on more than one page, number the pages consecutively.**

1. CS5346 Advanced Artificial Intelligence

   (a) Explain (in detail with examples) how Expert Systems use Forward Chaining and Backward Chaining methodologies in making decisions. Also compare advantages and disadvantages of these two chaining methodologies.

   (b) Consider the Wumpus environment E as a $4 \times 4$ grid of rooms (E2,3 represents 2nd row and 3rd column). Sensors have detected the following facts: S1,3; S2,2; S2,3; S2,4; S3,3. Where Si,j indicates stench in the i th row and j th column. Stench is not detected in any other room.

   Write rules in propositional logic to determine the position of the Wumpus. Using resolution rule, develop a proof tree to determine the location of the Wumpus.

2. CS5329 Algorithm Design and Analysis

   Assuming the Partition function for Quicksort algorithm always produce a 9-to-1 split, draw a recurrence tree and calculate the total running time for Quicksort.

3. CS5329 Algorithm Design and Analysis

   (a) Write the definition of the "Big-O" notation. Assuming that you are given an algorithm containing $T(n) = 2n + 10n^3 + 10^{11}$ primitive operations, estimate the running time of this algorithm using the definition you provided.

4. CS5318 Design of Programming Languages.

   (a) Discuss in detail and using examples the role of abstraction in programming language design.

   (b) Context Free Grammars.

| | | |
|---|---|---|
| expr | → | expr + term \| term |
| term | → | factor * term \| factor |
| factor | → | (expr) \| number |
| number | → | number digit \| digit |
| digit | → | 0 \| 1 \| 2 \| 3 \| 4 \| 5 \| 6 \| 7 \| 8 \| 9 |

   Using the given grammar, draw a parse tree for the arithmetic expression given below, and compute its value:

$$3*4+5+6*2$$

   (c) Attribute Grammars.

| | Grammar Rule | Semantic Rules |
|---|---|---|
| 1. | $<bin-num> \rightarrow <bin-dig>_1 \ . \ <bin-dig>_2$ | $val(<bin-num>) = val(<bin-dig>_1)$<br>$+val(<bin-dig>_2)/2**len(<bin-dig>_2)$ |
| 2. | $<bin-dig>_1 \rightarrow <bin-dig>_2 <bit>$ | $val(<bin-dig>_1) = 2*val(<bin-dig>_2)$<br>$+val(<bit>)$<br>$len(<bin-dig>_1) = len(<bin-dig>_2)+1$ |
| 3. | $<bin-dig> \rightarrow <bit>$ | $val(<bin-dig>) = val(<bit>)$<br>$len(<bin-dig>) = 1$ |
| 4. | $<bit> \rightarrow 0$<br>$<bit> \rightarrow 1$ | $val(<bit>) = 0$<br>$val(<bit>) = 1$ |

   For the given attribute grammar,

   i. Identify the attributes in the grammar.

ii. For each grammar symbol that has an attribute, and tell if its attributes are inherited or synthesized.

iii. Using the given attribute grammar, construct and decorate a parse tree for the string

`10.001`

5. CS5338 Formal Languages Answer the following True or False questions. Some terms used in the questions are listed below:

DFSM: Deterministic Finite State Machine

NDFSM: Non-deterministic Finite State Machine

DPDA: Deterministic Pushdown Automata

NDPDA: Non-deterministic Pushdown Automata

DTM: Deterministic Turing Machine

NDTM: Non-deterministic Turing Machine

D: Decidable

SD: Semi-decidable

(a) Any DFSM halts in at most $|w|$ steps.

(b) Any NDFSM halts in at most $|w|$ steps.

(c) For any NDFSM M we can find a DFSM M', such that $L(M) = L(M')$.

(d) Let M be an NDFSM. If M only has one ε-transition, then M must halt.

(e) Any DPDA always halts in at most $|w|$ steps.

(f) Any NDPDA always halts in at most $|w|$ steps.

(g) For any NDPDA M, we can find an DPDA M', such that $L(M) = L(M')$.

(h) For any NDPDA M, we can find an NDPDA M' that halts in at most $|w|$ steps, such that $L(M) = L(M')$.

(i) Let M be an NDFSM. We can find some DPDA M' that decides $L(M)$.

(j) Any regular language or context-free language can be decided by some TM.

(k) NDTM may not halt but DTM must halt.

(l) For any NDTM M, we can find an DTM that halts and $L(M) = L(M')$.

(m) Let $ndtime(f(n))$ be the set of languages that can be decided by some nondeterministic Turing machine in time $O(f(n))$. Every language in $ndtime(2^n)$ is decidable.

(n) If both L and $\neg L$ are in SD, then L is in D.

(o) NP-complete is a superset of NP-hard.

(p) If P = NP, then NP-complete languages will be in P as well.

(q) If we can verify a certificate of L, then L is in NP.

(r) If L is NP-hard, L must be in NP.

(s) The union of a finite number of countably infinite sets is countable.

(t) The union of a countably infinite number of countably infinite sets is countable.

(u) Regular languages and context-free languages are in D but not in P.

(v) If L = L1 ∩ L2 and L is regular, then L1 and L2 must be regular.

(w) If L is regular, then $\neg L$ is also regular.

(x) $\neg A^n B^n C^n$ is context-free, but $A^n B^n C^n$ is not. Thus context-free languages are not closed under complement.

(y) Church-Turing thesis implies that we need computation models that are more powerful than Turing machines for those problems in $\neg SD$.

**Department of Computer Science**
**Core Graduate Comprehensive Exam**
**Spring 2011**

- **Answer the questions on the paper supplied.**
- **Answer question 1. Answer question 2 or answer question 3. Answer question 4 or answer question 5 or answer question 6 or answer question 7. You should answer a total of three questions.**
- **Start each question on a new page. Write on only one side of the paper.**
- **Write your SIX-DIGIT Texas State ID in the top right corner of each page of your answer. Do NOT put your name anywhere on the answers.**
- **Put the number of the question being answered in the top left corner of each answer page.**
- **If the answer to a question is written on more than one page, number the pages consecutively.**

1. CS 5346 Advanced Artificial Intelligence

   (a) Explain consistency. Also, prove that if $h(n)$ is consistent, the values of $f(n)$ along any path are nondecreasing.

   (b) Trace the MIN MAX algorithm (shown in Figure #2) on the example shown in Figure #1. MAX makes the first move at the root node A. Evaluation function values are computed from MAX's point of view.

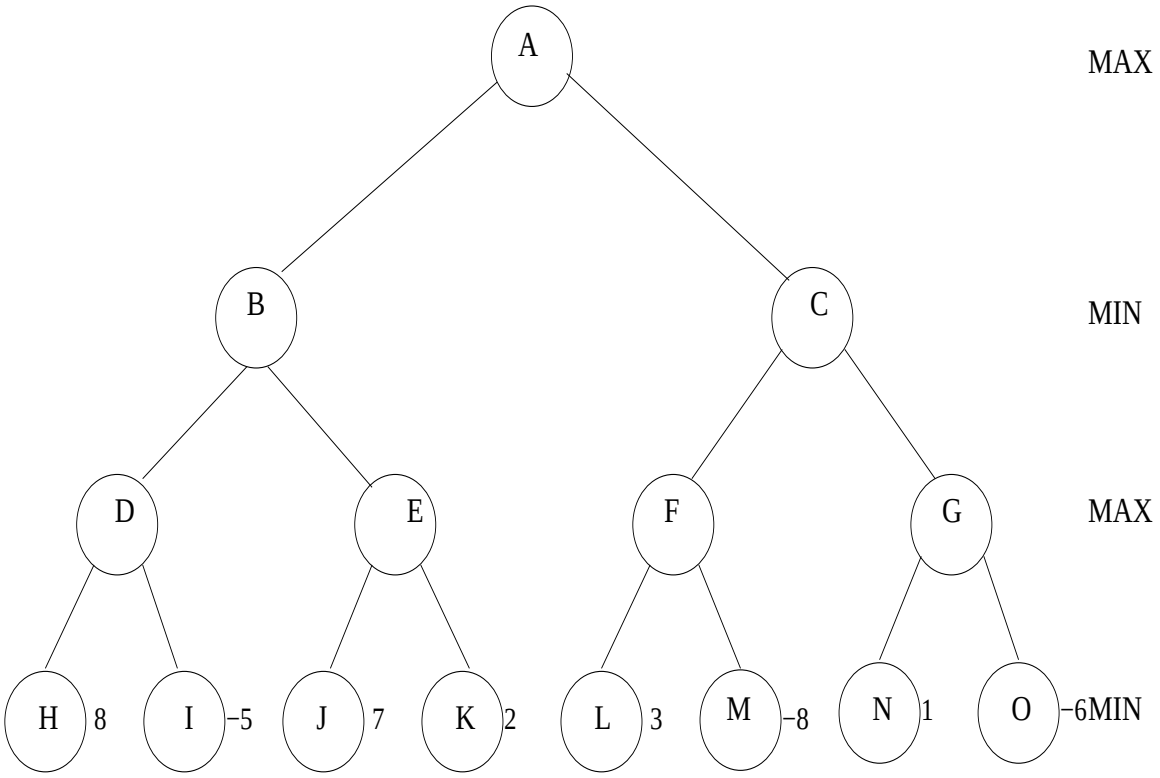



Figure 1: Figure #1

MINIMAX (CURRENT, 0, PLAYER-ONE)

if PLAYER-ONE is to move, or

MINIMAX (CURRENT, 0, PLAYER-TWO)

if PLAYER-TWO is to move.

***Algorithm: MINIMAX (Position, Depth, Player)***

1. If *DEEP-ENOUGH (Position, Depth), then return the structure*

VALUE = STATIC (*Position*, *Player*);

PATH = nil

This indicates that there is no path from this node and that its value is that determined by the static evaluation function.

2. Otherwise, generate one more ply of the tree by calling the function MOVE-GEN (Position Player) and setting SUCCESSORS to the list it returns.

3. If SUCCESSORS is empty, then there are no moves to be made, so return the same structure that would have been returned if DEEP-ENOUGH had returned true.

4. If SUCCESSORS is not empty, then examine each element in turn and keep track or the best one. This is done as follows:

Initialize BEST-SCORE to the minimum value that STATIC can return. It will be updated to reflect the best score that can be achieved by an element of SUCCESSORS.

For each element SUCC of SUCCESSORS, do the following:

(a) Set RESULT-SUCC to MINIMAX (SUCC, *Depth + I*, *OPPOSITE (Player)*)

This recursive call to MINIMAX will actually carry out the exploration of SUCCC.

(b) Set NEW-VALUE to - VALUE (RESULT-SUCC). This will cause it to reflect the merits of the position from the opposite perspective from that of the next lower Ievel.

(c) If NEW-VALUE > BEST-SCORE, then we have found a successor that is better than any that have been examined so far. Record this by doing the following:

(i) Set BEST-SCORE to NEW-VALUE.

(ii) The best known path is now from CURRENT to SUCC and then on to the appropriate path down from SUCC as determined by the recursive call to MINIMAX. So set BEST-PATH to the result of attaching SUCC to the front of PATH(RESULT-SUCC).

5. Now that all the successors have been examined, we know the value of Position as well as which path to take from it. So return the structure

VALUE = BEST-SCORE

PATH = BEST-PATH

**Figure #2**

2. CS 5391 Survey of Software Engineering
   (a) Draw a representation of the Extended V-Model.
   (b) What is the contribution of this model to software process and software quality?
3. CS 5391 Survey of Software Engineering Explain the following software process models:
   (a) Incremental development model
   (b) Component-based development model

   In the above process models, how do we choose which one to use?
4. CS5306 Advanced Operating Systems

   Consider three different file protection mechanisms: capabilities, access control lists, and the Unix rwx bits. Explain how these mechanisms work and discuss which of these mechanisms can be used, why, and how for each of the following protection problems:
   (a) Ken wants his files readable by everyone except this office mate.
   (b) Mitch and Steve want to share some secret files.
   (c) Linda wants some of her files to be public.
   For Unix, assume that groups are categories such as faculty, students, staff, etc.
5. CS 5310 Network and Communication Systems

   Describe all possible problems in signal transmissions and concisely explain the reason of each problem.
6. CS5332 Data Base Theory and Design

   1. Given the following fields in a business table:

   Project Number

   Project Name

   Employee Number

   Employee Name

   Employee Job Class

   Employee Charge Fee Per Hour based on the Job Class

   Total Hours worked

(a) Specify all functional dependencies of all these fields in this table.

(b) Draw an E/R Model for this problem after normalization up to 3NF.

You must specify the primary keys, foreign keys, and relationships with all required attributes. You must also use crow feet notation for E/R diagram.

7. CS5332 Data Base Theory and Design

The customer database consists of the relations defined in the four tables shown below:

Customer(<u>custNo</u>, custName, state, phone)
Item(<u>itemNo</u>, ItemName, itemPrice, qtyOnHand)
Invoice(<u>invNo</u>, inDate, *custNo*)
Invitem(<u>*invNo, itemNo*</u>, qty)

Primary keys are underlined. Foreign keys are shown in italic font. Primary keys that are also foreign keys are shown in underline italic font. For example, the invNo and itemNo attributes in Invitem relation is the primary key for Invitem and is also foreign keys of tables Invoice and Item. Below are sample data from each of the above relation.

```
Customer table
--------------
custNo   custName    state     phone
211       Garcia      TX       732-555-1000
212       Parkin      NY       212-555-2000
225       Ellen       NJ       973-555-3333
239       Bayer       FL       401-555-7777

Item table
----------
itemNo     itemName   itemPrice     qtyOnHand
 1          Screw      5.00           50
 2          Nut        2.25           110
 3          Bolt       3.99           75
 4          Hammer     10.00          125
 5          Washer     6.00           300
 6          Nail       0.99           400

Invoice table
-------------
invNo      invDate        custNo
1001        05-Sep-07      212
1002        17-Sep-07      225
1003        17-Sep-07      239
1004        18-Sep-07      211
1005        21-Sep-07      212

invItem table
-------------
invNo        itemNo           qty
1001          1                5
1001          3                5
1001          5                9
1002          1                2
1002          2                3
1003          1                7
1003          2                1
1004          4                5
1005          4                10
```

Write SQL to answer the following queries:

(a) Find items which have the itemPrice between $5 and $10?

(b) Count the number of items ordered in each invoice.

(c) Find customers who are not from “NY” or “NJ”

(d) Find items that are cheaper than nut. You must display the itemName and the associated itemPrice. Beware that nut can be written in both upper and lower case.

(e) Find the invoices in which three or more items are ordered.

(f) List the invoices, customer names and the items purchased by them. The result should display invoices, customer names and item names together ordered by the customer name.

**Department of Computer Science**
**Core Graduate Comprehensive Exam**
**Fall 2011**

- **Answer the questions on the paper supplied.**
- **Answer question 1. Answer question 2 or answer question 3. Answer question 4 or answer question 5 or answer question 6 or answer question 7. You should answer a total of three questions.**
- **Start each question on a new page. Write on only one side of the paper.**
- **Write your SIX-DIGIT Texas State ID in the top right corner of each page of your answer. Do NOT put your name anywhere on the answers.**
- **Put the number of the question being answered in the top left corner of each answer page.**
- **If the answer to a question is written on more than one page, number the pages consecutively.**

1. **CS 5329 Algorithm Design and Analysis**

   Draw a recursion tree for the merge-sort algorithm. Prove the total running time is $O(nlgn)$ (You must show the calculation to get the result.)

2. **CS 5318 Design of Programming Languages**

   The following BNF specification defines the language of Roman numerals less than 1000:

$$
\begin{aligned}
&< \text{roman} > ::= < \text{hundreds} >< \text{tens} >< \text{units} > \\
&< \text{hundreds} > ::= < \text{low hundreds} > | \mathbf{CD} | \mathbf{D} < \text{low hundreds} > | \mathbf{CM} \\
&< \text{low hundreds} > ::= e | < \text{low hundreds} > \mathbf{C} \\
&< \text{tens} > ::= < \text{low tens} > | \mathbf{XL} | \mathbf{L} < \text{low tens} > | \mathbf{XC} \\
&< \text{low tens} > ::= e | < \text{low tens} > \mathbf{X} \\
&< \text{units} > ::= < \text{low units} > | \mathbf{IV} | \mathbf{V} < \text{low units} > | \mathbf{IX} \\
&< \text{low units} > ::= e | < \text{low units} > \mathbf{I}
\end{aligned}
$$

   Define attributes for this grammar to carry out two tasks:

   (a) Restrict the number of X's in <low tens>, the I's in <low units>, and the C's in <low hundreds> to no more than three.

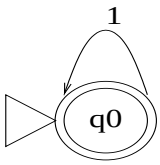
   (b) Provide an attribute for <roman> that gives the decimal value of the Roman numeral being defined.

   (c) Decorate the parse tree for **CMVIII** to show how its decimal value is computed.

   Note that you may define any other attributes needed for these tasks, but do not change the BNF grammar.

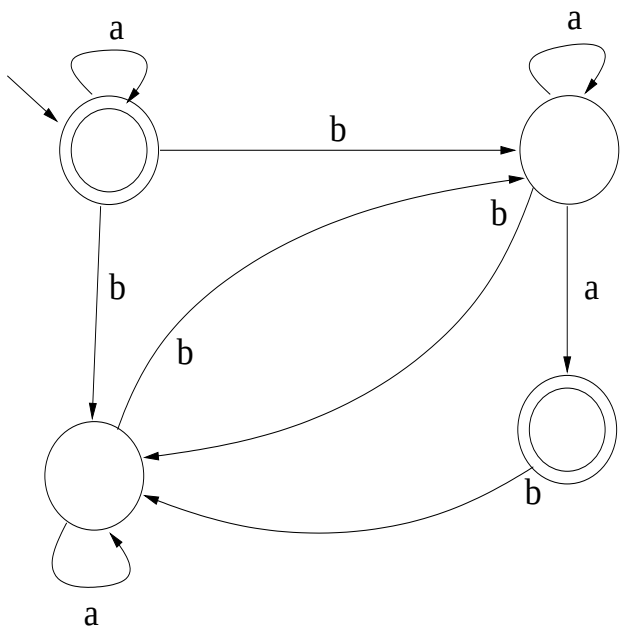
3. **CS 5338 Formal Languages**

   (a) Build a deterministic FSM (finite state machine) for each of the following languages:
   Example: $\{w \in \{0,1\}^* : \text{none of the prefixes of } w \text{ ends in } 0\}$




   i. $\{w \in \{0,1\}^* : w \text{ has 001 as a substring}\}$
   ii. $\{w \in \{0,1\}^* : w \text{ does not have 001 as a substring}\}$
   iii. $\{w \in \{a,b\}^* : \text{every } a \text{ in } w \text{ is immediately preceded and followed by } b\}$

   (b) Consider the following NDFSM (non-deterministic finite state machine) $M$:




For each of the following strings $w$, determine whether $w \in L(M)$:

i. *aabbba*
ii. *bab*
iii. *baba*

4. **CS5306 Advanced Operating Systems**

Your OS has a set of queues, each of which is protected by a lock. To enqueue or dequeue an item, a thread must hold the lock associated with the queue.

You need to implement a transfer routine that dequeues an item from one queue and enqueues it on another, and the transfer must appear to occur atomically. This is your first attempt:

```
void transfer(Queue *queue1, Queue *queue2)
{
  Item thing; /* the thing being transferred */

  queue1→lock.Acquire();
  thing = queue1→Dequeue();
  if (thing ! = NULL ) {
      queue2→lock.Acquire();
      queue2→Enqueue(thing);
      queue2→lock.Release();
  }
  queue1→lock.Release();
}
```

You may assume that queue1 and queue2 never refer to the same queue and that you have a function Queue::Address() which takes a queue and returns, as an unsigned integer, its address.

(a) Explain how using this implementation of transfer() can lead to deadlock.

(b) Write a modified version of transfer() that avoids deadlock and does the transfer atomically.

(c) If the transfer does not need to be atomic, how might you change your solution to achieve a higher degree of concurrency? Justify why your modification increases concurrency.

5. **CS 5310 Network and Communication Systems**

   Answer the following questions based on the TCP/IP model:

   (a) What are the main differences between TCP and UDP based on congestion control?

   (b) What is the purpose of the transport layer and why it is referred to as End-to-End?

   (c) Explain briefly how IP packets are forwarded by routers.

6. **CS5332 Data Base Theory and Design**

   Consider the following employee database, where the primary keys are underlined.

   Employee(ename:string, street:string, city:string)
   Works(*employee:string, company:string*, salary:real)
   Company(cname:string, city:string)
   Manages(*employee:string, company:string*)

   Give a single SQL statement for each of the following queries:

   (a) *(4 marks)* Find the names, street, and cities of residence of all employees who work for the "First Bank Corporation" and earn more than $40,000.

   (b) *(4 marks)* Find the names of all employees in the database who live in the same cities as the companies for which they work.

   (c) *(4 marks)* Give all managers of "First Bank Corporation" a 10 percent salary raise.

   (d) *(4 marks)* Find the names of all employees in the database who earn more than every employee of "Small Bank Corporation".

   (e) *(7 marks)* Find those companies whose employees earn a higher salary, on average, than the average salary at "First Bank Corporation".

7. **CS5332 Data Base Theory and Design**

   The following is a table for patient, dentist and surgery appointment. Indicate the functional dependencies on the table and design an E/R model with PK's, FK's, and Relationships up to 3NF.

| staffNo | dentistName | patNo | patName | appointment date | appointment time | surgeryNo |
|---|---|---|---|---|---|---|
| S1011 | Tony Smith | P100 | Gillian White | 12-Sep-08 | 10.00 | S15 |
| S1011 | Tony Smith | P105 | Jill Bell | 12-Sep-08 | 12.00 | S15 |
| S1024 | Helen Pearson | P108 | Ian MacKay | 12-Sep-08 | 10.00 | S10 |
| S1024 | Helen Pearson | P108 | Ian MacKay | 14-Sep-08 | 14.00 | S10 |
| S1032 | Robin Plevin | P105 | Jill Bell | 14-Sep-08 | 16.30 | S15 |
| S1032 | Robin Plevin | P110 | John Walker | 15-Sep-08 | 18.00 | S13 |

**Department of Computer Science**
**Core Graduate Comprehensive Exam**
**Spring 2012**

- **Answer the questions on the paper supplied.**
- **Answer question 1. Answer question 2. Answer question 3 or answer question 4. You should answer a total of three questions.**
- **Start each question on a new page. Write on only one side of the paper.**
- **Write your SIX-DIGIT Texas State ID in the top right corner of each page of your answer. Do NOT put your name anywhere on the answers.**
- **Put the number of the question being answered in the top left corner of each answer page.**
- **If the answer to a question is written on more than one page, number the pages consecutively.**

1. **CS 5329 Algorithm Design and Analysis**

   Given a constant fraction $p$ ( $p = 0.2$ ), and assuming we always have $p$ and

   $(1-p)$ proportional partitions for the Quick-Sort algorithm,

   Prove that the total running time for such Quick-Sort is still $\Theta$ ( $nlgn$ ).

   (Hint: you may use recursion tree and some log algebra.)

2. **CS 5391 Survey of Software Engineering**

   Describe the following process models and their application scope.

   (a) incremental model

   (b) prototyping model

3. **CS 5318 Design of Programming Languages**

   Given the following BNF:

   $< expr > ::= (< list >) | a$
   $< list > ::= < list >, < expr > \mid < expr >$

   (a) Draw the parse tree for $((a,a),a,(a))$.
   (b) Design appropriate attribute(s) and write an attribute grammar to compute the level of parentheses. For example, the level of parentheses of the sentence in (a) is 2.
   (c) Decorate the parse tree in (a) with your attributes defined in (b) to show how the level of parentheses is computed.
   (d) Write EBNF rules and/or syntax diagrams for the language.
   (e) Write a recursive-descent parser for the language based on your EBNF rules in (d).

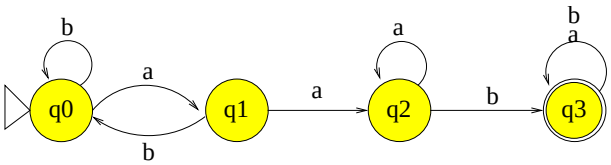
4. **CS 5338 Formal Languages**

   (a) Ture or False questions. Example: This is Spring 2012. (True)

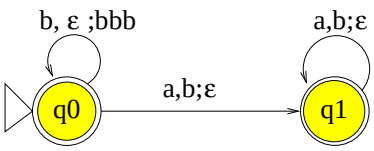
      i. The union of a finite number of countably infinite sets is countable. ( )
      ii. The union of a countably infinite number of countably infinite sets is countable. ( )
      iii. Any DFSM halts in at most $|w|$ steps. ( )
      iv. Any NDFSM halts in at most $|w|$ steps. ( )
      v. Any DPDA always halts in at most $|w|$ steps. ( )
      vi. Any NDPDA always halts in at most $|w|$ steps. ( )
      vii. Church-Turing thesis implies that we need computation models that are more powerful than Turing machines for those problems in $\neg SD$. ( )
      viii. Let $ndtime(f(n))$ be the set of languages that can be decided by some nondeterministic Turing machine in time $O(f(n))$. Every language in $ndtime(2^n)$ is decidable. ( )
      ix. $NP$-hard is a subset of $NP$-complete. ( )
      x. If $P = NP$, then $NP$-complete languages will be in $P$ as well. ( )

   (b) Short questions.

i. Given the following FSM M, write down three strings that are in $L(M)$.




ii. Given the following PDA M, write down three strings that are in $L(M)$.



**Department of Computer Science**
**Core Graduate Comprehensive Exam**
**Fall 2012**

- **Answer the questions on the paper supplied.**
- **Answer question 1 or Answer question 2. Answer question 3 or 4. Answer question 5 or 6 or 7. Answer question 8 or 9. You should answer a total of four questions.**
- **Start each question on a new page. Write on only one side of the paper.**
- **Write your SIX-DIGIT Texas State ID in the top right corner of each page of your answer. Do NOT put your name anywhere on the answers.**
- **Put the number of the question being answered in the top left corner of each answer page. Put the CORRECT question number to avoid missing your answer.**
- **If the answer to a question is written on more than one page, number the pages consecutively.**

## Group 1

1. **CS 5329 Algorithm Design and Analysis**

   Use recursion tree and log algebra to find the total running time of the recurrence equation:

   $T(n) = T(n/3) + T(2n/3) + cn$

   (You are required to write the detail algebra for the calculation.)

2. **CS 5329 Algorithm Design and Analysis**

   Provide a pseudocode for the Insertion Sort algorithm. Analyze the running time performance of your implementation of the Insertion Sort in terms of Big-Oh notation separately for "assignments" and "comparisons" for the best-case scenario.

## Group 2

3. **CS 5346 Advanced Artificial Intelligence**

   (a) A database of a county bank's loan department is represented as positive and negative instances in the following table:

| Individual | INC | BAL | APP | RATING | OK |
|---|---|---|---|---|---|
| 1 | 1 | 0 | 0 | 1 | 0 |
| 2 | 0 | 0 | 1 | 0 | 0 |
| 3 | 1 | 1 | 0 | 1 | 1 |
| 4 | 0 | 1 | 1 | 1 | 1 |
| 5 | 0 | 1 | 1 | 0 | 0 |
| 6 | 1 | 1 | 1 | 0 | 1 |
| 7 | 1 | 1 | 1 | 1 | 1 |
| 8 | 1 | 0 | 1 | 0 | 0 |
| 9 | 1 | 1 | 0 | 0 | 0 |

   Derive a set of rules that covers all and only the positive instances.

   (b) (i) An algorithm has the following evaluation function:

   $f(n) = (2-w) * g(n) + w * h(n)$

   For what kind of search does this perform for $w = 0, w = 1, w = 2$? For what values of $w$ is it complete and optimal?

   (ii) Another algorithm uses the following evaluation function:

   $f(n) = (2-W) * [g(n) + W/(2-W) * h(n)]$

   Assuming $h(n)$ is admissible, what value of $W$ will make it behave exactly like $A^*$?

4. **CS 5391 Survey of Software Engineering**

   (a) state the pros and cons of lines of code and function points metrics

   (b) describe incremental development model and its limitation

**Group 3**

5. **CS 5306 Advanced Operating Systems**

   Suppose a machine with 4 physical pages starts running a program (in other words, the physical pages are initially empty). The program references the sequence of virtual pages as follows: A B C D E D C B A E D C B A C E

   (a) Explain each of the following paging algorithms.

   (b) For each paging algorithm, replicate the reference pattern and underline each reference that causes a page fault:

      i. LRU

      ii. FIFO

      iii. Optimum

6. **CS 5310 Network and Communication Systems**

   Define the link utilization of the Idle RQ protocol. Explain using your own words the concepts of link utilization. Is higher link utilization always good? Justify your answer.

7. **CS 5332 Data Base Theory and Design**

   Construct an E/R diagram for a database system for the following problem. You must specify the PKs, FKs, and relationships for each entity. Your relations (entities) must be normalized up to 3NF or BCNF. (Prefer you use crow feet notation for E/R model.)

   The problem: A large organization has several parking lots, which are used by staff. Each parking lot has a unique name, location, capacity, and number of floors (where appropriate). Each parking lot has parking spaces, which are uniquely identified using a space number. Members of staff can request the use of a parking space. Each member of staff has a unique number, name, telephone extension number, and vehicle license number.

**Group 4**

8. **CS 5318 Design of Programming Languages**

   Lambda calculus is the basis of functional programming. Assume the following functions in lambda calculus:

$$square = (\lambda x. * xx)$$
$$twice = (\lambda x. + xx)$$

   (a) Explain the applicative order and normal order reduction strategies.

   (b) Show the steps in the applicative order and normal order reductions of the expression (twice (twice square))

   (c) Compare the applicative order and normal order reduction strategies in terms of reduction efficiency.

9. **CS 5338 Formal Languages**

   $1 \sim 20$ are True or False questions. 21 and 22 are short-answer questions. Some terms used in the questions are listed below:

   DFSM: Deterministic Finite State Machine

   NDFSM: Non-deterministic Finite State Machine

   DPDA: Deterministic Pushdown Automata

   NDPDA: Non-deterministic Pushdown Automata

   DTM: Deterministic Turing Machine

   NDTM: Non-deterministic Turing Machine

   D: Decidable

   SD: Semi-decidable

   $w$: An input string

   $L$: A language

   $M$: A machine

(a) A DFSM $M$ must halt for any $w$.

(b) An NDFSM $M$ must halt for any $w$.

(c) For any NDFSM $M$ we can find a DFSM $M'$, such that $L(M) = L(M')$.

(d) Let $M$ be an NDFSM. If $M$ does not have ε-transitions, then $M$ must halt for any $w$.

(e) A DPDA $M$ must halt for any $w$.

(f) An NDPDA $M$ must halt for any $w$.

(g) For any NDPDA $M$, we can find a DPDA $M'$, such that $L(M) = L(M')$.

(h) For any NDPDA $M$, we can find an NDPDA $M'$ that halts, such that $L(M) = L(M')$.

(i) Let $M$ be an NDFSM. We can find some DPDA $M'$ that decides $L(M)$.

(j) Any regular language or context-free language can be decided by some TM.

(k) A DTM $M$ must halt for any $w$.

(l) If both $L$ and $\neg L$ are in SD, then $L$ is in D.

(m) NP-complete is a subset of NP-hard.

(n) If P = NP, then NP-complete languages will be in P as well.

(o) The union of a finite number of countably infinite sets is countable.

(p) The union of a countably infinite number of countably infinite sets is countable.

(q) Regular languages and context-free languages are in P.

(r) If $L = L1 \cap L2$ and $L$ is regular, then $L_1$ and $L_2$ must be regular.

(s) If $L$ is regular, then $\neg L$ is also regular.

(t) $\neg A^n B^n C^n$ is context-free, but $A^n B^n C^n$ is not. Thus context-free languages are not closed under complement.

(u) Given the FSM $M$ as in Fig. 1, write down the first five strings in $L(M)$ in lexicographical order.

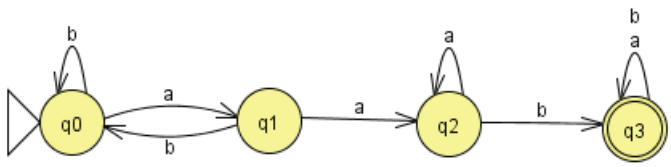



Figure 1: An FSM.

(v) Given the PDA $M$ as in Fig. 2, write down the first five strings in $L(M)$ in lexicographical order.

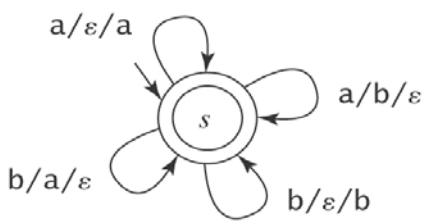



Figure 2: A PDA.

**Department of Computer Science**
**Core Graduate Comprehensive Exam**
**Spring 2013**

- **Answer the questions on the paper supplied.**
- **Answer question 1 or Answer question 2. Answer question 3 or 4. Answer question 5 or 6 or 7 or 8. Answer question 9 or 10. You should answer a total of four questions.**
- **Start each question on a new page. Write on only one side of the paper.**
- **Write your SIX-DIGIT Texas State ID in the top right corner of each page of your answer. Do NOT put your name anywhere on the answers.**
- **Put the number of the question being answered in the top left corner of each answer page. Put the CORRECT question number to avoid missing your answer.**
- **If the answer to a question is written on more than one page, number the pages consecutively.**

## Group 1

1. **CS 5329 Algorithm Design and Analysis**

   - Define mathematically the asymptotic upper bound big $O$ notation: $f(n) = O(g(n))$
     $O(g(n)) = \{$
     $\}$
   - Use recursion tree and log algebra to derive the maxima total running time of the recurrence function: $T(n) = T(n/3) + T(2n/3) + cn$

2. **CS 5329 Algorithm Design and Analysis**

   Give definition and an example of a double hashing scheme for building a hash table. Insert entries with keys $1, 21, 4, 13$ and 34 in a hash table of size five (i.e., $N = 5$) using the double hashing scheme you defined. Draw final table and show the calculations for full credit.

## Group 2

3. **CS 5346 Advanced Artificial Intelligence**

   (a) In the following cryptarithmetic problem, each letter stands for a distinct digit. The aim is to find a substitution of digits for letters such that the resulting sum is arithmetically correct, with the added restriction that no leading zeros are allowed.

   ```
       O  N  E
   + F O  U  R
   ---------------
     F  I  V  E
   ```

   Write a set of constraints which must be satisfied to solve this problem.

   (b) Convert the following English sentences into predicate logic sentences. Then convert predicate logic sentences into Atomic and Horn sentences.

   i. Horses are mammals.
   ii. Cows are mammals.
   iii. Pigs are mammals.
   iv. An offspring of a horse is a horse.
   v. Bluebeard is a horse.
   vi. Bluebeard is Charlie's parent.
   vii. Offspring and parent are inverse relations.
   viii. Every mammal has a parent.

   Using Backward Chaining method, draw the proof tree to prove Horse (Charlie).

4. **CS 5391 Survey of Software Engineering**

   Describe the pros and cons of the following models and point out their applications.

   (a) linear sequential

   (b) incremental

   (c) spiral

**Group 3**

5. **CS 5306 Advanced Operating Systems**

   There are two main approaches to organizing a server daemon such as a Web server:

   (a) Create a new kernel thread for each client (for each Web browser connection);

   (b) Use a single process responding to all clients, usually based on the select() system call. Compare and contract these two approaches.

6. **CS 5310 Network and Communication Systems**

   (a) Why Internet is a connectionless packet switched network?

   (b) Explain the meaning of TCP connection. Does the notion of TCP connection contradict the fact the Internet is a connectionless packet switched network?

   (c) If a user tries to send an email message from one of the Linux workstations in the CS lab to a friend, describe all the protocols and sequence of actions involved in sending the email message.

7. **CS 5332 Data Base Theory and Design**

   Construct an E/R diagram for a database system for the following problem. You must specify the PK's, FK's, and relationships for each entity. Your relations (entities) must be normalized up to 3NF or BCNF. (Prefer you use crow feet notation for E/R model. If you use other notations, you must specify the relationships clearly.)

   The problem: Design a database for Ray. He is interested in movies and wants to keep information on movies, actors, and directors in a database. The only user is Ray and he needs to produce the following reports:

   (a) For each director, list his or her number, name, and the year he or she was born. If the director is deceased, list the year of death.

   (b) For each movie, list its number, title, the year the movie was made, and its type (for example, Comedy, Drama, Science Fiction, and so on).

   (c) For each movie, list its number, title, the number and name of its director, the critics' rating, the MPAA rating (G, PG, PG-13, or R), the number of awards for which the movie was nominated, and the number of awards the movie won. (The critics rate the movie with a number of "stars." Four stars is the top rating possible. Zero stars is the worst possible rating.)

   (d) For each actor, list his or her number, name, birthplace, and the year he or she was born. If the actor is deceased, list the year of death.

   (e) For each movie, list its number and title, along with the number and name of the actors who appeared in it.

   (f) For each actor, list his or her number and name, along with the number and name of the movies in which the actor starred.

8. **CS 5332 Data Base Theory and Design**

   Consider the following schema:

   Suppliers (sid: integer, sname: string, address: string)

   Parts (pid: integer, pname: string, color: string)

   Catalog (sid: integer, pid: integer, cost: real)

   Write each of the following queries in relational algebra and SQL.

   (a) Find the names of suppliers who supply some red part.

   (b) Find the sids of suppliers who supply every red part.

**Group 4**

9. **CS 5318 Design of Programming Languages**

   (a) A number consists of digits from 0 to 9. For example, 123 is a number. Give two different grammars to define a number, one using a left-recursive rule and the other using a right-recursive rule.

   (b) A decimal number can be defined as a number followed by a decimal point (.) and another number. For example, 123.045 is a decimal number. Define attributes and write an attribute grammar to carry out the following tasks:

      i. Restrict the number of digits after the decimal point to no more than 6;

      ii. Compute the decimal value of any decimal number with no more than 6 digits after the decimal point.

   (c) Decorate the parse tree for 123.045 to show how its value is computed.

10. **CS 5338 Formal Languages**

    (a) Draw an FSM $M$ for language $L(M) = \{w \in \{a,b\}^* : w \text{ has } aab \text{ as a substring}\}$

    (b) Draw a PDA $M$ for language $L(M) = \{w \in \{a,b\}^* : \#_a(w) = \#_b(w)\}$. Note that $\#_a(w)$ denotes the number of $a$'s in $w$.

**Department of Computer Science**
**Core Graduate Comprehensive Exam**
**Spring** 2018

- **Answer the questions on the paper supplied.**
- **Answer question 1 or 2. Answer question 3 or 4. Answer question 5 or 6 or 7 or 8 or 9. Answer question 10 or 11 or 12 or 13. You should answer a total of four questions. Please Note: If you answer more than one question in one group, only the one with the LOWEST score will be counted.**
- **Start each question on a new page. Write on only one side of the paper.**
- **Write your SIX-DIGIT Texas State ID in the top right corner of each page of your answer. Do NOT put your name anywhere on the answers.**
- **Put the number of the question being answered in the top left corner of each answer page. Put the CORRECT question number to avoid missing your answer.**
- **If the answer to a question is written on more than one page, number the pages consecutively.**

**Group 1**

1. **CS 5329 Algorithm Design and Analysis**
   {*from Dr. Hwang*}

   For the recurrence formula from Merge Sort and Max-subrray,

   $T(n) = 2T(n/2) + n \;\; for \;\; n > 1$

   (a) Derive the total running time T(n) by using Recursion Tree method.

   (b) Derive the total running time T(n) by using Iteration method.

2. **CS 5329 Algorithm Design and Analysis**
   {*from Dr. Metsis*}

   We want an efficient algorithm to sort n elements G[1..n], where each element G[i] is of either 0 or 1. The sorting must be done IN-PLACE, using only O(1) amount of additional space.

   (a) Write a procedure (pseudocode) for SORT(G,n).

   (b) What is the time complexity of this algorithm?

   (c) Illustrate how the sorting is done on the following array of key values. Show the result after each data movement.
   0 1 0 1 0 0 1 0 0 0 1 1 0 1 1

**Group 2**

3. **CS 5346 Advanced Artificial Intelligence**
   {*from Dr. Ali*}

   (a) Explain Hill Climbing (with and without back tracking) in detail with examples and compare it with the A* algorithm.

   (b) A database of a county banks loan department is represented as positive and negative instances in the following table:

| Individual | BAL | RATING | APP | INC | OK |
|---|---|---|---|---|---|
| 1 | 1 | 0 | 0 | 1 | 0 |
| 2 | 0 | 0 | 1 | 0 | 0 |
| 3 | 1 | 1 | 0 | 1 | 1 |
| 4 | 0 | 1 | 1 | 1 | 1 |
| 5 | 0 | 1 | 1 | 0 | 0 |
| 6 | 1 | 1 | 1 | 0 | 1 |
| 7 | 1 | 1 | 1 | 1 | 1 |
| 8 | 1 | 0 | 1 | 0 | 0 |
| 9 | 1 | 1 | 0 | 0 | 0 |

Derive a set of rules which cover all and only the positive instances.

4. **CS 5391 Survey of Software Engineering**
   {*from Dr. Chen*}

   Explain the following software process models and point out their pros and cons:

   (a) Formal methods

   (b) Agile development model

**Group 3**

5. **CS 5306 Advanced Operating Systems**
   {*from Dr. Tamir*}

   Assume a single core system implementing an intra-core preemptive HRRN scheduling policy with a slice size of 1 second. Further assume that at time $T$ there are 3 tasks $\{T_0, T_1, T_2\}$ in the Ready Queue of the core with no task in the execution slot of the core. Additionally, assume that the tasks are compute-bound with no I/O whatsoever. Let $\{P_0, P_1, P_2\}$ be the remaining execution time of $\{T_0, T_1, T_2\}$ respectively and let $\{P_3, P_4, P_5\}$ be the current wait time of $\{T_0, T1, T_2\}$ respectively, where:

   $P_0 = 1,$

   $P_1 = 4,$

   $P_2 = 2,$

   $P_3 = 1,$

   $P_4 = 2,$

   $P_5 = 3,$

   1) Clearly describe the state of the system in the first 20 seconds following time $T$.

6. **CS 5306 Advanced Operating Systems**
   {*from Dr. Chen*}

   Design a distributed algorithm to synchronize physical clocks in a distributed system in a room.

7. **CS 5332 Data Base Theory and Design**
   {*from Dr. Hwang*}

   For the university real world problem including information for Class, Course, Instructor, Student, and Classroom. Draw an E/R diagram by creating essential attributes for all information with pk's, fk's, and relationships. You may need to add additional information as needed. (Notes: The Course is the course in government inventory such as CS 5332; while the Class is in the class scheduling such as CS 5332 Fall 2016 Sec 1 and CRN#. You must use crow feet notation. Do not choose this problem unless you master crow feet notation.)

8. **CS 5332 Data Base Theory and Design**
{ *from Dr. Ngu*}

Consider the following relational schema describing a part of a university database:

Department(dno:integer, dname:string)
Instructor(staffNo:integer, name:string, dno:integer)
Course(cid:string, name:string)
Class(cid:string, meetAt:string, room: string, staffNo:integer)
Enrolled(studentid:integer, cid:string)
Student(studentid:integer, name:string, major: string)

Assume the followings:

- the fields that are underlined are the primary keys of the relations
- there is a referential integrity constraint between Department and Instructor via 'dno'
- there is a referential integrity constraint between 'staffNo' in relation Instructor and 'staffNo' in relation Class
- there is a referential integrity constraint between 'cid' in relation Course and 'cid' in relation Class
- there are also referential integrity constraints between 'cid' in relation Course and relation Enrolled as well as between 'studentid' in relation Student and Enrolled.
- Some example values of meetAt are "Monday 12:30 pm", "Thursday 6:30 pm".
- Some example values of cid are "CS4332", "CS3343".

Implement solutions to the following:

(a) Which students are enrolled in 'CS4332' course, display both studentid and student name.

(b) How many students are enrolled in the course 'CS5332'?

(c) Which students have classes on Monday? Display names of students.

(d) How many students are enrolled in the courses that Dr. Hongchi Shi teaches?

(e) Show a list of all students (name and student-id) and the number of courses each student is enrolled in. A correct answer will involve using outerjoin since some students might not be enrolling in any course in a particular year.

9. **CS 5310 Network and Communication Systems**
{ *from Dr. Peng*}

(a) Describe the steps and protocols used when you send an email message using a computer in CS lab to a friend of yours who uses an email server on a campus in California.

(b) Describe briefly why network layer is not an end-to-end layer.

**Group 4**

10. **CS 5338 Formal Languages**
{ *from Dr. Zare* }

(a) True or false only. No need to make any argument or explain why.

i. The complement of a decidable language is decidable regular.
ii. A decidable language can be recognized by a push–down automaton.
iii. Boolean sustainability problem (SAT) is not in NP.
iv. There is a regular language that is in NP but not in P.
v. The language $A = \{0^n 1^{2n+m} 0 | m > 2, n < 3\}$ is regular.

(b) Determine the regular languages in the following list. For those language that you think are regular, provide a proof for your claim. For the rest, you do NOT need to do anything. The alphabet is $\Sigma = \{0,1\}$.

i. $L_1 = \{0^{m-1}1^{n+1} | n > m.\}$

ii. $L_2 = \{ww | w$ starts with 000 and contains 001 or 11 .$\}$

iii. $L_3 = \{0^n 10^m 101^k | n$ is odd and $k < 2.\}$

iv. $L_4 = \{wv | w$ starts with 000 and contains 001 or 11.$\}$

11. **CS 5338 Formal Languages**
{*from Dr. Gao*}

For each of the following questions, circle all the correct statements or write "none" if none of them are correct.

(a) The correct statements are:
(a) A language can have infinite number of strings.
(b) Each string in a language must be finite.
(c) A string in a language must be non-empty.
(d) A language must be non-empty.

(b) The correct statements are:
(a) A DFSM must halt.
(b) A NDFSM must halt.
(c) A NDFSM without ε-transitions must halt.

(c) The correct statements are:
(a) A DPDA must halt.
(b) A NDPDA must halt.
(c) A NDPDA without ε-transitions must halt.

(d) The correct statements are:
(a) Let M be a DFSM. If ε ∈ L(M), then the start state of M must be an accepting state.
(b) Let M be a DFSM. If the start state of M is an accepting state, then ε ∈ L(M).
(c) Let M be a NDFSM. If ε ∈ L(M), then the start state of M must be an accepting state.
(d) Let M be a NDFSM. If the start state of M is an accepting state, then ε ∈ L(M).

(e) The correct statements are:
(a) Regular languages are closed under union.
(b) Regular languages are closed under intersection.
(c) Regular languages are closed under complement.
(d) Regular languages are closed under set difference.

(f) The correct statements are:
(a) Context free languages are closed under union.
(b) Context free languages are closed under intersection.
(c) Context free languages are closed under complement.
(d) Context free languages are closed under set difference.

(g) The correct statements are:
(a) A DTM must halt.
(b) A DTM must halt in at most $|w|$ steps.
(c) A NDTM must halt.
(d) A NDTM must halt in at most $|w|$ steps.

(h) The correct statements are:
(a) In terms of computability, DFSM = NDFSM < DPDA < NDPDA < DTM = NDTM.
(b) In terms of computability, DFSM = NDFSM < DPDA = NDPDA < DTM = NDTM.
(c) In terms of computability, DFSM < NDFSM < DPDA < NDPDA < DTM < NDTM.

(i) The correct statements are:
(a) Decidable languages are closed under union.
(b) Decidable languages are closed under intersection.
(c) Decidable languages are closed under complement.
(d) Semi-decidable languages are closed under complement.

(j) The correct statements are:
(a) SAT is the first language proved to be NP-complete.
(b) 3-SAT is the first language proved to be NP-complete.
(c) Both SAT and 3-SAT are NP-hard.

12. **CS 5318 Design of Programming Languages**
{ *from Dr. Shi*}

Given the following grammar:

A → **a** E | **b** A A

B → **b** E | **a** B B

E → **a** B | **b** A | ε

(a) Give a left-most derivation and draw the parse tree for **a b b a a b a**.

(b) Is the grammar LL(1)? If so, show the parse table; if not, identify a prediction conflict.

(c) Design appropriate attribute(s) and write an attribute grammar to count the number of **a**'s and the number of **b**'s in the input sentence. For example, the number of **a**'s and the number of **b**'s the sentence in a) are 4 and 3, respectively.

(d) Decorate the parse tree for the sentence in a) with your attribute(s) defined in c) to show how the number of **a**'s and the number of **b**'s are counted.

13. **CS 5351 Parallel Processing**
{ *from Dr. Burtscher*}

Assume a shared-memory parallel program that requires synchronization to avoid potential data races. Explain under which conditions a critical section, an atomic operation, or a lock can be used to eliminate the data races. Next, compare and contrast the three approaches in terms of performance, portability, and ease-of-use for the programmer.

**Department of Computer Science**
**Core Graduate Comprehensive Exam**
**Fall** 2018

- **Answer the questions on the paper supplied.**
- **Answer question 1 or 2. Answer question 3 or 4. Answer question 5 or 6 or 7 or 8 or 9. Answer question 10 or 11 or 12. You should answer a total of four questions. Please Note: If you answer more than one question in one group, only the one with the LOWEST score will be counted.**
- **Start each question on a new page. Write on only one side of the paper.**
- **Write your SIX-DIGIT Texas State ID in the top right corner of each page of your answer. Do NOT put your name anywhere on the answers.**
- **Put the number of the question being answered in the top left corner of each answer page. Put the CORRECT question number to avoid missing your answer.**
- **If the answer to a question is written on more than one page, number the pages consecutively.**

## Group 1

1. **CS 5329 Algorithm Design and Analysis**
   {*from Dr. Hwang*}

   For the recurrence formula from Merge Sort and Max-subrray,

$$T(n) = 2T(n/2) + n \text{ for } n > 1$$

   (a)**Derive** the total running time T(n) by **using** Recursion Tree method.

   (b) **Derive** the total running time T(n) by **using** Iteration method.

   **(Answers alone will receive very minimum credit.)**

2. **CS 5329 Algorithm Design and Analysis**
   {*from Dr. Metsis*}

   **Dynamic Programming**

   The longest common subsequence (LCS) of two sequences of characters can be computed using the following algorithm:

```
LCS-LENGTH(X, Y)
m ← length[X]
n ← length[Y]
for i ← 1 to m
      do c[i, 0] ← 0
for j ← 0 to n
      do c[0, j] ← 0
for i ← 1 to m
      do for j ← 1 to n
            do if x_i = y_j
                  then c[i, j] ← c[i − 1, j − 1] + 1
                       b[i, j] ← "↖"
                  else if c[i − 1, j] ≥ c[i, j − 1]
                          then c[i, j] ← c[i − 1, j]
                               b[i, j] ← "↑"
                          else c[i, j] ← c[i, j − 1]
                               b[i, j] ← "←"
return c and b
```

   a) Given the sequences A = $a,e,a,a,g,s,s,t$ and B = $b,a,e,b,s,a,g,t$, find the longest common subsequence. Show the resulting tables $c$ and $b$. Table $c$ shows the length of the LCS, whereas table $b$ shows the path to be followed to extract the LCS itself.

   b) What is the time and space complexity of the above algorithm? Can we make any further improvements to either of them?

**Group 2**

3. **CS 5346 Advanced Artificial Intelligence**
   {*from Dr. Ali*}

   (a) Consider the Wumpus environment E as a 4 X 4 grid of rooms (E2,3 represents 2nd row and 3rd column). Sensors have detected the following facts:

   B2,1; B1,2; B2,3; B3,2 Where Bi,j indicates Breeze in the i-th row and j-th column. Breeze is not detected in any other room.

   Write rules in propositional logic to determine the position of the Pit and change them into Conjunctive Normal Form.

   Using resolution rule, develop a proof tree to determine the location of the Pit.

   (b) Represent the following story in propositional logic and then transform it to the conjunctive Normal Form:

   "James, Stevens and Nelson hold the jobs of dentist, pilot, and marine (not necessarily in that order). James owes the dentist $39. The marine's spouse prohibits borrowing money. Stevens is not married."

   Using resolution rule develop a proof tree and determine which person has which job.

4. **CS 5391 Survey of Software Engineering**
   {*from Dr. Chen*}

   Explain the following software process models and point out their pros and cons:

   (a) Formal methods

   (b) Agile development model

**Group 3**

5. **CS 5306 Advanced Operating Systems**
   {*from Dr. Tamir*}

   Assume a single core system implementing an intra-core preemptive HRRN scheduling policy with a slice size of 1 second. Further assume that at time $T$ there are 3 tasks $\{T_0, T_1, T_2\}$ in the Ready Queue of the core with no task in the execution slot of the core. Additionally, assume that the tasks are compute-bound with no I/O whatsoever. Let $\{P_0, P_2, P_2\}$ be the remaining execution time of $\{T_0, T_1, T_2\}$ respectively and let $\{P_3, P_4, P_5 4\}$ be the current wait time of $\{T_0, T_1, T_2\}$ respectively, where:

   $P_0 = 1,$

   $P_1 = 4,$

   $P_2 = 2,$

   $P_3 = 1,$

   $P_4 = 2,$

   $P_5 = 3,$

   Clearly describe the state of the system in each of the first 20 seconds following time $T$.

6. **CS 5306 Advanced Operating Systems**
   {*from Dr. Chen*}

   Design a distributed algorithm to synchronize physical clocks in a distributed system in a room.

7. **CS 5332 Data Base Theory and Design**
   {*from Dr. Hwang*}

   For the following real world universities problem including information

   for Class, Course, Instructor, Student, and Classroom.

   (a) Indicate all functional dependencies among all essential attributes including those you created.

   (b) Draw an E/R diagram by creating essential attributes for all information with pk's, fk's, and relationships.

You may need to add additional information as needed.

(Notes: Courses are in the university inventory, like CS 5332; while classes are the classes you are taking or were taken, like CS 5332 Spring 2018 Section 1.)

(You must use crow feet notation for this problem. If you can not master the notation, do not select this problem.)

8. **CS 5332 Data Base Theory and Design**
{ *from Dr. Ngu*} Consider the following employee database, where the primary keys are underlined.

   Employee(ename:string, street:string, city:string)
   Works(*employee:string, company:string*, salary:real)
   Company(cname:string, city:string)
   Manages(*employee:string, company:string*)

   Give a single SQL statement for each of the following queries:

   (a) *()* Find the names, street, and cities of residence of all employees who work for the "First Bank Corporation" and earn more than $40,000.

   (b) *()* Find the names of all employees in the database who live in the same cities as the companies for which they work.

   (c) *()* Give all managers of "First Bank Corporation" a 10 percent salary raise.

   (d) *()* Find the names of all employees in the database who earn more than every employee of the "Small Bank Corporation".

   (e) *()* Find the name of the company that has the most employees. Your solution should not involve an intermediate view.

9. **CS 5310 Network and Communication Systems**
{ *from Dr. Peng*}

   (a) Describe the steps and protocols used when you send an email message using a computer in CS lab to a friend of yours who uses an email server on a campus in California.

   (b) Describe briefly why network layer is not an end-to-end layer.

**Group 4**

10. **CS 5338 Formal Languages**
{ *from Dr. Gao*}

   For each of the following questions, circle all the correct statements or write "none" if none of them are correct.

   (a) The correct statements are:
   (a) A language can have infinite number of strings.
   (b) Each string in a language must be finite.
   (c) A string in a language must be non-empty.
   (d) A language must be non-empty.

   (b) The correct statements are:
   (a) A DFSM must halt.
   (b) A NDFSM must halt.
   (c) A NDFSM without ε-transitions must halt.

   (c) The correct statements are:
   (a) A DPDA must halt.
   (b) A NDPDA must halt.
   (c) A NDPDA without ε-transitions must halt.

(d) The correct statements are:
(a) Let M be a DFSM. If $\varepsilon \in L(M)$, then the start state of M must be an accepting state.
(b) Let M be a DFSM. If the start state of M is an accepting state, then $\varepsilon \in L(M)$.
(c) Let M be a NDFSM. If $\varepsilon \in L(M)$, then the start state of M must be an accepting state.
(d) Let M be a NDFSM. If the start state of M is an accepting state, then $\varepsilon \in L(M)$.

(e) The correct statements are:
(a) Regular languages are closed under union.
(b) Regular languages are closed under intersection.
(c) Regular languages are closed under complement.
(d) Regular languages are closed under set difference.

(f) The correct statements are:
(a) Context free languages are closed under union.
(b) Context free languages are closed under intersection.
(c) Context free languages are closed under complement.
(d) Context free languages are closed under set difference.

(g) The correct statements are:
(a) A DTM must halt.
(b) A DTM must halt in at most $|w|$ steps.
(c) A NDTM must halt.
(d) A NDTM must halt in at most $|w|$ steps.

(h) The correct statements are:
(a) In terms of computability, DFSM = NDFSM $<$ DPDA $<$ NDPDA $<$ DTM = NDTM.
(b) In terms of computability, DFSM = NDFSM $<$ DPDA = NDPDA $<$ DTM = NDTM.
(c) In terms of computability, DFSM $<$ NDFSM $<$ DPDA $<$ NDPDA $<$ DTM $<$ NDTM.

(i) The correct statements are:
(a) Decidable languages are closed under union.
(b) Decidable languages are closed under intersection.
(c) Decidable languages are closed under complement.
(d) Semi-decidable languages are closed under complement.

(j) The correct statements are:
(a) SAT is the first language proved to be NP-complete.
(b) 3-SAT is the first language proved to be NP-complete.
(c) Both SAT and 3-SAT are NP-hard.

11. **CS 5318 Design of Programming Languages**
{ *from Dr. Shi* }

(a) Discuss how an actual parameter is related to its corresponding formal parameter in the following four parameter-passing methods for procedure calls.

a. Pass by value

b. Pass by reference

c. Pass by name

(b) Given the output of the following program (written in C syntax) using the above three parameter-passing methods:

```
int i, j, a[5]; // a is a 5 element array with indices 0 - 4

void swap(int x, int y) {

int temp = x;

x = y;

y = temp;

}

int main() {

for (j=0; j<5; j++)

a[j] = j;

i = 1;

swap(i, a[i+1]);

printf("%d %d\n", i, a[2]);

return 0;

}
```

Assume that procedure arguments are evaluated from left to right. What numbers does the program print if both parameters in swap are passed by

a. value?

b. Reference?

c. Name?

Note that you can get partial credit if you show your hand evaluation of the code.

12. **CS 5351 Parallel Processing**
{ *from Dr. Burtscher*}

Assume a shared-memory parallel program that requires synchronization to avoid potential data races. Explain under which conditions a critical section, an atomic operation, or a lock can be used to eliminate the data races. Next, compare and contrast the three approaches in terms of performance, portability, and ease-of-use for the programmer.