Computer Science Department
Stanford University
Comprehensive Examination in Analysis of Algorithms
Autumn 1993

October 29, 1993

READ THIS FIRST!

1. You should write your answers for this part of the Comprehensive Examination in BLUE BOOKs. There are 5 problems in the exam; use a SEPARATE blue book for each problem. Be sure to write your MAGIC NUMBER on the cover of every blue book that you use.
2. The number of POINTS for each problem indicates how elaborate an answer is expected. The total number of points is 60, and the exam takes 60 minutes. This "coincidence" can help you plan your time.
3. This exam is CLOSED BOOK.
4. Show your work, since PARTIAL CREDIT will be given for incomplete answers.
5. There is only one page of the exam.

Comprehensive Exam: Analysis of Algorithms (Solutions) Autumn 1993

1. (8 points)

   Determine (as tightly as you can) the order of growth of the function $T(n)$ defined by the following recurrence.

   - $T(1) = 1$.
   - For $n \geq 2$,
   $$T(n) = 2T(n/2) + \log n.$$

   Solution : Since we are interested in determining simply the order of growth, we may assume that $n = 2^k$ for some $k \geq 1$. We will show that $T(n)$ grows as $O(n)$. The recurrence may be rewritten as
   $$t_k = 2t_{k-1} + k$$
   where $t_k = T(2^k)$.

   We now unfold the recurrence to obtain
   $$t_k = 2^k + 2^k \sum_{i=1}^{k} \frac{i}{2^i}$$

   It is a trivial exercise to show that $\sum_{i=1}^{k} \frac{i}{2^i} = O(1)$. Alternatively, one may obtain a precise expression for $S(k) = \sum_{i=1}^{k} \frac{i}{2^k}$ by simply observing that $S(k) - \frac{1}{2}S(k) = \sum_{i=1}^{k} \frac{1}{2^i} - \frac{k}{2^{k+1}}$. It follows that $t_k = 3.2^k - k - 2$. Substituting $k = \log n$, we get $T(n) = 3n - \log n - 2$.

2. (5 points)

   Determine the number of binary sequences which contain n zeros and m ones such that each pair of ones is separated by at least k zeroes.

   Solution : Let S be the set of the binary sequences mentioned above and let S' be the set of binary sequences of length $n+m-k(m-1)$ which contain precisely m ones. For every sequence $\sigma \in S$, consider the sequence $f(\sigma)$ which is obtained by simply deleting k zeros between every pair of successive ones in σ. It is easily seen that f defines a bijection from S to S'. Thus the number of desired sequences is simply $\binom{n+m-k(m-1)}{m}$.

3. (10 points)

   Suppose you are given an array A containing n keys which are not necessarily distinct. In fact, you are guaranteed that there are only $O(\log n)$ distinct keys in the array. Show that you can sort A using $O(n \log\log n)$ time.

   Solution : We use variant of insertion sort where the keys are inserted into a balanced binary search tree data structure (like the AVL trees or the red-black trees). The slight modification is that each node in the tree has a counter associated with it. We insert vertex of non-zero in-degree has a path leading into it from some vertex of zero in-degree. (Start at any non-zero in-degree vertex and keep moving back into any vertex with an edge coming into the current vertex. Since the graph is finite and acyclic, we will end up at a zero in-degree vertex.)

(b) (15 points) Describe an $O(|V| + |E|)$ time algorithm for finding a *minimum* start set in an *arbitrary* directed graph $G(V, E)$. Briefly justify the correctness of your algorithm.

Solution : Compute the strongly connected components of G, say $C_1, C_2, \ldots, C_k$ in $O(|V| + |E|)$ time. Construct the strongly connected component graph SCC which has a vertex corresponding to each strong component. There is an edge from a vertex i to j in SCC if and only if there exists $u \in C_i$ and $v \in C_j$ such that there is an edge from u to v in G. Clearly, the graph SCC is a directed acyclic graph. Now simply choose any vertex from each strongly connected component C_i such that the vertex i is a vertex of zero in-degree in SCC. It is easily seen that this set of vertices forms a start set, and the minimality of this set follows from the previous part of the question.

Computer Science Department
Stanford University
Comprehensive Examination in Automata, Languages, and Logic
Autumn 1993

October 29, 1993

READ THIS FIRST!

1. You should write your answers for this part of the Comprehensive Examination in BLUE BOOKS. There are three problems in the exam; use a SEPARATE blue book for each problem. Be sure to write your MAGIC NUMBER on the cover of every blue book that you use.

2. The number of POINTS for each problem indicates how elaborate an answer is expected. The total number of points is 60, and the exam takes 60 minutes. This "coincidence" can help you plan your time.

4. This exam is OPEN BOOK. You may use notes, articles, or books—but no help from other sentient agents such as other humans or robots.

5. Show your work, since PARTIAL CREDIT will be given for incomplete answers.

6. there are 3 pages of the exam.

Proof:

(a) 8 points

Use the pumping lemma to show that $\overline{L}$, the complement of L, is not regular. By Theorem 3.2 of Hopcroft and Ullman, it follows that L cannot be regular.

For any n, let $z = 0^n 1^n$. Observe that $z \in \overline{L}$, i.e., z has the same number of 0's and 1's. For any u, v, w such that $z = uvw$ and $|uv| \leq n$, observe that v must consist only of 0's. Clearly $uv^iw \notin \overline{L}$ for any $i > 1$, so $\overline{L}$ is not regular.

(b) 12 points

Suppose L is recursive, i.e., there is a TM M_L that accepts L and halts on all inputs. Use M_L to construct a TM to decide L_u (the universal language) as follows. Given an input $\langle M, w\rangle$, construct a TM M' from M by adding an automaton state q, such that every transition to an accepting state in M instead goes to q. Observe that M' enters q on w if and only if M accepts w. It follows that L cannot be recursive.

2

Proof:

(a) (10 points)

Either of the following solutions is correct.

$$\mathcal{F}': \quad \begin{array}{l} \textit{if} \left(\begin{array}{l} \textit{if } p(w) \textit{ then } q(w, f(x,y,w)) \\ \quad \wedge \\ \textit{if } q(w,z) \textit{ then } p(w) \end{array} \right) \\ \textit{then } r(x,y) \end{array}$$

$$\mathcal{F}': \quad \begin{array}{l} \textit{if} \left(\begin{array}{l} p(w) \wedge q(w, f(x,y,w)) \\ \quad \vee \\ \textit{not } p(w) \wedge \textit{not } q(w,z) \end{array} \right) \\ \textit{then } r(x,y) \end{array}$$

(b) (10 points)

$\mathcal{F}: \quad (\forall x')q(f(g(x')), g(x)) \ \wedge \ \neg q(f(a), g(x))$

3 **Proof:**

(a) 15 points

If N is definable in $\mathfrak{N}_A$, then N is eventually periodic. Choose M and p such that, for every $n > M$, $n \in N$ if and only if $n + p \in N$. Construct a finite automaton with states indexed $s = 0, 1, \ldots, M + p + 1$ such that there is a transition from $s = i$ to $s' = i + 1$ for all $0 \leq i \leq M + p$, and a transition from $s = M + p + 1$ to $s' = M + 1$. Each state $s = i$ is an accepting state if and only if $i \in N$.

Conversely, assume that the unary representation L of N is accepted by some finite state automaton. Construct a DFA $(Q, \Sigma, \delta, q_0, F)$ accepting L, and assume that every $q \in Q$ is reachable from the initial state q_0. Since Σ is a singleton set (containing $\mathbf{0}$), there can only be a single cycle in the DFA. There is an obvious mapping I from Q to the set $\{0, 1, \ldots, |Q| - 1\}$, where $I(q_0) = 0$ and $I(q') = I(q) + 1$ if $\delta(q) = q'$ (except the "last state" q_l such that $I(q_l) = |Q| - 1$). Take $M = I(\delta(q_l)) - 1$ and $p = |Q| - I(q_l)$. Clearly $\mathbf{0}^n \in L$ if and only if $\mathbf{0}^{n+p} \in L$ for every $n > M$, implying that N is eventually periodic, and therefore definable in $\mathfrak{N}_A$.

(b) 5 points

By the Gödel Incompleteness Theorem and Theorems 33E and 34A of Enderton, there are relations definable in $\mathfrak{N}$ that are not recursive.

Solutions to Comprehensive: Numerical Analysis (30 points). Autumn 1993

Problem I (16 points). **Iterative Solution of Linear Equations**

This question concerns the solution of systems of linear equations in the form

$$Ax = b, \tag{1}$$

where A is a $m \times m$ matrix and x and b are vectors of length m.

(a) Describe the simplest form of *iterative improvement* (also known as *residual correction*) to solve (1). Describe, and briefly explain, the effect of machine precision on this algorithm.

SOLUTION:

Given an approximation x^m to the solution x of (1), the residual r^m is defined by

$$r^m = b - Ax^m.$$

Iterative improvement generates the sequence of approximations

$$x^{m+1} = x^m + \hat{e}^m$$

where $\hat{e}$ is the computed solution of

$$Ae^m = r^m.$$

For such an iteration it is important to obtain accurate values for r^m relative to the precision used in the remainder of the calculation. The reason is simply that otherwise the errors in $\hat{e}^m$ may be comporable with the errors in the original calculation.

(b) Given a matrix C which approximates the inverse of A, consider the following general residual correction method for the solution of (1):

$$r^m = b - Ax^m,$$

$$x^{m+1} = x^m + Cr^m.$$

State the precise condition under which this iteration converges; prove your assertion.

SOLUTION: *The iteration converges provided that the spectral radius of the matrix $I - CA$ is less than one.* To prove this note that the iteration gives

$$x^{m+1} - x = x^m - x + C[b - Ax^m] = x^m - x + C[Ax - Ax^m].$$

Hence the error $\delta^m = x^m - x$ satisfies

$$\delta^{m+1} = [I - CA]\delta^m.$$

It is well-known that it is necessary and sufficient for the spectral radius of $I - CA$ to be less than one for the iterates δ^m to converge to zero, independent of the choice of initial vector.

(c) Given that A may be written as $A = L + D + U$ where L and U are lower and upper triangular respectively and D is diagonal, define the Jacobi and Gauss-Siedel iterations for the solution of (1).

SOLUTION: We have

$$(L + D + U)x = b.$$

The *Jacobi iteration* is to generate x^m according to

$$Dx^{m+1} = b - [L + U]x^m.$$

This also shows that $x^{m+1} \in [a,b]$ and the induction is complete. Letting $m \to \infty$ gives the desired result.

(c) Using part (b) show that Newton iteration converges to a root α of (2) provided that f is twice continuously differentiable and $f'(\alpha) \neq 0$ and the initial guess is sufficiently close to α.

SOLUTION: We can write Newton iteration in the form of (b) with

$$g(x) := x - \frac{f(x)}{f'(x)}.$$

Thus

$$g'(x) = 1 - \frac{f'(x)}{f'(x)} + \frac{f(x)f''(x)}{(f'(x))^2}.$$

Thus it follows that $g'(\alpha) = 0$ and hence there is an interval $[a,b]$, including α, and $\mu < 1$ such that

$$|g'(x)| \leq \mu \quad \forall x \in [a,b].$$

The result follows.

59