Written Exam

10:00 – 12:30, February 5, 2019

Entrance Examination (AY 2019)

Department of Computer Science
Graduate School of Information Science and Technology
The University of Tokyo

Notice:

(1) Do not open this problem booklet until the start of the examination is announced.

(2) Answer the following 4 problems. Use the designated answer sheet for each problem.

(3) Do not take the problem booklet or any answer sheet out of the examination room.

Write your examinee's number in the box below.

Examinee's number	No.

余白 (blank page)
計算などに使ってもよいが，切り離さないこと． Usable for memos; do not detach.

余白 (blank page)

計算などに使ってもよいが，切り離さないこと． Usable for memos; do not detach.

Problem 1

In this problem, the *size* of a circuit is the number of gates in the circuit, and the *depth* of a circuit is the maximum number of gates connected in series within the circuit. You can use only AND, OR, NOT gates and constants 0 and 1. All AND and OR gates have two inputs. Thus if you need more inputs for an AND or an OR operation, then you need to combine multiple two-input gates. Fan-out is not limited. Assume that n is a positive integer and $N = 2^n$.

Answer the following questions.

(1) Design a circuit described as follows. The input consists of $n+1$ bits: K_0, K_1, ..., K_{n-1}, and Z. Let $k = \sum_{j=0}^{n-1} K_j 2^j$.

The output consists of N bits: M_0, M_1, ..., M_{N-1}. If $Z = 1$, then all the output bits should be 0. Otherwise, $M_j = 1$ for every $j \in \{0, 1, \ldots, k\}$, and $M_j = 0$ for every $j \in \{k+1, k+2, \ldots, N-1\}$. That is, the lowest $k+1$ bits are 1's, and the other bits are 0's.

For example, if $n = 3$, $(K_2, K_1, K_0) = (1, 0, 1)$ and $Z = 0$, then $k = 5$, so we should have

$$(M_7, M_6, \ldots, M_0) = (0, 0, 1, 1, 1, 1, 1, 1).$$

You must design a circuit of size $O(N)$ and depth $O(\log N)$. Explain your design, and prove that the size and the depth of the circuit satisfy the conditions.

(2) Design a circuit described as follows. The input consists of N bits: I_0, I_1, ..., I_{N-1}. If $I_k = 1$ and $I_j = 0$ for every $j \in \{k+1, k+2, \ldots, N-1\}$, then we call k *the highest position of 1*. If all the input bits are 0's, then -1 is the highest position of 1.

The output of the designed circuit consists of $n+1$ bits: K_0, K_1, ..., K_{n-1}, and Z. If the highest position of 1 is nonnegative, then Z must be 0, and $\sum_{j=0}^{n-1} K_j 2^j$ must be the highest position of 1. If the highest position of 1 is -1, then $Z = 1$ and $K_j = 0$ for every $j \in \{0, 1, \ldots, n-1\}$.

For example, if $n = 3$ and $(I_7, I_6, \ldots, I_0) = (0, 0, 1, 0, 1, 1, 0, 1)$, then the highest position of 1 is 5. Thus $(K_2, K_1, K_0) = (1, 0, 1)$ and $Z = 0$.

You must design a circuit of size $O(N)$ and depth $O(\log N)$. Explain your design, and prove that the size and the depth of the circuit satisfy the conditions.

You can use the following equation:

$$\sum_{m=1}^{\infty} m r^m = \frac{r}{(1-r)^2} \qquad \text{for } |r| < 1.$$

Problem 2

Consider the Java program below to sort an array A in an ascending order. M, N, and K are positive integers, and A is an array of N nonnegative integers where $0 \leq$ A[i] $<$ M for all i $\in \{0, \ldots, N-1\}$. In this program, list is a class of an integer list with the following methods.

- lst.size(): returns the number of elements in the list lst.
- lst.get(i): returns the element at the i-th position in the list lst (the position number starts from 0).
- lst.insert(i, x): inserts x to the list lst at the i-th position.

B is an array of size K, whose elements are all initialized to empty lists. Suppose that the execution time of each of the above methods is constant. You can ignore overflow errors.

```
1  void mysort(int M, int N, int K, int[] A, list[] B) {
2      for (int i = 0; i < N; i++) {
3          int m = A[i] * K / M;
4          int j = 0;
5          for (; j < B[m].size(); j++) {
6              if (A[i] <= B[m].get(j)) {
7                  break;
8              }
9          }
10         B[m].insert(j, A[i]);
11     }
12     int i = 0;
13     for (int m = 0; m < K; m++) {
14         for (int j = 0; j < B[m].size(); j++) {
15             A[i] = [ X ];
16             i = i + 1;
17         }
18     }
19 }
```

Answer the following questions.

(1) Answer an appropriate expression to fill the blank [X].

(2) Let C be the number of times the line 6 is executed. Answer the largest value of C in terms of N and K. Also, answer the expected value of C in terms of N and K, assuming that A[i] is distributed independently uniformly randomly over the set $\{0, \ldots, M-1\}$. Suppose that K $\ll$ M for this question.

(3) Explain how the expected running time of this program varies when K changes, assuming that A[i] is distributed independently uniformly randomly.

(4) Discuss advantages and disadvantages of this algorithm in comparison to the quicksort algorithm.

Problem 3

Let Σ be a finite alphabet (i.e., a finite set of letters). We say that a word $v \in \Sigma^*$ is a *subsequence* of a word $w = a_1 \cdots a_n \in \Sigma^*$ if $v = a_{i_1} \cdots a_{i_k}$ for some $k \geq 0$ and $1 \leq i_1 < \cdots < i_k \leq n$. For example, aab is a subsequence of $acbabc$ (let $k = 3, i_1 = 1, i_2 = 4$, and $i_3 = 5$). We write $v \preceq w$ if v is a subsequence of w. Answer the following questions.

(1) Give a non-deterministic finite automaton with at most 4 states that accepts the language:

$$\{w \in \{a, b, c\}^* \mid aab \preceq w\}.$$

(2) Suppose that $L \subseteq \Sigma^*$ is the language accepted by a deterministic finite automaton $\mathcal{A} = (Q, \Sigma, \delta, q_0, F)$ (where Q is a finite set of states, $\delta \in Q \times \Sigma \to Q$ is the transition function, $q_0 \in Q$ is the initial state, and $F \subseteq Q$ is the set of final states). Give a non-deterministic finite automaton that accepts the language:

$$\{w \in \Sigma^* \mid v \preceq w \text{ for some } v \in L\}.$$

(3) Suppose that $L \subseteq \Sigma^*$ is the language accepted by a deterministic finite automaton $\mathcal{A} = (Q, \Sigma, \delta, q_0, F)$. Assume that the transition function $\delta \in Q \times \Sigma \to Q$ is a total function. Give a deterministic finite automaton that accepts the language:

$$\{w \in \Sigma^* \mid v \in L \text{ for every } v \in \Sigma^* \text{ such that } v \preceq w\}.$$

(4) Prove the correctness of your answer for question (3) above.

Problem 4

Consider an n dimensional linear system $A\boldsymbol{x} = \boldsymbol{b}$, where A, $\boldsymbol{x}$, and $\boldsymbol{b}$ are an n dimensional real coefficient matrix, an n dimensional real vector of unknowns, and an n dimensional real given vector, respectively. Assume that A is not singular and $\boldsymbol{b} \neq \boldsymbol{0}$. The vector norm and the matrix norm used in this problem are the 2-norm and the matrix norm corresponding to the 2-norm, respectively.

Answer the following questions.

(1) Answer the definition of the *condition number* of A.

(2) Suppose that $\tilde{\boldsymbol{x}} \neq \boldsymbol{0}$ is an approximate solution of the linear system. Using the residual $\boldsymbol{r} = \boldsymbol{b} - A\tilde{\boldsymbol{x}}$ and $\tilde{\boldsymbol{x}}$, find a rank 1 matrix E which satisfies $(A + E)\tilde{\boldsymbol{x}} = \boldsymbol{b}$ exactly.

(3) Consider the effect of inaccuracy δA of A. Namely, the linear system becomes

$$(A + \delta A)(\boldsymbol{x} + \delta\boldsymbol{x}) = \boldsymbol{b},$$

where $\delta\boldsymbol{x}$ is the effect on the solution vector $\boldsymbol{x}$. Assume that $A + \delta A$ is not singular. In addition, δA has nothing to do with E in the question (2). By evaluating $\delta\boldsymbol{x}$, prove that the relative inaccuracy of $\boldsymbol{x}$ is related to that of A by the inequality:

$$\frac{\|\delta\boldsymbol{x}\|}{\|\boldsymbol{x} + \delta\boldsymbol{x}\|} \leq \mathrm{cond}_2(A)\frac{\|\delta A\|}{\|A\|},$$

where $\mathrm{cond}_2(A)$ is the condition number of A.

(4) Prove that for any n dimensional real singular matrix B, the relation

$$\|A - B\| \geq \frac{1}{\|A^{-1}\|}$$

is always satisfied. You may use the following fact: when an n dimensional real matrix C is singular, there is a non-zero real vector $\boldsymbol{y}$ such that $C\boldsymbol{y} = \boldsymbol{0}$.

余白 (blank page)

計算などに使ってもよいが，切り離さないこと． Usable for memos; do not detach.

余白 (blank page)

計算などに使ってもよいが，切り離さないこと． Usable for memos; do not detach.