

Specialized Subjects

10:00-12:30, Monday, August 22, 2022

Instructions

1. Do not open this booklet before the examination begins.
2. This booklet contains five problems. The number of pages is seven excluding this cover sheet and blank pages. If you find missing or badly printed pages, ask the proctor for exchange.
3. Answer three problems. You can select any three out of the five. Your answer to each problem should be written on a separate sheet. You may use the reverse side of the sheet if necessary.
4. Fill the top parts of your three answer sheets as instructed below. Before submitting your answer sheets, make sure that the top parts are correctly filled.

専 門 科 目

第 問

↑ Write the problem No.

受験番号				
------	--	--	--	--

↑ Write the examinee No.

5. Submit all the three answer sheets with the examinee number and the problem number, even if your answer is blank.
6. Answer either in Japanese or in English.
7. This booklet and the scratch paper must be returned at the end of the examination.
8. This English translation is supplemental and provided for the convenience of applicants. The Japanese version is the formal one.

Problem 1

Consider two circuits consisting of resistors (resistance R , R_1 , R_2), capacitors (capacitance C , C_1 , C_2), a coil (inductance L), and an ideal operational amplifier (input impedance ∞ , output impedance 0, amplification factor ∞), as shown in Fig. 1 and Fig. 2. Answer the following questions.

- (1) The circuit shown in Fig. 1 can be represented as a four-terminal network circuit using F-matrix as

$$\begin{pmatrix} V_1 \\ I_1 \end{pmatrix} = \begin{pmatrix} F_{11} & F_{12} \\ F_{21} & F_{22} \end{pmatrix} \begin{pmatrix} V_2 \\ I_2 \end{pmatrix}.$$

Write F-matrices corresponding to each of the blocks (a)-(c) in Fig. 1 using the Laplace operator s . The voltages and currents correspond to the orientations shown in Fig. 1.

- (2) Based on the results of (1), calculate the F-matrix of the entire circuit in Fig. 1 and the voltage transfer function $H_1(s) = V_2(s)/V_1(s)$.
- (3) Find the transfer function $H_2(s) = V_2(s)/V_1(s)$ for the transmission circuit shown in Fig. 2. The derivation process should also be clearly stated.
- (4) The circuits in Fig. 1 and Fig. 2 can serve as filters with the same characteristics concerning voltage amplitude. Suppose the ratios of capacitances and resistances are $C_2/C_1 = \alpha$ and $R_2/R_1 = \beta$, respectively. Show the relationship required between α and β as a necessary condition for the circuit in Fig. 2 to have the same characteristics as the circuit in Fig. 1.
- (5) Assume that condition in (4) holds and $C = C_1$. Express the values of L and R using R_1 , R_2 , C_1 , and C_2 , when the circuit in Fig. 1 has the same voltage amplitude characteristics as the circuit in Fig. 2.

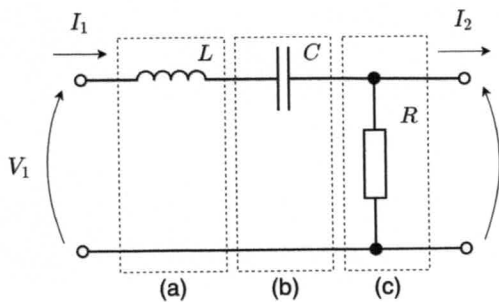


Fig. 1

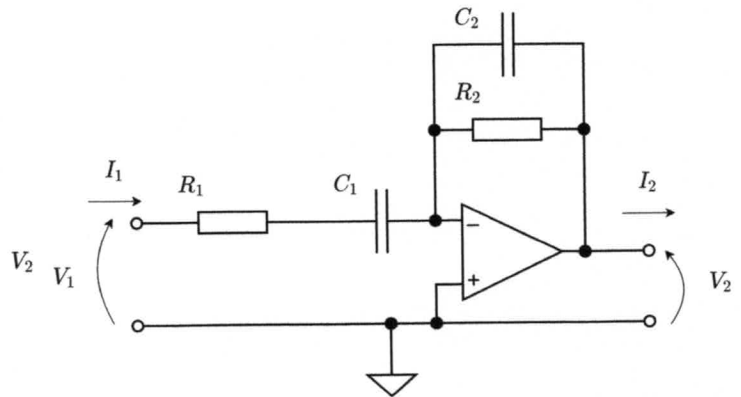
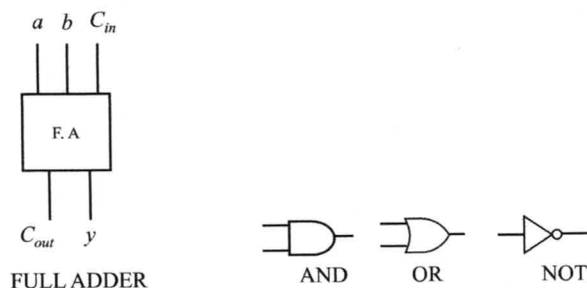


Fig. 2

Problem 2

Let us design a circuit that adds a 4-bit signed integer $A(A_{3:0})$ and a 3-bit signed integer $B(B_{2:0})$ and outputs a 4-bit signed integer $Y(Y_{3:0})$. Two's complement is used to represent signed integers. Let A_3 , B_2 , and Y_3 be the MSB (Most Significant Bit). Answer the following questions.

- (1) Show the ranges of possible values of A and B in decimal notation.
- (2) Show the truth table for the 1-bit full adder shown on the left side of the figure below. Let a and b be operands, C_{in} be the input carry, y be the output, and C_{out} be the output carry.
- (3) Show the circuit that computes $Y = A + B$ by combining the full-adder symbols used in (2). Also, show the input/output signal lines. You can use supply voltage V_{DD} and ground GND if necessary.
- (4) Let us add an overflow detection function to the circuit in (3). The additional circuit takes a 1-bit signal D as output, with $D = 1$ when an overflow occurs and $D = 0$ otherwise. Show the truth table of the circuit to output D by using necessary signals. Also, show the circuit that generates D by combining necessary signals and gates from $A_{3:0}$, $B_{2:0}$, $Y_{3:0}$, AND, OR, and NOT shown on the right side of the figure below. The number of gates and inputs can be increased, but the circuit should be simple.
- (5) Let us design a circuit to compute $Y = A - B$ instead of adding A and B . Show the circuit by combining full-adders, AND, OR, and NOT gates. Similar to (3), input/output signal lines should also be shown. You can use V_{DD} and GND if necessary.
- (6) Let us add an overflow detection function to the circuit in (5) as in (4). The additional circuit takes a 1-bit signal D as output, with $D = 1$ when an overflow occurs and $D = 0$ otherwise. Show the truth table of the circuit to output D by using necessary signals. Also, show the circuit that generates D by combining necessary signals and gates from $A_{3:0}$, $B_{2:0}$, $Y_{3:0}$, AND, OR, and NOT shown on the right side of the figure below. The number of gates and inputs can be increased, but the circuit should be simple.



Problem 3

Fig. 1 shows a weighted directed graph G_1 that represents a transportation network where a vertex is a distribution center and an edge is a route between two centers. Suppose that we compute the maximum flow or the maximum amount of goods to be transported from the origin s to the destination t in the transportation network. Each edge has a non-negative integer capacity that is the maximum amount of goods to be transported. A flow is defined as the amount of goods to be transported on each edge. Answer the following questions.

- (1) One of well-known algorithms to solve the maximum flow problem introduces a directed graph called the residual graph where we iteratively augment the flow on each edge, and find a path called an augmenting path from the origin to the destination until such a path cannot be found. An augmenting path is comprised of edges that have non-negative integer flows. A partial source code that implements this algorithm is shown in Algorithm 1. Fill out (A) in the *MaxFlow* function in the algorithm.
- (2) Answer the maximum flow of a transportation network G_1 shown in Fig. 1 by using Algorithm 1.
- (3) Answer the time complexity of the algorithm shown in Algorithm 1 and describe the reason that justifies your answer. As for the analysis, assume that a weighted directed graph G is given, use $|V|$, $|E|$, and *max_flow* as the number of vertices, the number of edges, and the maximum flow respectively.
- (4) Consider that we find a maximum matching for a bipartite graph. A bipartite graph $G = (V, E)$ is an undirected graph whose vertex set V can be divided into two independent subsets, X and Y , and every edge in E connects a vertex in X to one in Y . Suppose that M is a subset of E where any two edges do not share a common vertex, M is called a bipartite matching of a graph G . A maximum matching in a bipartite graph corresponds to the bipartite matching with the maximum number of edges.
 - (a) Suppose that we treat a problem for finding a maximum matching in a bipartite graph as the maximum flow problem, we need to extend the bipartite graph. Draw the structure of the extended graph including vertices, edges, and capacities.
 - (b) Suppose that a bipartite graph G_2 in Fig. 2 is given, answer the value of maximum matching and show one of matching patterns.
- (5) Describe a real-world use case that can leverage the max-flow problem and explain its reason briefly.

Algorithm 1

```

1 // A variable "graph" : a transportation network
2 // A variable "N" : the number of vertices in the graph
3 // A variable "s" : a vertex identifier for the origin
4 // A variable "t" : a vertex identifier for the destination
5 int MaxFlow(int graph[][], int N, int s, int t) {
6     int u, v;
7
8     // Initialize a residual graph "rGraph" with edge
9     // capacities of a given transportation network
10    int rGraph[][] = new int[N][N];
11
12    for (u = 0; u < N; u++)
13        for (v = 0; v < N; v++)
14            rGraph[u][v] = graph[u][v];
15
16    // A parent variable is an array where each element index represents
17    // a vertex identifier and stores the parent vertex identifier of the vertex
18    int parent[] = new int[N];
19    int max_flow = 0;
20
21    while (true) {
22        // Search an augmenting path in a residual graph with depth
23        // first search and store the found path into the "parent" variable.
24        // If all the vertices are found, exit this while loop.
25        if(DepthFirstSearch(rGraph, s, t, parent) == false) break;
26
27        // Firstly set the minimum flow in the augmenting path
28        // to the maximum value of an integer
29        int path_flow = Integer.MAX_VALUE;
30
31        for (v = t; v != s; v = parent[v]) {
32            u = parent[v];
33            path_flow = Math.min(path_flow, rGraph[u][v]);
34        }
35
36        // Update flows of all the edges on an augmenting path
37        // and their inverse edge in a residual graph
38        for (v = t; v != s; v = parent[v]) {
39            u = parent[v];
40            rGraph[u][v] -= path_flow;
41            rGraph[v][u] += (A);
42        }
43
44        // Update a maximum flow
45        max_flow += path_flow;
46    }
47
48    return max_flow;
49 }

```

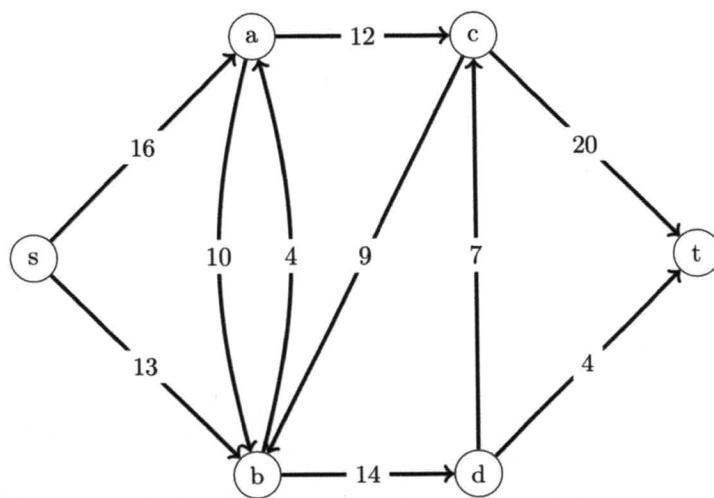


Fig. 1

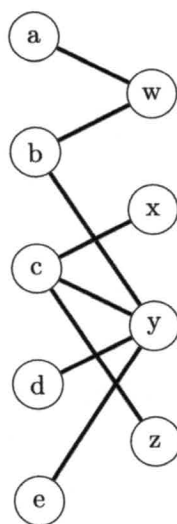


Fig. 2

Problem 4

Answer the following questions about routing in computer networks.

- (1) Flooding is a simple routing algorithm. Describe briefly how flooding performs routing. Also, give one advantage of using flooding.
- (2) Describe briefly what information a router exchanges with its neighboring routers in distance vector routing.
- (3) For distance vector routing, answer (a) the name of a representative algorithm for finding the shortest path from a source to each destination and (b) the name of a representative protocol used for interior gateway routing in the Internet.
- (4) BGP (Border Gateway Protocol) is a distance vector routing protocol. BGP is widely used as an exterior gateway routing protocol in the Internet. Describe major differences between distance vector routing protocols for an interior gateway and BGP.
- (5) Describe briefly the “count to infinity” problem in distance vector routing.
- (6) Consider that a network consists of V routers and E lines. For link state routing, answer (a) the name of a representative algorithm for finding the shortest path from a source to each destination and (b) the time complexity of the algorithm.
- (7) Consider updating all the link states for link state routing in the same network as in (6). Answer (a) the complexity of the total number of messages and (b) the complexity of the total size of messages.
- (8) OSPF (Open Shortest Path First) is a type of link state routing. The OSPF protocol is widely used for interior gateway routing in the Internet. Describe “area” in an OSPF’s autonomous system. Also, describe why “area” is needed.

Problem 5

Answer the following questions about discrete signal processing.

- (1) Show the definition of the \mathcal{Z} -transform $X(z)$ of a discrete signal sequence x_n ($n = 0, 1, 2, \dots$) defined for $n \geq 0$ where z is a complex number.
- (2) By using the definition, find the \mathcal{Z} -transform of a geometric sequence $x_n = p^n$ where p is a real number.
- (3) We consider the zero state response of the discrete time system shown in the figure below, where a, b and c are real numbers. Give $X(z)$ and $Y(z)$ by using the \mathcal{Z} -transform $Q(z)$ of the system's internal state q_n . Here, $X(z)$ and $Y(z)$ are the \mathcal{Z} -transforms of the system's input x_n and output y_n , respectively.
- (4) Give the transfer function $H(z)$ of the system.
- (5) Next, we find the system's frequency response by using the transfer function $H(z)$. Consider the input sequence $x_n = e^{jn\omega T}$ sampled at the interval T from a complex exponential function $x(t) = e^{j\omega t}$ ($t \geq 0$). Derive $Y(z)$ and then find the output sequence y_n .

