**Academic Year of 2021**
**Admission to the Master's Program**
**Department of Intelligence Science and Technology**
**Graduate School of Informatics, Kyoto University**
**(Fundamentals of Informatics)**
**(International Course)**

13:00 - 15:00, February 3, 2021

**NOTES**

1. This is the Question Booklet in 6 pages including this front cover.
2. Do not open the booklet until you are instructed to start.
3. After start, check the number of pages and notify proctors (professors) immediately if you find missing pages or unclear printings.
4. This booklet has 4 questions written in English. **Solve all questions.**
5. Write your answers in English, unless specified otherwise.
6. Read carefully the notes on the Answer Sheets as well.

Use one answer sheet for each of F1–1, F1–2, F2–1, and F2–2.

Q.1 Consider the system

$$\begin{aligned} x + 2y + z &= 3, \\ ay + 4z &= 10, \\ 2x + 6y + az &= b. \end{aligned}$$

(1) Find those values of $a$ for which the system has a unique solution.

(2) Find those pairs of values $(a, b)$ for which the system has more than one solution.

Q.2 Consider a 3×3 matrix

$$A = \begin{bmatrix} 3 & 1 & -1 \\ 1 & 3 & -1 \\ 1 & 1 & 1 \end{bmatrix}.$$

(1) Find all eigenvalues of $A$.

(2) Find a maximum set $S$ of linearly independent eigenvectors of $A$.

(3) Is $A$ diagonalizable? If yes, find $P$ such that $D = P^{-1}AP$ is diagonal. Otherwise, give a proof that $A$ is not diagonalizable.

<u>Use one answer sheet for each of F1–1, F1–2, F2–1, and F2–2.</u>

Q.1 Given the following information, compute $\partial z/\partial x$ and $\partial z/\partial y$. Using these results, determine $dz/dx$.

$$z = 2y - x^4y^2, \quad y = \cos(x)$$

Q.2 Evaluate the integral

$$\int_{-1}^{0} \left( \frac{1}{1-x} + \sqrt{1-x} \right) dx.$$

Q.3 Find the maximum rate of change of $f(x, y) = \sqrt{x^2 + y^3}$ at $(1, 1)$ and the direction in which this maximum rate of change occurs.

<u>Use one answer sheet for each of **F1-1**, **F1-2**, **F2-1**, and **F2-2**.</u>

Q. Answer the following questions about binary search trees.

(1) Draw one of the binary search trees with the smallest height that have nine elements {8,13,10,7,1,6,17,14,11}.

(2) Draw one of the highest binary search trees with the same set of elements as those in (1).

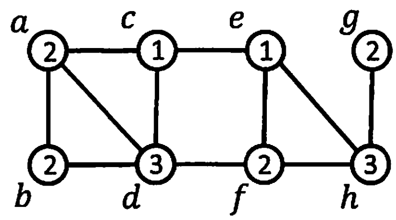
(3) When one of the elements is queried with the equal probability in the binary search tree that you answered in (1), what is the expected number of nodes that are visited to find the element?

(4) Suppose a binary search tree that is a complete binary tree with $2^d - 1$ distinct elements, where $d$ is a positive integer. When one of the elements is queried with the equal probability, what is the expected number of nodes that are visited to find the element?

<u>Use one answer sheet for each of F1–1, F1–2, F2–1, and F2–2.</u>

Q. Let $G = (V, E)$ be a simple undirected graph, where $V$ is the set of vertices and $E$ is the set of edges. A positive integer $w(v)$, called a *weight*, is associated with each vertex $v \in V$. A subset $S \subseteq V$ is called an *independent set* if for any pair of vertices $u, v \in S$, $(u, v) \notin E$ holds. The weight of an independent set $S$ is defined as $w(S) = \sum_{v \in S} w(v)$. An independent set of the maximum weight is called a *maximum independent set* of $G$ and its weight is denoted by $\mathrm{IN}(G)$.

(1) Give a maximum independent set and $\mathrm{IN}(G_1)$ of the following graph $G_1$. Letters "$a$" – "$h$" are the names of vertices and an integer written inside a vertex is its weight.

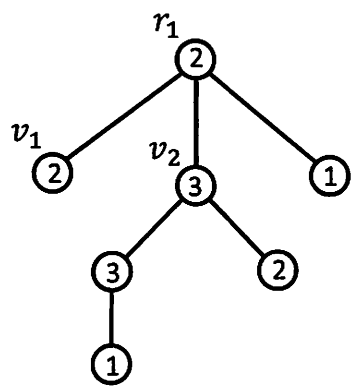



$G_1$

(2) From now, we consider only rooted trees. Let $T$ be a rooted tree. For a vertex $v$, $T(v)$ denotes the subtree of $T$ rooted at $v$. Let

- $X(v) := \max\{w(S) \mid (S \text{ is an independent set of } T(v)) \wedge (v \in S)\}$,
- $Y(v) := \max\{w(S) \mid (S \text{ is an independent set of } T(v)) \wedge (v \notin S)\}$, and
- $Z(v) := \mathrm{IN}(T(v))$.

Give $X(v_1)$, $Y(v_1)$, $Z(v_1)$, $X(v_2)$, $Y(v_2)$, and $Z(v_2)$ for the following tree $T_1$, where $r_1$ denotes the root of $T_1$.




$T_1$

(Continued on the next page)

(3) The following Algorithm 1 is a pseudo-code of an algorithm computing $\mathrm{IN}(T)$ of a given tree $T$ rooted at $r$. Fill the blanks (a)–(f). If necessary, use the notation $C(v)$ to denote the set of $v$'s children.

---

**Algorithm 1**

---

**Input:** A tree $T$ rooted at $r$.
**Output:** $\mathrm{IN}(T)$.

1: Let all the vertices be unmarked.
2: **while** there is an unmarked vertex **do**
3:     Let $v$ be an unmarked vertex that has no unmarked child.
4:     Mark $v$.
5:     **if** $v$ is a leaf **then**
6:         Let $X(v) :=$ [ (a) ], $Y(v) :=$ [ (b) ], and $Z(v) :=$ [ (c) ].
7:     **else**
8:         Let $X(v) :=$ [ (d) ], $Y(v) :=$ [ (e) ], and $Z(v) :=$ [ (c) ].
9:     **end if**
10: **end while**
11: Output [ (f) ].

---

(4) Suppose that, to determine the order of vertices to be processed at line 3 of Algorithm 1, we do the depth-first search from the root $r$. For this purpose, which is appropriate for vertex ordering: (i) preorder, (ii) inorder, or (iii) postorder? Answer with reasons.

(5) Let $n$ be the number of vertices of an input tree $T$. Show the time-complexity of Algorithm 1 with reasons.