Academic Year of 2023
Admission to the Master's Program
Intelligence Science and Technology Course
Graduate School of Informatics, Kyoto University
(Fundamentals of Informatics)
(International Program)

13:00 - 15:00, February 8, 2023

NOTES

1. This is the Question Booklet in 5 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 the notes on the Answer Sheets as well.

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

Answer the following questions about a real matrix

$$\mathbf{A} = \begin{pmatrix} 1 & a & 0 \\ b & 3 & 0 \\ -1 & 1 & 2 \end{pmatrix}.$$

Q.1 Derive the constraint on a and b such that the eigenvalues of $\mathbf{A}$ are different real numbers.

Q.2 Let λ_1 and λ_2 be the maximum and the minimum eigenvalues of $\mathbf{A}$, respectively. Let $\mathbf{v}_1$ and $\mathbf{v}_2$ be the eigenvectors corresponding to λ_1 and λ_2, respectively. When $a = 6$ and $b = 4$, answer the following questions.

(1) Derive λ_1, λ_2, $\mathbf{v}_1$, and $\mathbf{v}_2$.

(2) Derive an orthonormal basis of the subspace $W = \{\mathbf{x} \in \mathbb{R}^3 \,|\, \mathbf{x} = p\mathbf{v}_1 + q\mathbf{v}_2\}$, where p and q are real numbers.

(3) Compute the orthogonal projection of $\mathbf{r} = [1,\ 4,\ -6]^\top$ onto the subspace W, where $\top$ is the transpose operator.

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

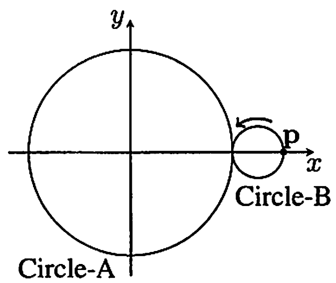
Q.1 Derive the angles of intersection between the curves $2x^2 + y^2 = 20$ and $4y^2 - x^2 = 8$.

Q.2 Evaluate the following limit.

$$\lim_{x \to 1} x^{\frac{1}{x-1}}$$

Q.3 Consider a circle of radius 4 (Circle-A) and another circle of radius 1 (Circle-B). Answer the following questions, when Circle-A stays static while Circle-B completely rolls along the circumference of Circle-A once without slipping.

(1) Sketch the curve traced by the fixed point p on the circumference of Circle-B (see the right figure).
(2) Derive the length of the curve in (1).
(3) Derive the area enclosed by the curve in (1).



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

Q. Answer the following questions about *merge sort*. We assume that sorting is in ascending order.

(1) Given an array $A_1 = [5, 3, 20, 1, 8]$, illustrate the procedure of *merge sort* for A_1.

(2) With the procedure answered in (1), show the time complexity and the space complexity of *merge sort* for an array of n elements with reasons.

(3) Explain the difference between *merge sort* for arrays and *merge sort* for linked lists in terms of computational complexity.

(4) Explain two advantages and one disadvantage of *merge sort* for arrays in comparison with *quick sort* for arrays.

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

Q. Let $G(V, E)$ be a simple directed graph where $V = \{v_1, \ldots, v_n\}$, $m = |E|$, and each edge $(v_i, v_j) \in E$ has a positive integer weight $d(v_i, v_j)$. The following is the core part of a pseudocode of an algorithm for computing the weights of the shortest paths from a vertex v_1 to all vertices, where a path is called the shortest if the sum of weights of the edges is minimized, and P and D denote one-dimensional integer arrays. Note that the weight of the shortest path from v_1 to each v_i will be stored in $D[i]$ after the execution of this algorithm, and $+\infty$ denotes a sufficiently large integer which represents that there is no path from v_1 to v_i.

```
for i = 1 to n do begin D[i] ← +∞;  P[i] ← −1; end
D[1] ← 0;  P[1] ← 0;
for k = 1 to n − 1 do begin
  for all (v_i, v_j) ∈ E do begin
    if D[j] > D[i] + d(v_i, v_j) then begin
      D[j] ← D[i] + d(v_i, v_j);
      P[j] ← i;
    end
  end
end
```

Answer the following questions.

(1) Let $n = 4$, $E = \{(v_1, v_2), (v_1, v_3), (v_2, v_4)\}$, and $d(v_i, v_j) = 1$ for each $(v_i, v_j) \in E$. Show $D[1], \ldots, D[4]$ after the execution of the algorithm.

(2) Suppose that $d(v_i, v_j)$ can be obtained in $O(1)$ time for each $(v_i, v_j) \in E$. Analyze the time complexity of this algorithm.

(3) Explain how to modify the code for outputting a shortest path from v_1 to a specified vertex v_i, by using P. If there is no path from v_1 to v_i, "No Path" should be outputted.

(4) In some cases, it is not necessary to iterate the outer loop (i.e., loop on 'k') $n - 1$ times. For a graph $G(V, E)$, let $\mathrm{minrep}(G(V, E))$ denote the minimum number of iterations for any ordering of edges so that the shortest path weights from v_1 to all nodes in V are computed. Suppose that the weight of each edge is 1 (i.e., $d(v_i, v_j) = 1$ for each $(v_i, v_j) \in E$). Show E of a graph $G(V, E)$ for each of the following cases:
(i) $m = n(n-1) - 1$ and $\mathrm{minrep}(G(V, E)) = 2$,
(ii) $m = \frac{n(n-1)}{2} + (n - 1)$ and $\mathrm{minrep}(G(V, E)) = n - 1$.