平成28年度
東京大学大学院情報理工学系研究科 コンピュータ科学専攻 入学試験問題

専門科目 I

平成27年8月25日
10:00 – 12:30

注意事項

(1) 試験開始の合図があるまで，この問題冊子を開けないこと．

(2) 4 題すべてに答えよ．問題ごとに指定された解答用紙を使用すること．

(3) 解答用紙および問題冊子は持ち帰らないこと．

Specialized Subjects I

10:00 – 12:30, August 25, 2015

Entrance Examination (AY 2016)
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.

下欄に受験番号を記入すること．
Write your examinee's number in the box below.

受験番号	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.

問題 1

2つのn次元整数ベクトルx, yについて，これらの第i要素をそれぞれx_i, y_iとするとき，任意の$i \in [1, n]$について$x_i \equiv y_i \pmod 2$が成り立つことを$x \doteq y$と書きあらわすことにする．同様に，2つの$n \times n$整数行列A, Bについて，これらのi行j列要素をそれぞれa_{ij}, b_{ij}とするとき，任意の$i, j \in [1, n]$について$a_{ij} \equiv b_{ij} \pmod 2$が成り立つことを$A \doteq B$と書きあらわすことにする．

なお以下では，各要素が0または1であるベクトルを「0-1ベクトル」とよぶ．また，各要素が互いに独立に，それぞれ等確率で0または1の値をとるベクトルを「ランダムな0-1ベクトル」とよぶ．さらに，零ベクトル（全要素が0であるベクトル）をo，零行列（全要素が0である行列）をOと表すものとする．

以下の問いに答えよ．

(1) ランダムな0-1ベクトル$x \in \{0,1\}^3$を考える．このとき，

$$\begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix} \cdot x \quad \doteq \quad o$$

となる確率を示せ．

(2) $A \doteq O$を満たさない$n \times n$の整数行列A，およびランダムな0-1ベクトル$x \in \{0,1\}^n$を考える．このとき，$A \cdot x \doteq o$となる確率は1/2以下であることを示せ．

(3) $A \cdot B \doteq C$を満たさない$n \times n$の整数行列の3つ組A, B, C，およびランダムな0-1ベクトル$x \in \{0,1\}^n$を考える．このとき，$A \cdot B \cdot x \doteq C \cdot x$となる確率は1/2以下であることを示せ．

(4) 任意の3つの$n \times n$の整数行列A, B, Cに対し，条件$A \cdot B \doteq C$を満たしているならば必ず「条件が満たされている」と回答し，満たされていないならば9/10よりも高い確率で「条件が満たされていない」と回答することができる$O(n^2)$のアルゴリズムを示せ．

Problem 1

Given two n-dimensional integer vectors x and y, let us write $x \doteq y$ if we have $x_i \equiv y_i \pmod 2$ for each $i \in [1, n]$. Here x_i and y_i denote the i-th elements of the vectors x and y, respectively. Likewise, given two $n \times n$ integer matrices A and B, we write $A \doteq B$ if we have $a_{ij} \equiv b_{ij} \pmod 2$ for each $i, j \in [1, n]$. Here a_{ij} and b_{ij} denote the elements in the i-th row and the j-th column of the matrices A and B, respectively.

In what follows, a vector all of whose elements are either 0 or 1 is referred to as a '0-1 vector'. A vector each of whose elements is chosen from 0 and 1, with equal probabilities and independently from the other elements, is referred to as a 'random 0-1 vector'. The zero vector (*i.e.*, the vector all of whose elements are 0) is denoted by o; and the zero matrix (*i.e.*, the matrix all of whose elements are 0) is denoted by O.

Answer the following questions.

(1) Let $x \in \{0, 1\}^3$ be a random 0-1 vector. Derive the probability with which

$$\begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix} \cdot x \doteq o$$

holds.

(2) Let A be an $n \times n$ integer matrix that does not satisfy $A \doteq O$, and $x \in \{0, 1\}^n$ be a random 0-1 vector. Prove that the probability with which $A \cdot x \doteq o$ holds is no greater than $1/2$.

(3) Let A, B and C be $n \times n$ matrices that do not satisfy $A \cdot B \doteq C$, and $x \in \{0, 1\}^n$ be a random 0-1 vector. Prove that the probability with which $A \cdot B \cdot x \doteq C \cdot x$ holds is no greater than $1/2$.

(4) Show an $O(n^2)$ algorithm that: takes three $n \times n$ integer matrices A, B and C; always answers "SATISFIED" if the condition $A \cdot B \doteq C$ is satisfied; and answers "NOT SATISFIED", with a probability greater than $9/10$, if the condition $A \cdot B \doteq C$ is not satisfied.

問題 2

$\mathbb{N} = \{0, 1, 2, \dots\}$ を非負整数全体の集合とする．

$\mathcal{A} = (Q, \Sigma, \delta, q_0, F)$ を決定性有限オートマトンとする．ここで Q は有限の状態集合，Σ は有限の文字集合，$\delta\colon Q \times \Sigma \to Q$ は遷移関数，$q_0 \in Q$ は初期状態，$F \subseteq Q$ は受理状態の集合である．以下，Σ 上の有限語全体の集合を Σ^* と書く（すなわち $\Sigma^* = \bigcup_{n\in\mathbb{N}} \Sigma^n$）．また ε は空語をあらわす．

決定性有限オートマトンの最小化のための次のような構成を考えよう．Q 上の二項関係の列 $R_0, R_1, R_2 \dots$（ただし各 $n \in \mathbb{N}$ に対して $R_n \subseteq Q \times Q$）を次のように帰納的に定義する．

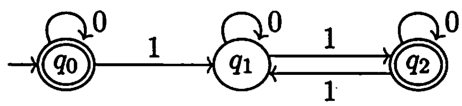
$$R_0 = Q \times Q \qquad R_{n+1} = \Phi(R_n) \tag{†}$$

ここで Φ は，二項関係 $R \subseteq Q \times Q$ に対して次の二項関係 $\Phi(R) \subseteq Q \times Q$ を返すような関数である．

$$(q, q') \in \Phi(R) \iff \left(\begin{array}{l} q \in F \iff q' \in F \text{ , かつ,} \\ \text{任意の } a \in \Sigma \text{ に対して } \big(\delta(q, a), \delta(q', a)\big) \in R \text{ .} \end{array} \right)$$

以下の問いに答えよ．

(1) 決定性有限オートマトン $\mathcal{A}$ が下に図示されたものであるとき，各 $n \in \mathbb{N}$ に対して二項関係 R_n を求めよ．ただし $\Sigma = \{0, 1\}$ とし，二重円 ◎ は受理状態をあらわす．



(2) 関数 Φ が単調であること，すなわち，$R \subseteq R'$ ならば $\Phi(R) \subseteq \Phi(R')$ となること，はすぐに確かめられる．この事実および，R_0 が Q 上の二項関係の中で最大のものであることを用いて，(†) で定義された二項関係の列 $R_0, R_1, R_2 \dots$ が

$$R_0 \supseteq R_1 \supseteq R_2 \supseteq \cdots \tag{‡}$$

をみたすことを示せ．

(3) 下降列 (‡) の極限 $\bigcap_{n\in\mathbb{N}} R_n$ を R_ω と書く．下降列 (‡) が有限ステップで極限にいたるかどうか，すなわち，ある非負整数 $n \in \mathbb{N}$ が存在して $R_n = R_{n+1} = R_{n+2} = \cdots = R_\omega$ となるか答えよ．証明または反例も与えよ．

(4) 遷移関数 δ の有限語への拡張を $\delta^*\colon Q \times \Sigma^* \to Q$ と書く．すなわち $q \in Q$，$a \in \Sigma$，$w \in \Sigma^*$ に対して

$$\delta^*(q, \varepsilon) = q \ , \qquad \delta^*(q, aw) = \delta^*\big(\delta(q, a), w\big) \ .$$

任意の非負整数 n（ただし $n \geq 1$）について次が成立することを，n に関する帰納法によって示せ．

状態 $q, q' \in Q$ が $(q, q') \in R_n$ となるとき，長さ $n-1$ の任意の語 $w \in \Sigma^{n-1}$ について

$$\delta^*(q, w) \in F \iff \delta^*(q', w) \in F$$

が成り立つ．

(5) 二つの状態が「同じ言語を受理する」二項関係を $\approx$ と書く．すなわち

$$(q, q') \in \approx \iff \big(\text{任意の語 } w \in \Sigma^* \text{ に対して，} \ \delta^*(q, w) \in F \iff \delta^*(q', w) \in F\big)$$

ふたつの二項関係 R_ω と $\approx$ の間に，包含関係 $R_\omega \subseteq \approx$ が成り立つことを証明せよ．

(6) この逆，すなわち $\approx \subseteq R_\omega$ が成り立つことを証明せよ．ただし Φ の単調性を用いて良い．また，次の簡単に確かめられる事実を用いてよい：ふたつの二項関係 $\approx$ と $\Phi(\approx)$ の間に，包含関係 $\approx \subseteq \Phi(\approx)$ が成立する．

Problem 2

Let $\mathbb{N} = \{0, 1, 2, \dots\}$ denote the set of all nonnegative integers.

Let $\mathcal{A} = (Q, \Sigma, \delta, q_0, F)$ be a deterministic finite automaton (DFA). Here Q is a finite set of states; Σ is a finite alphabet; $\delta \colon Q \times \Sigma \to Q$ is a transition function; $q_0 \in Q$ is an initial state; and $F \subseteq Q$ is the set of accepting states. In what follows we let Σ^* denote the set of finite words over Σ (that is, $\Sigma^* = \bigcup_{n \in \mathbb{N}} \Sigma^n$), and ε denote the empty word.

Let us consider the following construction that minimizes DFAs. We define a sequence $R_0, R_1, R_2 \dots$ of binary relations over Q (hence $R_n \subseteq Q \times Q$ for each $n \in \mathbb{N}$), in the following inductive way.

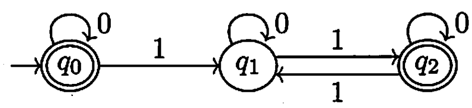
$$R_0 = Q \times Q \qquad R_{n+1} = \Phi(R_n) \tag{$\dagger$}$$

Here Φ is the function that, given $R \subseteq Q \times Q$, returns the following binary relation $\Phi(R) \subseteq Q \times Q$.

$$(q, q') \in \Phi(R) \iff \left(\begin{array}{l} q \in F \iff q' \in F \ ; \\ \text{and for each } a \in \Sigma, \quad \big(\delta(q,a),\, \delta(q',a)\big) \in R \ . \end{array} \right)$$

Answer the following questions.

(1) Let a DFA $\mathcal{A}$ be the one depicted below. Describe the binary relation R_n for each $n \in \mathbb{N}$. Here $\Sigma = \{0, 1\}$, and a double circle $\circledcirc$ designates an accepting state.



(2) It is straightforward to see that the function Φ is monotone, that is, $R \subseteq R'$ implies $\Phi(R) \subseteq \Phi(R')$. Use this fact, and the fact that R_0 is the greatest binary relation over Q, in showing the following: the sequence $R_0, R_1, R_2 \dots$ defined in ($\dagger$) satisfies

$$R_0 \supseteq R_1 \supseteq R_2 \supseteq \cdots \ . \tag{$\ddagger$}$$

(3) Let R_ω be the limit $\bigcap_{n \in \mathbb{N}} R_n$ of the descending chain ($\ddagger$). Answer whether the chain ($\ddagger$) reaches its limit within finitely many steps, that is, whether there is a nonnegative integer $n \in \mathbb{N}$ such that $R_n = R_{n+1} = R_{n+2} = \cdots = R_\omega$. Give a proof or a counterexample, too.

(4) We extend the transition function δ to finite words and define the function $\delta^* \colon Q \times \Sigma^* \to Q$ by: for $q \in Q$, $a \in \Sigma$ and $w \in \Sigma^*$,

$$\delta^*(q, \varepsilon) = q \ , \qquad \delta^*(q, aw) = \delta^*\big(\delta(q,a),\, w\big) \ .$$

Prove, by induction, that the following holds for each integer n such that $n \geq 1$.

> If two states $q, q' \in Q$ satisfy $(q, q') \in R_n$, then for any word $w \in \Sigma^{n-1}$ of length $n - 1$ we have
>
> $$\delta^*(q, w) \in F \iff \delta^*(q', w) \in F \ .$$

(5) Let $\approx$ be the binary relation between states that they "accept the same language." That is,

$$(q, q') \in \approx \iff \big(\text{for each word } w \in \Sigma^*, \quad \delta^*(q,w) \in F \iff \delta^*(q', w) \in F\big)$$

Prove that, between the two binary relations R_ω and $\approx$, we have inclusion $R_\omega \subseteq \approx$.

(6) Prove that the converse holds, that is, $\approx \subseteq R_\omega$. Here you can use that Φ is monotonic. You can also use the following fact (that is easily verified): between the two binary relations $\approx$ and $\Phi(\approx)$, we have inclusion $\approx \subseteq \Phi(\approx)$.

問題 3

生物の遺伝情報（遺伝子）はその細胞の中の常染色体に主に収納されている．細胞の中で常染色体は父親由来，母親由来それぞれ1本ずつで1対となっており，それぞれの細胞は決まった数の常染色体の対を持っている．常染色体の対においてそれぞれの常染色体が父親または母親のどちらに由来するかは，遺伝子の働きに影響を及ぼさない．

生物が子孫を残すとき，父親から子に対して，このような常染色体の対からいずれか1本が等確率で選択されて伝えられる．母親からも同様に1本が伝えられ，これら2本が子の常染色体の対をなす．

今，ある生物のある常染色体上のある遺伝子にA，aの二つの型（遺伝学ではアリルと呼ばれる）があるとし，以後この遺伝子に注目する．

この遺伝子を生物がどのように持っているか（遺伝子型）は，Aを2個，Aとa一つずつ，aを2個の3パターンのうちいずれかとなる．

以下の問いに答えよ．

(1) 両親の遺伝子型がAa及びAaの時，子の遺伝子型がAAの確率を求めよ．

(2) 父親，子の遺伝子型がそれぞれAa, aaであるとする．また，母親の2本の常染色体のそれぞれにおいて，アリルA及びaがそれぞれ確率2/3，1/3で互いに独立に出現するとする．このとき，母親の遺伝子型として，もっともらしさが最大のものを求めよ．

サイズが十分に大きい無作為交配集団（雌雄間の交配は無作為）を考える．この集団においては世代の間で重複がなく，移住もなく，遺伝子の突然変異もないものとする．また（第0世代を含む）各世代において，対となっている染色体の間でアリルの出現は独立であるとする．第n世代誕生後のアリルAとaの頻度をそれぞれp_nとq_nと表す．ただし，$p_n + q_n = 1$である．この時3つの遺伝子型AA, Aa, aaの頻度は，それぞれp_n^2，$2p_nq_n$，q_n^2となる．

以下の問いに答えよ．

(3) アリルaを持つことが生存に不利に働く状況を考える．3つの遺伝子AA，Aa，aaそれぞれについて，成体になり子孫を残すまで生存する確率（適応度と呼ぶ）を1，$1-s$，$1-2s$とおく．このとき，第$(n+1)$世代誕生後におけるaの頻度q_{n+1}に対し，$q_n - q_{n+1}$をq_nとsを使って表せ．

(4) 3つの遺伝子型AA，Aa，aaそれぞれについて，適応度を1，1，0とおく．第0世代誕生時のaの頻度が1/100のとき，第n世代誕生後におけるaの頻度が1/10000以下になるような最小のnを求めよ．

Problem 3

Genes, i.e., genetic information of an organism, are contained largely in the autosomal chromosomes in its cells. In a cell, a pair is formed by one autosomal chromosome derived from the father, and one derived from the mother. A cell in an organism has a constant number of such pairs of autosomal chromosomes. In a pair of autosomal chromosomes, the origin of each autosomal chromosome (that is, whether it is derived from the father or from the mother) does not cause any difference in the function of the genes.

When organisms procreate, one autosomal chromosome from the father's pair is randomly chosen (with equal probabilities) and is delivered to the child. The same occurs from the mother to the child; and the resulting two autosomal chromosomes form the pair of the child.

Let us now imagine a species, each member of which has a certain autosomal chromosome that includes a certain gene that is either of the type A or of the type a. These types are called alleles in genetics. Let us focus on this specific gene in what follows.

It follows that genotypes, i.e., possible patterns of the gene of each member of the species, are the following three: two A's; one A and one a; and two a's.

Answer the following questions.

(1) Give the probability with which the genotype of a child is AA, assuming that the genotypes of its parents are both Aa.

(2) Assume that the genotype of a child is aa and that of its father is Aa. Assume further that, on each of the two autosomal chromosomes of the mother, the alleles A and a appear independently from each other, with the probabilities $2/3$ and $1/3$, respectively. Give the genotype of the mother that is most likely.

Consider a random mating population that is large enough, in which the mating between a male and a female occurs randomly. Suppose that there is no overlap between different generations, no migration, or no mutation, in the population. Suppose also that an allele (A or a) appears independently from each other in a pair of autosomal chromosomes, in each generation (including the 0-th one). Let us express the frequencies of having alleles A and a after the birth of the n-th generation by p_n and q_n, respectively. Note here that $p_n + q_n = 1$. It follows that the frequencies of having three genotypes—AA, Aa and aa—are p_n^2, $2p_nq_n$, and q_n^2, respectively.

Answer the following questions.

(3) Consider a situation in which having the allele a is disadvantageous in survival. For each of the three genotypes AA, Aa, and aa, let us define the probability of survival until an organism becomes adult and procreates (the probability is called fitness) by 1, $1-s$, and $1-2s$, respectively.

Concerning the frequency q_{n+1} of having the allele a after the birth of the $(n+1)$-th generation, express $q_n - q_{n+1}$ using q_n and s.

(4) For each of the genotypes AA, Aa, and aa, let their fitness be 1, 1, and 0, respectively. Assume that the frequency of the allele a is $1/100$ when the 0-th generation is born. Give the minimum of n such that the frequency of the allele a is no greater than $1/10000$ after the n-th generation is born.

問題 4

プロセス $p_1, p_2, \ldots, p_M$ が，資源 $r_1, r_2, \ldots, r_N$ のうちいくつかを用いながら作業を行うようなシステムを考える．ただし以下を仮定する．

- 各プロセス p_i が作業を完了するためには，n_i 個の資源を同時に用いることを必要とするものとする．
- 資源は占有されるが，消費されない．すなわち，あるプロセス p_i が資源 r_j を用いて作業を行っても，p_i が作業を完了したあとは，他のプロセス p_k が同じ資源 r_j を用いることができる．
- 資源 $r_1, r_2, \ldots, r_N$ はすべて同種であるとする．すなわち，各プロセスにとっては資源の個数のみが関心事である．
- 2つ以上のプロセスが1つの資源を同時に使用することはできないものとする．

以下の問いに答えよ．

(1) $M = N = 2$ かつ $n_1 = n_2 = 2$ であるとする．このシステムで発生しうるデッドロックの例を1つ挙げて説明せよ．特に，資源の割り当てと解放がどのようなタイミングで行われたときにデッドロックが発生するかを述べること．

(2) 問い (1) のシステムでデッドロックを防止するためにはどのような方法があるか．例を1つ挙げて説明せよ．また，その方法に制約や不利益があればこれについて述べよ．

(3) $M = 3$，$N = 4$ かつ $n_1 = n_2 = n_3 = 2$ であるとする．このシステムはデッドロックを発生するか．理由もつけて答えよ．

(4) M, N を1以上の任意の整数とし，すべての $i \in [1, M]$ に対して $1 \leq n_i \leq N$ とする．また $n = n_1 + n_2 + \cdots + n_M$ と定義する．デッドロックが発生しないことが任意の $n_1, \ldots, n_M$ のえらび方に対して保証されるような n の最大値を M，N を用いて表せ．理由も述べよ．

Problem 4

Consider a system in which processes $p_1, p_2, \ldots, p_M$ operate using some of resources $r_1, r_2, \ldots, r_N$. We assume the following.

- Each process p_i needs to simultaneously use n_i resources in order to complete its task.
- Resources get occupied but they do not get consumed. That is, even if a process p_i operates using a resource r_j, another process p_k can use the same resource r_j after p_i completes its task.
- Resources $r_1, r_2, \ldots, r_N$ are all of the same kind. That is, only the number of resources matters to each process.
- It is prohibited that two processes use the same resource at the same time.

Answer the following questions.

(1) Let $M = N = 2$ and $n_1 = n_2 = 2$. Describe an example of deadlock that may occur in this system. Describe, in particular, what timing of assignment and release of resources leads to deadlock.

(2) Describe an example of a technique that prevents deadlock in the system in Question (1). Discuss restrictions and disadvantages of the technique, if any.

(3) Let $M = 3$, $N = 4$ and $n_1 = n_2 = n_3 = 2$. Answer whether this system exhibits deadlock. Explain why.

(4) Let M, N be arbitrary integers that are no less than 1; and assume that $1 \leq n_i \leq N$ holds for each $i \in [1, M]$. Let us define $n = n_1 + n_2 + \cdots + n_M$. Give, in terms of M, N, the maximum value of n for which the system is guaranteed not to exhibit deadlock for each choice of $n_1, \ldots, n_M$. Explain why.

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.

平成28年度
東京大学大学院情報理工学系研究科 コンピュータ科学専攻 入学試験問題

専門科目 II

平成27年8月25日
13:30 – 16:00

注意事項

(1) 試験開始の合図があるまで, この問題冊子を開けないこと.

(2) 4題すべてに答えよ. 問題ごとに指定された解答用紙を使用すること.

(3) 解答用紙および問題冊子は持ち帰らないこと.

Specialized Subjects II

13:30 – 16:00, August 25, 2015

Entrance Examination (AY 2016)
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.

下欄に受験番号を記入すること.
Write your examinee's number in the box below.

受験番号	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.

問題 1

d 次元の実ベクトルで表されるパターン $\boldsymbol{x} \in \mathbb{R}^d$ を，2つのクラス $y = +1, -1$ のいずれかに分類するパターン認識の問題を考える．分類器の学習のために，n 個の訓練標本

$$\{ (\boldsymbol{x}_i, y_i) \mid \boldsymbol{x}_i \in \mathbb{R}^d,\ y_i \in \{+1, -1\},\ i = 1, \ldots, n \}$$

が与えられるとする．ここで，$(\boldsymbol{x}_i, y_i)$ はパターン $\boldsymbol{x}_i$ がクラス y_i に属するという意味である．
以下の問いに答えよ．

(1) n 個の訓練標本のうちクラス $+1$，-1 に属するパターンの数をそれぞれ n_+，n_- で表す．クラス $+1$ に属する n_+ 個のパターンの平均ベクトル $\boldsymbol{c}_+$ と，クラス -1 に属する n_- 個のパターンの平均ベクトル $\boldsymbol{c}_-$ を求めよ．

(2) $\|\boldsymbol{x} - \boldsymbol{c}_+\| < \|\boldsymbol{x} - \boldsymbol{c}_-\|$ ならばパターン $\boldsymbol{x}$ をクラス $+1$ に，$\|\boldsymbol{x} - \boldsymbol{c}_+\| > \|\boldsymbol{x} - \boldsymbol{c}_-\|$ ならばパターン $\boldsymbol{x}$ をクラス -1 に割り当てるという分類器を考える．ただし $\|\cdot\|$ はユークリッドノルムを表す．この分類器によってクラス $+1$ に分類されるパターンが属する領域と，クラス -1 に分類されるパターンが属する領域の境界を表す方程式を求めよ．

パラメータ $\boldsymbol{w} \in \mathbb{R}^d$ を用いて，$\boldsymbol{w}^\top \boldsymbol{x} > 0$ ならばパターン $\boldsymbol{x}$ をクラス $+1$ に，$\boldsymbol{w}^\top \boldsymbol{x} < 0$ ならばパターン $\boldsymbol{x}$ をクラス -1 に割り当てるという線形分類器を定義する．ただし，$(\cdot)^\top$ は転置を表す．$y_i \boldsymbol{w}^\top \boldsymbol{x}_i$ で定義される値を i 番目の訓練標本 $(\boldsymbol{x}_i, y_i)$ に対するマージンとよぶことにすると，この線形分類器がパターン $\boldsymbol{x}_i$ を正しくクラス y_i に分類するための条件は，マージンを用いて $y_i \boldsymbol{w}^\top \boldsymbol{x}_i > 0$ と書ける．
以下の問いに答えよ．

(3) 上記の線形分類器がパターン $\boldsymbol{x}_i$ を正しくクラス y_i に分類できないとき，パラメータ $\boldsymbol{w}$ を

$$\boldsymbol{w}_{\text{new}} = \boldsymbol{w} + y_i \boldsymbol{x}_i$$

と更新することにする．この更新により，i 番目の訓練標本 $(\boldsymbol{x}_i, y_i)$ に対するマージンは減少しないことを示せ．

(4) 上記の線形分類器がパターン $\boldsymbol{x}_i$ を正しくクラス y_i に分類できないとき，パラメータ $\boldsymbol{w}$ を

$$\boldsymbol{w}_{\text{new}} = \underset{\boldsymbol{w}'}{\operatorname{argmin}} \left[\|\boldsymbol{w}' - \boldsymbol{w}\|^2 + (1 - y_i \boldsymbol{w}'^\top \boldsymbol{x}_i)^2 \right]$$

と更新することにする．この最適化問題を解き，$\boldsymbol{w}_{\text{new}}$ を陽に求めよ．

Problem 1

Consider the pattern recognition problem of classifying a d-dimensional real vectorial pattern $\boldsymbol{x} \in \mathbb{R}^d$ into one of the two classes $y = +1, -1$. For training a classifier, suppose that n training samples

$$\left\{ (\boldsymbol{x}_i, y_i) \;\middle|\; \boldsymbol{x}_i \in \mathbb{R}^d,\ y_i \in \{+1, -1\},\ i = 1, \ldots, n \right\}$$

are provided, where $(\boldsymbol{x}_i, y_i)$ means that the pattern $\boldsymbol{x}_i$ belongs to the class y_i.

Answer the following questions.

(1) Among the n training samples, let n_+ and n_- be the numbers of patterns in the classes $+1$ and -1, respectively. Find the mean vector $\boldsymbol{c}_+$ of the n_+ patterns in the class $+1$, and the mean vector $\boldsymbol{c}_-$ of the n_- patterns in the class -1.

(2) Consider the classifier that assigns a sample $\boldsymbol{x}$ to the class $+1$ if $\|\boldsymbol{x} - \boldsymbol{c}_+\| < \|\boldsymbol{x} - \boldsymbol{c}_-\|$, and to the class -1 if $\|\boldsymbol{x} - \boldsymbol{c}_+\| > \|\boldsymbol{x} - \boldsymbol{c}_-\|$. Here $\|\cdot\|$ denotes the Euclidean norm. Give an equation for the boundary between: the region to which the patterns classified into the class $+1$ belong; and the region to which the patterns classified into the class -1 belong.

For a parameter $\boldsymbol{w} \in \mathbb{R}^d$, consider the linear classifier that assigns a sample $\boldsymbol{x}$ to the class $+1$ if $\boldsymbol{w}^\top \boldsymbol{x} > 0$, and to the class -1 if $\boldsymbol{w}^\top \boldsymbol{x} < 0$. Here $(\cdot)^\top$ denotes the transpose. Let us call the value $y_i \boldsymbol{w}^\top \boldsymbol{x}_i$ the margin for the i-th training sample $(\boldsymbol{x}_i, y_i)$. Then the condition for this linear classifier to correctly classify the pattern $\boldsymbol{x}_i$ into the class y_i can be expressed, in terms of the margin, as $y_i \boldsymbol{w}^\top \boldsymbol{x}_i > 0$.

Answer the following questions.

(3) When the linear classifier shown above does not correctly classify the pattern $\boldsymbol{x}_i$ into the class y_i, let us update the parameter $\boldsymbol{w}$ by

$$\boldsymbol{w}_{\text{new}} = \boldsymbol{w} + y_i \boldsymbol{x}_i \ .$$

Prove that this parameter update does not decrease the margin for the i-th training sample $(\boldsymbol{x}_i, y_i)$.

(5) When the linear classifier shown above does not correctly classify the pattern $\boldsymbol{x}_i$ into the class y_i, let us update the parameter $\boldsymbol{w}$ by

$$\boldsymbol{w}_{\text{new}} = \operatorname*{argmin}_{\boldsymbol{w}'} \left[\|\boldsymbol{w}' - \boldsymbol{w}\|^2 + (1 - y_i \boldsymbol{w}'^\top \boldsymbol{x}_i)^2 \right] \ .$$

Solve this optimization problem and obtain $\boldsymbol{w}_{\text{new}}$ explicitly.

問題 2

$\mathbb{N}$ を非負整数全体の集合とする．状態の集合 Q を $Q = \mathbb{N} \times \mathbb{N} \times \mathbb{N}$ と定義し，Q 上の遷移関係 $\longrightarrow$ を以下のように定める．

$$
\begin{aligned}
(a,b,c) &\longrightarrow (a-1, b-1, c+2) && (\text{ただし } a > 0 \text{ かつ } b > 0 \text{ のとき}) \\
(a,b,c) &\longrightarrow (a+2, b-1, c-1) && (\text{ただし } b > 0 \text{ かつ } c > 0 \text{ のとき}) \qquad (\dagger) \\
(a,b,c) &\longrightarrow (a-1, b+2, c-1) && (\text{ただし } c > 0 \text{ かつ } a > 0 \text{ のとき})
\end{aligned}
$$

関係 $\longrightarrow$ の反射推移閉包を $\longrightarrow^*$ と記す．

以下の問いに答えよ．

(1) $(1,2,3) \longrightarrow^* q$ となる状態 $q \in Q$ をすべて挙げ，状態遷移グラフを描け．

(2) $(a,b,c) \longrightarrow q$ となる状態 q が存在しないとき，状態 (a,b,c) をデッドロック状態とよぶ．状態 (a,b,c) がデッドロック状態であるための必要十分条件を述べよ．

(3) 状態 (a,b,c) に対して，$(a,b,c) \longrightarrow^* q$ となるデッドロック状態 q が存在するための必要十分条件を述べよ．

(4) いま，各状態 (a,b,c) において，冒頭 $(\dagger)$ において定義した 3 つの遷移のうちの 1 つがつぎのような確率で選ばれて起こるものとする．

$$
\begin{aligned}
(a,b,c) &\longrightarrow (a-1, b-1, c+2) && \cdots && \text{確率 } ab/(ab+bc+ca) \\
(a,b,c) &\longrightarrow (a+2, b-1, c-1) && \cdots && \text{確率 } bc/(ab+bc+ca) \\
(a,b,c) &\longrightarrow (a-1, b+2, c-1) && \cdots && \text{確率 } ca/(ab+bc+ca)
\end{aligned}
$$

いま，初期状態を $(1,2,3)$ とし，上記のような確率的なやり方で十分多くの遷移を行ったとする．このような遷移を行ったのちに，現在の状態が $(1,2,3)$，$(3,1,2)$，$(2,3,1)$ のいずれかであるような確率を求めよ．

Problem 2

Let $\mathbb{N}$ be the set of all nonnegative integers. Let Q be a set of states defined by $Q = \mathbb{N} \times \mathbb{N} \times \mathbb{N}$, and let a transition relation $\longrightarrow$ on Q be defined as follows.

$$\begin{array}{lll} (a,b,c) \longrightarrow (a-1,b-1,c+2) & \text{(if } a>0 \text{ and } b>0) & \\ (a,b,c) \longrightarrow (a+2,b-1,c-1) & \text{(if } b>0 \text{ and } c>0) & (\dagger) \\ (a,b,c) \longrightarrow (a-1,b+2,c-1) & \text{(if } c>0 \text{ and } a>0) & \end{array}$$

Let $\longrightarrow^*$ denote the reflexive transitive closure of $\longrightarrow$.

Answer the following questions.

(1) Enumerate all states $q \in Q$ such that $(1,2,3) \longrightarrow^* q$, and draw a state transition graph.

(2) A state (a,b,c) is called a deadlock state if there exists no state q such that $(a,b,c) \longrightarrow q$. Give a necessary and sufficient condition for a state (a,b,c) to be a deadlock state.

(3) Give a necessary and sufficient condition for a state (a,b,c) to have a deadlock state q such that $(a,b,c) \longrightarrow^* q$.

(4) Assume that, at each state (a,b,c), one out of the three transitions defined in the above $(\dagger)$ is chosen to take place, with the following probabilities.

$$\begin{array}{ll} (a,b,c) \longrightarrow (a-1,b-1,c+2) & \text{with the probability } ab/(ab+bc+ca) \\ (a,b,c) \longrightarrow (a+2,b-1,c-1) & \text{with the probability } bc/(ab+bc+ca) \\ (a,b,c) \longrightarrow (a-1,b+2,c-1) & \text{with the probability } ca/(ab+bc+ca) \end{array}$$

Now let an initial state be $(1,2,3)$, and consider repeating the above probabilistic transitions for sufficiently many times. Compute the probability with which, after such transitions, the current state is either $(1,2,3)$, $(3,1,2)$ or $(2,3,1)$.

問題 3

Σ を文字の有限集合とする．Σ 上の文字列の集合 $S = \{w_1, w_2, \ldots, w_n\}$ を計算機上に表現して，所属性判定問題を解くことを考える．すなわち，文字列 w を入力として，$w \in S$ であれば「yes」，$w \notin S$ であれば「no」を出力したい．ただし文字の種類の数（すなわち Σ の大きさ）を m，文字列 $w_1, w_2, \ldots, w_n$ の平均の長さを ℓ とする．また，入力文字列 w の平均の長さも同じく ℓ とする．集合 S に属する文字列の数は n である．

各問に答えるにあたって他に数値が必要になる場合には，その値を適切な変数で表現したのちその変数を用いて答えること．また，図示にあたっては以下の文字列の集合 S_0 を例として用いよ：$S_0 = \{\mathtt{CAT}, \mathtt{CAP}, \mathtt{CAPE}, \mathtt{REASON}, \mathtt{RAINBOW}\}$.

以下の問いに答えよ．

(1) まず素朴な手法として，文字列の集合 S を基本的な文字列オブジェクト（文字の配列）の連結リストとして表現することを考える．この手法で S を計算機上に表現した場合の，必要なメモリ容量と，所属性判定問題の計算の手間（平均）を示せ．簡単な説明も付けること．

(2) 次に，文字列の集合 S をハッシュ（開番地法）を用いて表現することを考える．この場合の必要なメモリ容量と，所属性判定問題の計算の手間（平均）を示せ．簡単な説明も付けること．ハッシュ関数の定義などの詳細は適宜決めてよい．

(3) 文字列の集合 S を 2 分探索木を用いて表現することを考える．ただし，木の頂点それぞれに文字列を格納し，文字列の同士の大小関係の比較は辞書順を使うものとする．この場合の必要なメモリ容量と，所属性判定問題の計算の手間（平均）を示せ．簡単な説明も付けること．

また，上記の例 S_0 を表現した場合のデータ構造を図示せよ．ただし，木の構築にあたっては，S_0 の各要素を上記に示す順に挿入するものとする．

(4) 文字列の集合を表現する手法として，トライと呼ばれる木構造を用いる方法がある．これは，木の根からそれぞれの葉までの経路を一つの文字列に対応させるもので，各中間頂点の持つ配列には文字の種類の数（m）だけ子ノードへのポインタが格納されている．この方法で文字列の集合 S を計算機上に表現した場合の，必要なメモリ容量と，所属性判定問題の計算の手間（平均）を示せ．簡単な説明も付けること．また，上記の例 S_0 を表現した場合のデータ構造を図示せよ．

(5) トライを用いた場合の 1 つの欠点として，$\ell \gg n$ である場合にメモリを無駄に占有するという問題がある．この問題を解決する方法を 1 つ挙げて，その動作を例を用いて説明せよ．

(6) トライを用いた場合のもう 1 つの欠点として，m が大きい場合にメモリを無駄に占有するという問題がある．この問題を解決する方法を 1 つ挙げて，その動作を例を用いて説明せよ．

Problem 3

Let Σ be a finite set of characters. Let $S = \{w_1, w_2, \ldots, w_n\}$ be a set of strings over Σ; we consider representing S on a computer and solving its membership problem. That is, given an input string w, we would like to answer "yes" if $w \in S$, and "no" if $w \notin S$. Here let m be the number of characters (i.e. the size of Σ); and ℓ be the average length of the strings $w_1, w_2, \ldots, w_n$. We assume that the average length of an input string w is ℓ, too. Recall that n is the number of strings in the set S.

In case you need other parameters answering the questions below, introduce suitable variables for those parameters and use them in your answers. In illustration of your answers, use the following set S_0 as an example: $S_0 = \{\mathtt{CAT}, \mathtt{CAP}, \mathtt{CAPE}, \mathtt{REASON}, \mathtt{RAINBOW}\}$.

Answer the following questions.

(1) A naive approach is to represent the set S as a linked list of basic string objects (i.e. arrays of characters). Answer the amount of memory needed, and the average complexity of the membership problem, in this setting. Give a brief explanation of your answer.

(2) Let us now consider representing the set S using hashing (with open addressing). Answer the amount of memory needed, and the average complexity of the membership problem. Give a brief explanation of your answer; you can choose and fix further details, like the definition of a hash function.

(3) Let us consider representing the set S using a binary search tree. Here each node of the tree stores a string; and strings are compared with respect to the lexicographic order. Answer the amount of memory needed, and the average complexity of the membership problem. Give a brief explanation of your answer.

Illustrate the data structure that represents the above example S_0. Assume here that the tree is constructed by inserting each element of S_0 in the order shown above.

(4) A trie is a tree structure that is often used to represent a set of strings. In a trie, one path from the root to a leaf corresponds to one string; and each internal node has an array, of size m (the number of characters), that stores pointers to its children nodes.

Answer the amount of memory needed, and the average complexity of the membership problem, in this setting. Give a brief explanation of your answer. Illustrate the data structure that represents the above example S_0.

(5) One potential disadvantage of using a trie is that, in case $\ell \gg n$, memory usage can be excessive. Describe a countermeasure, and explain how it works with an example.

(6) Another potential disadvantage of using a trie is that, in case m is large, memory usage can be excessive. Describe a countermeasure, and explain how it works with an example.

問題 4

キャッシュについて以下のような状況を考える．キャッシュは L バイトに固定したブロックサイズを持ち，ウェイ数 W のセットアソシアティブ方式またはダイレクトマップ方式 ($W = 1$ にあたる) であるとする．置換アルゴリズムは LRU であり，またライトバック方式を仮定する．

この問題では $N \times N$ の 2 つの行列の積の計算を考える．行列は C 言語により

```
float A[N][N], B[N][N], C[N][N];
```

と宣言されている．ここで，float は 4 バイトであり，行列 A, B, C は連続したメモリ領域に割り当てられ，行優先形式（すなわち A[0][0] の次に A[0][1] が来る）である．A の最初の要素のアドレスはブロックサイズ L に対して整列していると仮定する．また，計算内容（以下,「行列積」と呼ぶ）は以下の通りである．

```
for (i=0; i< N; i++)
  for (k=0; k< N; k++) {
    float a_ik = A[i][k];
    for (j=0; j< N; j++) {
      float b_kj = B[k][j];
      C[i][j] += a_ik * b_kj;
    }
  }
```

キャッシュヒット率の計算においては，配列 A, B, C に対するデータアクセスのみを考え，その他のデータアクセスや命令フェッチについては考えない．

以下の問いに答えよ．

(1) 一般にキャッシュを用いるとプログラム実行が高速化する理由を簡潔に述べよ．

(2) キャッシュ容量が 8192 バイトで $L = 64$, $W = 4$ のとき，$N = 512$ の行列積でのキャッシュヒット率（概数でよい）を求めよ．

(3) 次に，キャッシュ容量が 2048 バイトで $L = 64$, $W = 1$ のとき，$N = 512$ の行列積でのキャッシュヒット率（概数でよい）を求めよ．

(4) 問い (3) の条件で，この行列積を高速実行するプログラミング手法を一つ説明し，キャッシュヒット率を計算するなどして高速化の効果を定量的に説明せよ．(必要ならば，メモリとキャッシュの遅延や帯域などについて，適当な仮定をおいてよい．)

Problem 4

Consider the following situation on cache. The cache has fixed-length block size of L bytes, and is W-way set-associative or direct-mapped (i.e. $W = 1$). Assume write-back cache with the LRU replacement policy.

We consider multiplication of two $N \times N$ matrices in this problem. The matrices are declared in C language as follows.

```
float A[N][N], B[N][N], C[N][N];
```

Here the size of `float` is 4 bytes, the matrices `A`, `B`, and `C` are allocated to contiguous areas of memory, and the format is row-major order (that is, the element that follows `A[0][0]` is `A[0][1]`). Assume that the address of the first element of `A` is aligned to the block size L. The computation (called "matrix multiplication" hereafter) is the following.

```
for (i=0; i< N; i++)
  for (k=0; k< N; k++) {
    float a_ik = A[i][k];
    for (j=0; j< N; j++) {
      float b_kj = B[k][j];
      C[i][j] += a_ik * b_kj;
    }
  }
```

In calculating cache hit ratios, only the data access to the arrays `A`, `B` and `C` should be taken into account. Disregard access to the other data and instruction fetch.

Answer the following questions.

(1) Describe briefly why cache can accelerate computation in general.

(2) Assume that the cache capacity is 8192 bytes, $L = 64$ and $W = 4$. Calculate the cache hit ratio (approximately) in matrix multiplication with $N = 512$.

(3) Next, assume that the cache capacity is 2048 bytes, $L = 64$ and $W = 1$. Calculate the cache hit ratio (approximately) in matrix multiplication with $N = 512$.

(4) Assume the same conditions as Question (3). Describe one programming technique that speeds up the matrix multiplication, and explain the speed-up effect quantitatively, for example by calculating the cache hit ratio. (You can introduce your own assumptions, if you want, for example on the latencies and the bandwidths of the main memory and the cache.)

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.