平成 27 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 26 年 8 月 19 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 19, 2014

Entrance Examination (AY 2015)
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

DNA の変異蓄積過程を考える．実数 $t \geq 0$ と整数 $n \geq 0$ に対して，$P(n,t)$ を時刻 $t$ において変異箇所数が $n$ である確率とする．関数 $P(n,t)$ は以下の微分方程式に従うものとする．

$$\begin{aligned} \frac{dP(0,t)}{dt} &= -\omega P(0,t) \\ \frac{dP(n,t)}{dt} &= \omega\big(P(n-1,t) - P(n,t)\big) \quad (n \geq 1) \end{aligned} \tag{$*$}$$

ただしここで $\omega > 0$ は定数であり，また $P(0,0) = 1$ とする．

以下の問いに答えよ．

(1) $P(0,t)$ を $t$ を用いて表せ．

(2) 関数

$$G(s,t) = \sum_{n=0}^{\infty} s^n P(n,t)$$

を考える．$\dfrac{dG(s,t)}{dt}$ を $G(s,t)$ を用いて表せ．ここで $|s| \leq 1$ とし，また式 $(*)$ を用いてよい．

(3) 問い (2) で導いた微分方程式を解き，$G(s,t)$ を $\omega$ を用いて表せ．

(4) $G(s,t)$ の $s$ に関するテイラー展開を考えることにより，$P(n,t)$ を求めよ．

(5) $G(s,t)$ の $s$ に関する微分を考えることにより，変異箇所数 $n$ の時刻 $t$ における期待値および分散を求めよ．

## Problem 1

Let us consider accumulation of mutations in DNA. For a real number $t \geq 0$ and an integer $n \geq 0$, let $P(n,t)$ denote the probability with which the number of mutations is $n$ at time $t$. The function $P(n,t)$ is subject to the following differential equations.

$$\begin{aligned} \frac{dP(0,t)}{dt} &= -\omega P(0,t) \\ \frac{dP(n,t)}{dt} &= \omega\big(P(n-1,t) - P(n,t)\big) \quad (n \geq 1) \end{aligned} \tag{$*$}$$

Here $\omega > 0$ is a constant, and we let $P(0,0) = 1$.

Answer the following questions.

(1) Express $P(0,t)$ using $t$.

(2) Consider a function

$$G(s,t) = \sum_{n=0}^{\infty} s^n P(n,t) \ .$$

Express $\frac{dG(s,t)}{dt}$ using $G(s,t)$. Here assume $|s| \leq 1$; you can use the equations $(*)$.

(3) Solve the differential equation answered in Question (2), and express $G(s,t)$ using $\omega$.

(4) Find the value of $P(n,t)$, by taking a Taylor expansion of $G(s,t)$ with respect to $s$.

(5) Find the expected value and the variance of the number $n$ of mutations, at time $t$, by taking derivatives of $G(s,t)$ with respect to $s$.

## 問題 2

アルファベット $\Sigma$ 上の言語 $S, T \subseteq \Sigma^*$ に対して，次のように演算 $\cdot, +, (\_)^*$ を定める．

$$
\begin{aligned}
S \cdot T &:= \{st \mid s \in S, t \in T\} && \text{連結} \\
S + T &:= S \cup T && \text{和集合} \\
S^* &:= \{s_1 s_2 \dots s_n \mid n \in \mathbb{N}, s_i \in S\} && \text{反復}
\end{aligned}
$$

ここで $\mathbb{N} = \{0, 1, 2, \dots\}$ は自然数の集合である．演算の結合の強さは $(\_)^* > \cdot > +$ であるとし，括弧を適宜省略する．

また，いくつかの特別な言語を次のように書きあらわす．

$$
\begin{aligned}
1 &:= \{\varepsilon\} && \text{長さ 0 の語（空語）} \varepsilon \text{ のみを含む言語} \\
0 &:= \emptyset && \text{空言語} \\
a &:= \{a\} && \text{長さ 1 の語 } a \text{ のみを含む言語．ただし } a \in \Sigma
\end{aligned}
$$

以下の問いに答えよ．

(1) 所与の言語 $S, T \subseteq \Sigma^*$ に対して，次のような $L$ についての再帰方程式を考える．

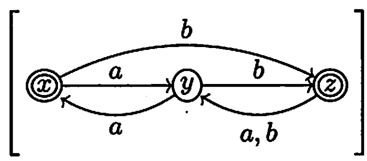
$$L = S \cdot L + T \qquad (\dagger)$$

$L = S^* \cdot T$ がこの再帰方程式の解であることを示せ．

(2) 再帰方程式 $(\dagger)$ の解のうち，解 $L = S^* \cdot T$ は包含関係について最小である．このことを示せ．

証明にあたっては，事実 $S^* = \bigcup_{n \in \mathbb{N}} S^n = \bigcup_{n \in \mathbb{N}} \overbrace{S \cdot S \cdot \cdots \cdot S}^{n}$ を用いてよい．

(3) 次の決定性有限オートマトン $\mathcal{A}$ を考える．

$\mathcal{A} =$ 


状態 $x, y, z$ を初期状態とした場合の受理言語をそれぞれ $L_x, L_y, L_z$ と書く．$L_x$ を $L_y, L_z, \cdot, +, a, b, 1$ を用いて表せ．

(4) 上記のオートマトン $\mathcal{A}$ において，$y$ を初期状態とした場合の受理言語を正規表現で表せ．

(5) $S$ が空語 $\varepsilon$ を含まないとき，再帰方程式 $(\dagger)$ に対して $L = S^* \cdot T$ は唯一の解である．下記はこの事実の証明である．空欄 (a)–(d) をうめよ．

$S = \emptyset$ のときは明らかに成立．

以下，$S \neq \emptyset$ とする．$L = S^* \cdot T + U$（ただし $(S^* \cdot T) \cap U = \emptyset$ とする）が解であると仮定する．すると $S^* \cdot T + U = S^* \cdot T + S \cdot U$ がなりたつ (なぜならば $\boxed{\text{(a)}}$)．この両辺に対して $\boxed{\text{(b)}}$ をとると，$U = (S \cdot U) \cap U$．ゆえに包含関係 $U$ $\boxed{\text{(c)}}$ $S \cdot U$ が成立する．これは $U = \emptyset$ の場合のみ可能である．なぜならば $\boxed{\text{(d)}}$．

## Problem 2

Let $\Sigma$ be an alphabet and $S, T \subseteq \Sigma^*$ be languages over $\Sigma$. We define operations $\cdot, +, (\_)^*$ over those languages as follows.

$$
\begin{array}{rcll}
S \cdot T & := & \{st \mid s \in S, t \in T\} & \text{concatenation} \\
S + T & := & S \cup T & \text{union} \\
S^* & := & \{s_1 s_2 \ldots s_n \mid n \in \mathbb{N}, s_i \in S\} & \text{repetition}
\end{array}
$$

Here $\mathbb{N} = \{0, 1, 2, \ldots\}$ is the set of natural numbers. We let the binding strength of the operations be $(\_)^* > \cdot > +$, and omit parentheses when confusion is unlikely.

We also introduce notations for some special languages.

$$
\begin{array}{rcll}
1 & := & \{\varepsilon\} & \text{only includes the emptyword } \varepsilon \text{ (whose length is 0)} \\
0 & := & \emptyset & \text{the empty language} \\
a & := & \{a\} & \text{only includes the word } a \text{ of length 1, where } a \in \Sigma
\end{array}
$$

Answer the following questions.

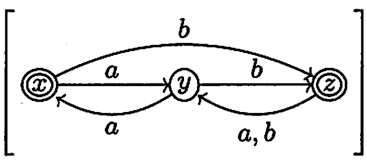
(1) For given languages $S, T \subseteq \Sigma^*$, consider the following recursive equation over $L$.

$$L = S \cdot L + T \qquad (\dagger)$$

Show that $L = S^* \cdot T$ is a solution of the recursive equation.

(2) Among the solutions of the recursive equation ($\dagger$), $L = S^* \cdot T$ is the smallest with respect to inclusion. Prove this. In the proof you can use the fact that $S^* = \bigcup_{n \in \mathbb{N}} S^n = \bigcup_{n \in \mathbb{N}} \overbrace{S \cdot S \cdot \cdots \cdot S}^{n}$.

(3) Let $\mathcal{A}$ be the following deterministic finite automaton.

$\mathcal{A} =$ 


Let $L_x, L_y$, and $L_z$ stand for the accepted languages of $\mathcal{A}$ when the initial state is $x, y$ and $z$, respectively. Represent $L_x$ using $L_y, L_z, \cdot, +, a, b$ and $1$.

(4) Let $y$ be the initial state in the above automaton $\mathcal{A}$. Give a regular expression that represents its accepted language.

(5) If $S$ does not contain the emptyword $\varepsilon$, $L = S^* \cdot T$ is the unique solution of the recursive equation ($\dagger$). This fact is proved as follows; fill in the blanks (a)–(d).

> The claim is obvious when $S = \emptyset$.
>
> Now assume $S \neq \emptyset$, and let $L = S^* \cdot T + U$ be a solution where $(S^* \cdot T) \cap U = \emptyset$. Then we have $S^* \cdot T + U = S^* \cdot T + S \cdot U$; this is because $\boxed{\text{(a)}}$. Taking $\boxed{\text{(b)}}$ of the both sides, we obtain $U = (S \cdot U) \cap U$. Therefore we have inclusion $U \boxed{\text{(c)}} S \cdot U$. This is only possible if $U = \emptyset$, because $\boxed{\text{(d)}}$.

## 問題 3

辺にコストが設定されている連結な無向有限グラフ $G = (V, E)$ が与えられたときに，すべての頂点を連結し，かつ閉路を含まないような部分グラフでコスト最小になるものを以下のようなアルゴリズムによって求める問題を考える．

- 1 つの頂点 $s \in V$ を任意に選んで，集合 $U$ を $U = \{s\}$ とする．
- 以下の性質 $(*)$ をみたすテーブルを用意する．

  $(*)$ テーブルは $|V \setminus U|$ 個のエントリからなる．各エントリは，頂点 $v \in V \setminus U$，集合 $U$ 中の頂点の中で頂点 $v$ に最も近いもの $u$（$u$ を $v$ の最寄頂点とよぶ），および $v, u$ 間の距離 $d(v, u)$ の 3 つ組 $\bigl(v,\ u,\ d(v, u)\bigr)$ からなる．

  ここで，頂点の間の距離とは，その頂点間を結ぶ辺のコストのことを指すものとする．直接辺で結ばれていない場合の距離は $\infty$ とする．

  初期状態においては，各頂点 $v \in V \setminus U$ に対して，エントリ $\bigl(v,\ s,\ d(v, s)\bigr)$ をテーブルに格納する．

- 空の辺の集合 $T$ を用意し，以下の手順を $V \setminus U$ が空になるまで繰り返し，最後に得られた $T$ が求める部分グラフを構成する．
  - 集合 $V \setminus U$ の頂点のうち，テーブルに格納されている距離のもっとも小さい頂点 $v$ を選び，$v$ とその最寄頂点 $u \in U$ を結ぶ辺を $T$ に加える．またこのとき，$\boxed{\text{(a)}}$ を $\boxed{\text{(b)}}$ に加え，さらにそれに合わせてテーブルの内容を性質 $(*)$ をみたすように更新する．

(1) アルゴリズムが正しく動作するように，空欄 $\boxed{\text{(a)}}$ および $\boxed{\text{(b)}}$ をうめよ．

(2) 上記のアルゴリズムによって，グラフの辺のコストがすべて非負であった場合に，求めたい部分グラフが得られることを説明せよ．

(3) 上記のアルゴリズムは，辺のコストに負のものがあっても正しく動作するか，答えよ．正しく動作しない場合には反例を示せ．正しく動作する場合には，根拠を示せ．

(4) グラフが密な場合 $(|E| \approx |V|^2)$ には，どのようなデータ構造を用いるべきか，説明せよ．またその場合の時間計算量とその根拠を示せ．

(5) グラフが疎な場合 $(|E| = O(|V|))$ には，どのようなデータ構造を用いるべきか，説明せよ．またその場合の時間計算量とその根拠を示せ．

## Problem 3

Let $G = (V, E)$ be an undirected, connected, and finite graph with a cost associated with each edge. Among those subgraphs of $G$ which contain all vertices of $G$, are connected, and do not contain any loop, we shall find one with the minimum total edge cost by the following algorithm.

- Pick an arbitrary vertex $s \in V$ and let a set $U$ be $U = \{s\}$.
- Prepare a table that satisfies the following property $(*)$.

  $(*)$ The table consists of $|V \setminus U|$ entries. Each entry is a triple $\big(v,\ u,\ d(v,u)\big)$ of: a vertex $v \in V \setminus U$; the vertex $u$ in $U$ that is closest to $v$; and the distance $d(v,u)$ between $v$ and $u$.

  Here, the distance between two vertices is defined as the cost of the edge connecting the two vertices. If there is no edge that directly connects the two vertices, the distance is defined to be $\infty$.

  Initially, the table stores the entry $\big(v,\ s,\ d(v,s)\big)$ for each $v \in V \setminus U$.
- Introduce a set $T$ of edges that is initially empty, and repeat the following procedure until $V \setminus U$ is empty. The resulting $T$ will represent a desired subgraph.
  - Among the vertices in the set $V \setminus U$, pick a vertex $v$ with the minimum distance stored in the table. Add the edge connecting $v$ and its closest vertex $u$, as stored in the table, to $T$. At the same time, add $\boxed{\text{(a)}}$ to $\boxed{\text{(b)}}$. Finally, update the table so that it satisfies the property $(*)$.

(1) Fill in the blanks $\boxed{\text{(a)}}$ and $\boxed{\text{(b)}}$ so that the algorithm works correctly.

(2) Show that the above algorithm gives a desired subgraph when all edge costs are non-negative.

(3) Does the above algorithm work correctly even when the graph contains an edge with a negative cost? If no, give a counter example. If yes, explain the reason.

(4) Explain what kind of data structure to use when the graph is dense ($|E| \approx |V|^2$). Give the time complexity when using that data structure, and explain its reason.

(5) Explain what kind of data structure to use when the graph is sparse ($|E| = O(|V|)$). Give the time complexity when using that data structure, and explain its reason.

## 問題 4

3 進数の演算を行う論理回路を考える．ここでは，3 進数の各桁は 0, 1, 2 の値のいずれかであり，3 進数の 1 桁を 2 ビットで表現するものとする．

以下の問いに答えよ．

(1) 10 進数の 20 および 65 を 3 進数で表せ．

(2) 3 進数 1 桁の半加算器および全加算器を AND，OR，NOT ゲートを用いて設計せよ．まず真理値表を示し，次に論理回路を示せ．

(3) 3 進数 4 桁の加算器を，問い (2) で設計した半加算器および全加算器を用いて設計せよ．

(4) 3 進数 16 桁のキャリールックアヘッド加算器を設計せよ．

## Problem 4

Consider logic circuits that realize arithmetic operations on ternary numbers. Here one digit of a ternary number is either 0, 1 or 2, and a digit of a ternary number is represented by two bits.

Answer the following questions.

(1) Represent decimal numbers 20 and 65 by ternary numbers.

(2) Design a half adder and a full adder of 1-digit ternary numbers using AND, OR and NOT gates. Firstly show their truth tables; and then design logic circuits.

(3) Design an adder of 4-digit ternary numbers using the half adders and the full adders designed in Question (2).

(4) Design a carry-lookahead adder of 16-digit ternary numbers.

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.