大学院情報理工学研究科
博士前期課程一般入試　入学試験問題
（2022年8月17日実施）

【情報学専攻】

専門科目：［選択問題］

※注意事項

1．試験開始の合図があるまで問題冊子を開いてはいけない。

2．選択問題の問題冊子はこの注意事項を含めて１４枚、解答用紙は８枚である。（マークシート１枚を含む）

3．試験開始の合図の後、全ての解答用紙に受験番号を記入すること。
マークシートに受験番号をマークする際には左詰めで記入し、氏名は記入しないこと。

4．試験時間は必須問題と選択問題をあわせて180分である。

5．選択問題では、４科目の中から３科目を選んで解答すること。
また、選択した３科目は、選択科目記入シートに必ず○印を記入すること。
（採点は選択科目記入シートに○印が記入された科目についてのみ行う。誤記入、記入もれに十分注意すること。）
「確率・オペレーションズリサーチ」では、問２か問３を、
「計算機工学」は4-1［形式言語理論］か、4-2［計算機アーキテクチャ］を選択すること。

6．解答は、必ず当該科目の解答用紙を使用すること。
（解答用紙には問題番号が記入されているので、解答する科目番号が記入されている解答用紙を使用すること。「離散数学」問１～問３はマークシートを使用すること。）
また、解答用紙は裏面を使用してもよいが、その場合は表面下に「裏面へ続く」と記入すること。

7．選択科目記入シートは、試験終了後に必ず提出すること。

8．試験中に問題冊子の印刷不鮮明、ページの落丁・乱丁及び解答用紙の汚れ等に気付いた場合は、手を挙げて監督者に知らせること。

9．試験終了後、この問題冊子は持ち帰ること。

10．解答は英語でもよい。

問題は次のページからです。

このページは問題冊子の枚数には
含みません。

大学院情報理工学研究科　博士前期課程：一般入試（2022 年 8 月 17 日実施）

選択問題

情報学専攻

科目の番号

1　アルゴリズムとデータ構造

以下の二分探索木に関連する問に下記の C 言語のコードを参照して答えよ.

```
// 木構造は以下で定義する
typedef struct _Node_ {
  int d; // データ要素
  struct _Node_ *l; // 左部分木
  struct _Node_ *r; // 右部分木
} Node;

Node* NewNode(int x)
{
    Node *p = malloc(sizeof(Node));
    p->d = x;
    p->l = p->r = NULL;
    return p;
}
```

```
Node *insertA(int x, Node *p) // 二分探索木へ挿入
{
  if(p == NULL){
    Node *q = NewNode(x);
    return q;
  }
  if(x < p->d)       p->l = insertA(x, p->l);
  else if(x > p->d)  p->r = insertA(x, p->r);
  return p;
}
```

コード中の // 以下の表記は，行末までコメントを表す．ノードを指すポインタが NULL であった場合はノードが無いことを表す．また関数 malloc() はノードの領域確保を行い，そのポインタを返す．

N 個の要素を持つ配列 C[]全体を二分探索木に格納する場合は， p = NULL の状態を初期状態とし，

```
for(i = 0; i < N; i++){  p = insertA(C[i], p);  }
```

という操作を行う．最終的に得られた p を根へのポインタとして扱う．

1. 配列 C1[] = {4, 6, 2, 3, 5, 1, 7} と C2[] = {1, 7, 2, 6, 3, 5, 4} とを格納した場合の，木を図示し，木の高さをそれぞれ答えよ．ただし，木の高さは 0 から数えるものとする．すなわち根ノードしかない木の場合，その高さは 0 となる．

2. 1. で構築した，それぞれの木に対して，前順（行きがけ），中順（通りがけ）になぞりをかけノードの要素を順次出力していく．それぞれの木から得られる出力を答えよ．

3. 関数 insertA()を用いて構築される二分探索木はデータの格納順に依存する．データ {1, 2, 3, 4} の 4 つの要素に関して，あらゆる格納順を考え，関数 insertA() を用いて二分探索木を構築したとき，高さ 2 の木となる順列の個数と高さ 3 の木となる順列の個数をそれぞれ答えよ．

4. 二分探索木の高さは検索性能に強い影響を与える．いまデータが N 個あった場合，関数 insertA() によって構築される二分探索木のうち，検索性能の意味で最良の木の高さと最悪の木の高さを$O(\cdot)$（ビッグ・オー）表記を用いて表せ．

【次ページに続く】

【前ページから続く】
さらに以下のようなコードを考える.

```
Node *rot1(Node *p)
{
    Node *tmp = p->l;
    p->l = tmp->r;
    tmp->r = p;
    return tmp;
}

Node *rot2(Node *p)
{
    p->l = rot3(p->l);
    return rot1(p);
}
```

```
Node *rot3(Node *p)
{
    Node *tmp = p->r;
    p->r = tmp->l;
    tmp->l= p;
    return tmp;
}

Node *rot4(Node *p)
{
    p->r = rot1(p->r);
    return rot3(p);
}
```

```
Node *insertB(int x, Node *p) // 平衡木への挿入
{
  int b;
  if(p == NULL){
    Node *q = NewNode(x);
    return q;
  }
  if(x < p->d)        p->l = insertB(x, p->l);
  else if(x > p->d)   p->r = insertB(x, p->r);
  else return p;
  b = diff(p); //右部分木と左部分木の高さの差を計算する
  if(b > 1){ // 右側の子の方が高い (+1 より大きい)
    if(diff(p->r) >= 0)   p = [  (A)  ];
    else                  p = [  (B)  ];
  }
  else if(b < -1){
    if(diff(p->l) <= 0)   p = [  (C)  ];
    else                  p = [  (D)  ];
  }
  return p;
}
```

ただし, 関数 diff(p) は p で示されるノードの右左の部分木の高さの差を整数値で与える関数である. すなわち diff(p) = (p の右部分木(p->r)の高さ)-(p の左部分木(p->l)の高さ) を与える.

5. 配列 D1[] = {4, 3, 2, 1} を要素順に関数 insertA() を用いて構築した木構造の根へのポインタを r1 とする. さらに p1 = rot1(r1) として得られる木構造(p1 を根とする木構造)を図示せよ. 同様に D2[] = {4, 1, 3, 2} に関して得られる木構造の根へのポインタを r2 としたとき, p2 = rot2(r2) として得られる木構造を図示せよ.

6. 任意の部分木構造の高さの違いが一定の範囲にあるものを<u>平衡木</u>と呼ぶ. ここでは任意のノードにおいて左右の子の部分木の高さの差が 1 以内の平衡木を構築する関数 insertB()を考える. 関数 insertB()中では変数 b の値で右左の子の部分木の高さの差を取り調整する. この関数 insertB()中の空欄(A)～(D)に当てはまる語句を記述せよ.

二分探索木(Binary Search Tree), 部分木(Sub-tree), ノード(Node), 配列(Array), 根(Root), 木の高さ(Tree height), 前順(Pre-order), 中順(In-order), なぞり(Traverse), 平衡木(Balanced Tree)

大学院情報理工学研究科　博士前期課程：一般入試（2022 年 8 月 17 日実施）

選択問題

情報学専攻

科目の番号

2　確率・オペレーションズリサーチ

この科目（確率・オペレーションズリサーチ）を受験する場合には，問 1 は必ず解答し，問 2 と問 3 はいずれか一方のみを選択して解答すること．

問 1　確率変数Xの従う確率分布が次の確率密度を持つとする．ただし$I(\cdot)$は引数が真であれば 1，偽であれば 0 を返す識別関数とする．

$$f_X(x) = \left(-\frac{1}{2}x + 1\right) \cdot I(0 \le x < 2)$$

問 1-1：確率変数Xの平均を求めよ．
問 1-2：確率変数Xの分散を求めよ．

次に，確率$1/2$で1を，同じく$1/2$で-1をそれぞれ値に取り，Xと互いに独立な確率変数Yを考える．

問 1-3：$Z = XY$とする．Zの確率密度$f_Z(z)$を求めよ．
問 1-4：同じくZの分散を求めよ．

Keywords *確率変数: random variable, 確率分布: probability distribution, 確率密度: probability density, 真: true, 偽: false, 識別関数: indicator function, 平均: mean, 分散: variance, 確率: probability, 互いに独立: independent with each other.*

問 2　この問題を選択する場合には以下のＡ，Ｂの双方に解答すること．そして問３に解答してはいけない．

A　互いに独立に平均がλのポアソン分布に従う確率変数$X_1, X_2, \ldots$の累積和 $S_n = \sum_{i=1}^{n} X_i$ を考える．

問 2-1：S_nの確率関数$p_n(s)$を答えよ．
問 2-2：$\lambda = 5,\ n = 20$として，S_{20}が120を超える確率を求めよ．必要に応じて標準正規分布表を用いても良い．
問 2-3：$S_1, S_2, \ldots$を順に観測していくとする．S_nが初めてある正の整数Kを超えたときのnを，確率変数Nとする．Nの確率関数を，$p_n(s)$と$p_{n-1}(s)$を用いて表せ．

B　あるゲーム機は，メダルを投入してボタンを押すと，正の実数Xを表示する．実数Xは確率密度関数fを持つ確率分布Fに従う確率変数とみなすことができ，その平均はμとする．この数字がδをある正数として，$\mu + \delta$を超えると，メダルが$a(>1)$枚戻ってくる．$\mu + \delta$以下ならば，メダルは戻ってこない．また，複数回のゲームは互いに独立とする．
　このゲームについて，以下の問いに答えよ．

問 2-4：Xが$\mu + \delta$以上となる確率の上限を導け．

（次ページへ続く）

（前ページから続く）

問 2-5： $\mu = 2$，$\delta = 2$とする．このとき，10 回の連続したゲームの中で戻ってくるメダルの枚数の<u>期待値</u>は，いくつ以下になるか．問 2-4 の<u>結果</u>を用いて<u>評価</u>せよ．
問 2-6： Xの平均μであることに加えて，Xの分散がσ^2とわかっているとき，Xが$\mu + \delta$以上となる確率の上限を導け．
問 2-7： $\mu = 2$，$\delta = 2$に加えて，$\sigma = 1$とする．このとき，10 回の連続したゲームの中で戻ってくるメダルの枚数の期待値は，いくつ以下になるか．問 2-6 の結果を用いて評価せよ．

<u>Keywords</u> *ポアソン分布: Poisson distribution, 累積和: cumulative sum, 確率関数: probability function, 超え: over, 標準正規分布表: normal distribution table, 初めて: at the first time, 正の整数: positive integer, ゲーム機: game machine, メダル: medal, 投入: input, 正の実数: positive real number, 表示: show, 戻って: return, 以下: less than or equal to, こない: not, 回: times, 以上: more than or equal to, 上限: supremum, 期待値: expected value, 結果: result, 評価: evaluate*

標準正規分布表

	0.00	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09
0.0	0.50000	0.50399	0.50798	0.51197	0.51595	0.51994	0.52392	0.52790	0.53188	0.53586
0.1	0.53983	0.54380	0.54776	0.55172	0.55567	0.55962	0.56356	0.56749	0.57142	0.57535
0.2	0.57926	0.58317	0.58706	0.59095	0.59483	0.59871	0.60257	0.60642	0.61026	0.61409
0.3	0.61791	0.62172	0.62552	0.62930	0.63307	0.63683	0.64058	0.64431	0.64803	0.65173
0.4	0.65542	0.65910	0.66276	0.66640	0.67003	0.67364	0.67724	0.68082	0.68439	0.68793
0.5	0.69146	0.69497	0.69847	0.70194	0.70540	0.70884	0.71226	0.71566	0.71904	0.72240
0.6	0.72575	0.72907	0.73237	0.73565	0.73891	0.74215	0.74537	0.74857	0.75175	0.75490
0.7	0.75804	0.76115	0.76424	0.76730	0.77035	0.77337	0.77637	0.77935	0.78230	0.78524
0.8	0.78814	0.79103	0.79389	0.79673	0.79955	0.80234	0.80511	0.80785	0.81057	0.81327
0.9	0.81594	0.81859	0.82121	0.82381	0.82639	0.82894	0.83147	0.83398	0.83646	0.83891
1.0	0.84134	0.84375	0.84614	0.84849	0.85083	0.85314	0.85543	0.85769	0.85993	0.86214
1.1	0.86433	0.86650	0.86864	0.87076	0.87286	0.87493	0.87698	0.87900	0.88100	0.88298
1.2	0.88493	0.88686	0.88877	0.89065	0.89251	0.89435	0.89617	0.89796	0.89973	0.90147
1.3	0.90320	0.90490	0.90658	0.90824	0.90988	0.91149	0.91309	0.91466	0.91621	0.91774
1.4	0.91924	0.92073	0.92220	0.92364	0.92507	0.92647	0.92785	0.92922	0.93056	0.93189
1.5	0.93319	0.93448	0.93574	0.93699	0.93822	0.93943	0.94062	0.94179	0.94295	0.94408
1.6	0.94520	0.94630	0.94738	0.94845	0.94950	0.95053	0.95154	0.95254	0.95352	0.95449
1.7	0.95543	0.95637	0.95728	0.95818	0.95907	0.95994	0.96080	0.96164	0.96246	0.96327
1.8	0.96407	0.96485	0.96562	0.96638	0.96712	0.96784	0.96856	0.96926	0.96995	0.97062
1.9	0.97128	0.97193	0.97257	0.97320	0.97381	0.97441	0.97500	0.97558	0.97615	0.97670
2.0	0.97725	0.97778	0.97831	0.97882	0.97932	0.97982	0.98030	0.98077	0.98124	0.98169
2.1	0.98214	0.98257	0.98300	0.98341	0.98382	0.98422	0.98461	0.98500	0.98537	0.98574
2.2	0.98610	0.98645	0.98679	0.98713	0.98745	0.98778	0.98809	0.98840	0.98870	0.98899
2.3	0.98928	0.98956	0.98983	0.99010	0.99036	0.99061	0.99086	0.99111	0.99134	0.99158
2.4	0.99180	0.99202	0.99224	0.99245	0.99266	0.99286	0.99305	0.99324	0.99343	0.99361
2.5	0.99379	0.99396	0.99413	0.99430	0.99446	0.99461	0.99477	0.99492	0.99506	0.99520
2.6	0.99534	0.99547	0.99560	0.99573	0.99585	0.99598	0.99609	0.99621	0.99632	0.99643
2.7	0.99653	0.99664	0.99674	0.99683	0.99693	0.99702	0.99711	0.99720	0.99728	0.99736
2.8	0.99744	0.99752	0.99760	0.99767	0.99774	0.99781	0.99788	0.99795	0.99801	0.99807
2.9	0.99813	0.99819	0.99825	0.99831	0.99836	0.99841	0.99846	0.99851	0.99856	0.99861
3.0	0.99865	0.99869	0.99874	0.99878	0.99882	0.99886	0.99889	0.99893	0.99896	0.99900

（次ページへ続く）

（前ページから続く）

$\boxed{\text{問3}}$ この問題を選択する場合には，問 2 に解答してはいけない．

以下の式 (1)-(5) からなる最大化問題をシンプレックス法で解くことを考える:

$$
\begin{aligned}
& max\ 5x_1 + 9x_2 + 7x_3 && (1)\\
& s.t.\ \ x_1 + 3x_2 + 2x_3 \le 10, && (2)\\
& \quad 3x_1 + 4x_2 + 2x_3 \le 12, && (3)\\
& \quad 2x_1 + \ \ x_2 + 2x_3 \le 8, && (4)\\
& \quad\ x_1, x_2, x_3 \ge 0. && (5)
\end{aligned}
$$

このためには，以下の手続きを実行しなければならない．

ステップ1. 不等式制約をスラック変数 x_4, x_5, x_6 を用いて等式制約に変換する．

ステップ2. 最初の非基底変数を x_1, x_2, x_3，基底変数を x_4, x_5, x_6 として，実行基底解を求める．ここで，基底解とは 6 つの変数と変換した 3 つの等式制約からなる連立 1 次方程式の解である．また，実行基底解とはその変数の値が全て非負となる基底解のことである．

ステップ3. 求めた実行基底解の値を改善できるかどうかを考えるために，シンプレックス表を作る．もし改善できなければシンプレックス法を終了する．改善できる場合は，目的関数の値を最も大きく改良する非基底変数を 1 つ選ぶ．ここでは，目的関数における係数が最も大きい非基底変数である x_2を選ぶ．

ステップ4. 現行の基底変数 x_4, x_5, x_6 の中から非基底変数にする変数を選ぶ．ステップ 3 で選んだ非基底変数を増加させると，等式制約を満たすように，現行の基底変数の値を減少させなければならない．したがって，現行の基底変数が負にならない範囲で，選んだ非基底変数を増加させる．ここでは非基底変数 x_2 を 3 まで増加させることができ，3 つの基底変数 x_4, x_5, x_6 はそれぞれ 1,0,5 にまで減少する．この結果，0 にまで減少する x_5 を次の非基底変数とし，x_2 が次の基底変数となる．

ステップ5. 次に基底解に入る基底変数と次に基底解から出る非基底変数の交点係数を 1 とし，その列の他の係数を 0 にするようにシンプレックス表を更新する．ここでは，x_2 および x_5 の交点変数を 1 とし，x_2 の列の他の係数，つまり，式 (2) および (4) に対応するx_2 の係数を 0 にする．このようにシンプレックス表を更新すると，実行基底解も更新されるので，ステップ 3 にもどってこれまでの手順を繰り返す．

問 3-1: ステップ 2 で求めた最初の実行基底解を求めよ．

問 3-2: ステップ 3 で最初（0 回目）に作ったシンプレックス表を解答欄にしたがって作成せよ．

問 3-3: 2 回目のステップ 3 および 4 で，次に基底解に入る基底変数と次に基底解から出る非基底変数を求め，ステップ 5 で更新された実行基底解を示せ．

問 3-4: 式 (1)-(5) の最大化問題の最適解（各変数の値）とその解における目的関数の値をシンプレックス法で求めよ．なお，解答欄にしたがってシンプレックス表を完成させよ．ただし，0 回目の表は問 3-2 で解答しているので，ここには 1 回目から 3 回目までシンプレックス表を書くこと．

Keywords　*最大化問題: maximization problem, シンプレックス法: simplex method, 不等式制約: inequality constraints, スラック変数: slack variable, 等式制約: equality constraints, 非: non, 基底変数: basic variable, 実行基底解: basic feasible solution, 非負: non-negative, 目的関数: objective function, 係数: coefficient, 表: table, 最適解: optimal solution, 完成: complete*

選択問題

情報学専攻

科目の番号

3 離散数学

注意：離散数学の**問 1〜問 3** はマークシートに解答しなさい．
解答にあたっては，$\boxed{1}$ 〜 $\boxed{30}$ に当てはまる最も適切なものを，
選択肢から選びなさい．

問 1. 以下の論理関数 $P(x,y,z)$ について考える．次の空欄に当てはまる記号を，次ページの選択肢から選べ．なお，同じ番号の空欄には同じものが入るが，異なる番号の空欄に同じものが入ることもある．また，順不同の箇所はどの順序で解答しても正解とする．

$$P(x,y,z) = \neg(x \Rightarrow (y \wedge \neg z))$$

(1) 論理関数 $P(x,y,z)$ を連言標準形に書き換えると，

$$\boxed{1} \wedge (\boxed{2} \vee \boxed{3})$$

と表される．

(2) 論理関数 $P(x,y,z)$ を選言標準形に書き換えると，

$$(\boxed{4} \wedge \boxed{5}) \vee (\boxed{4} \wedge \boxed{6})$$

と表される．

論理関数：logic function, 連言標準形：conjunctive normal form, 選言標準形：disjunctive normal form

【次のページへ続く】

選択問題

情報学専攻

科目の番号

3 離散数学

【前のページから続く】

(3) 論理関数 $P(x,y,z)$ の真理値表を完成させよ.

x	y	z	$P(x,y,z)$
T	T	T	$\boxed{7}$
T	T	F	$\boxed{8}$
T	F	T	$\boxed{9}$
T	F	F	$\boxed{10}$
F	T	T	$\boxed{11}$
F	T	F	$\boxed{12}$
F	F	T	$\boxed{13}$
F	F	F	$\boxed{14}$

選択肢：
0 x 1 y 2 z 3 $\neg x$ 4 $\neg y$ 5 $\neg z$ 6 $\forall x$ 7 $\exists x$ 8 T 9 F

真理値表：truth table

【次のページへ続く】

選択問題

情報学専攻

科目の番号

3　離散数学

【前のページから続く】

問 2.　x および y をそれぞれ 0 以上 3 以下の整数とする．図 1 の 4×4 のマスで表される格子において，x 行目 y 列目のマスを $C(x, y)$ と表す．たとえば図 1 では 1 行目 3 列目のマスの値が 1 であるため，$C(1, 3) = 1$ である．

図 2〜図 4 それぞれについて真となる命題を，2 つずつ選択肢より選べ．異なる番号の空欄に同じものが入ることもある．また，順不同の箇所はどの順序で解答しても正解とする．

	0	1	2	3
0				
1				1
2				
3				

図 1

	0	1	2	3
0			1	
1		1		
2				1
3	1			

図 2

	0	1	2	3
0				
1	1	1	1	1
2				
3				

図 3

	0	1	2	3
0	1			1
1		1		1
2		1		1
3	1			1

図 4

図 2 について真となる命題は，$\boxed{15}$ および $\boxed{16}$ である．

図 3 について真となる命題は，$\boxed{17}$ および $\boxed{18}$ である．

図 4 について真となる命題は，$\boxed{19}$ および $\boxed{20}$ である．

選択肢：

0 $\forall x \forall y (C(x, y) = 1)$　1 $\forall x \exists y (C(x, y) = 1)$　2 $\forall y \exists x (C(x, y) = 1)$

3 $\exists x \forall y (C(x, y) = 1)$　4 $\exists y \forall x (C(x, y) = 1)$

整数：integer, 真：true, 命題：proposition

【次のページへ続く】

選択問題

情報学専攻

科目の番号

3 離散数学

【前のページから続く】

問 3. 集合 X, Y に対して $|X \cap Y| = |X \cup Y|$ であることと $X = Y$ が同値であることを示したい．次の文章の空欄に当てはまるものを，以下の選択肢から選べ．

$X = Y \Rightarrow X \cap Y = X \cup Y$ は明らかなので，$X = Y$ であることは $|X \cap Y| = |X \cup Y|$ となるための $\boxed{21}$ 条件である．以下では，$X = Y$ が $\boxed{22}$ 条件であること示す．

和集合 と 積集合 の定義から

$$\boxed{23} \subseteq X \subseteq \boxed{24} \qquad \cdots\cdots\cdots\cdots\cdots\cdots\cdots\ ①$$

が成り立つ．仮定から $|X \cap Y| = |X \cup Y|$ であることを用いると，

$$|X| = \boxed{25} \qquad \cdots\cdots\cdots\cdots\cdots\cdots\cdots\ ②$$

となる．ここで $X = \boxed{23} \cup \boxed{26}$ であり，$\boxed{23}$ と $\boxed{26}$ は排反であるから，$|X| = \boxed{27} + \boxed{28}$ となるので，式 ② から $\boxed{28} = 0$ すなわち $\boxed{26} = \emptyset$ が従う．ゆえに，$X \boxed{29} Y$ が成り立つ．

式 ① を Y に関する式に変えて同様の議論を行うと $X \boxed{30} Y$ が示せるので，$X = Y$ が導かれる．（証明終わり）

選択肢：
⓪ 必要　① 十分　② $X \cap Y$　③ $|X \cap Y|$　④ $X \cup Y$　⑤ $|X \cup Y|$
⑥ $X \backslash Y$　⑦ $|X \backslash Y|$　⑧ $\subseteq$　⑨ $\supseteq$

集合: set, 同値: equivalence, 和集合: union, 積集合: join, 排反: disjoint

【次のページへ続く】

選択問題

情報学専攻

科目の番号

3 離散数学

【前のページから続く】

問 4. 以下の問いに答えよ.

(1) 集合 X に対する任意の写像 $f : X \to X$ と, X の任意の部分集合 A に対して一般に次の (a), (b) が成り立つ. $\boxed{1}$, $\boxed{2}$ に当てはまる記号を以下の枠から選べ.

(a) $f^{-1} \circ f(A)$ $\boxed{1}$ A.

(b) $f \circ f^{-1}(A)$ $\boxed{2}$ A.

(1) の選択肢：

$=, \subset, \subseteq, \supset, \supseteq, \in, \ni$

(2) 一般に $f^{-1} \circ f = f \circ f^{-1}$ が成り立つかを調べたい. この式が成り立つことを証明するために確認する関係式を, 次の枠内の式や記号を用いて書け.

(2) の選択肢：

$\forall x \in X,\ \exists x \in X,\ \forall a \in A,\ \exists a \in A,\ \forall A \subseteq X,\ \exists A \subseteq X,$

$f^{-1} \circ f(x) = f \circ f^{-1}(x),\ f^{-1} \circ f(a) = f \circ f^{-1}(a),$

$f^{-1} \circ f(X) = f \circ f^{-1}(X),\ f^{-1} \circ f(A) = f \circ f^{-1}(A).$

(3) 小問 (2) で答えた関係式が成り立つ場合には証明を書き, 成り立たない場合にはそのような f の反例を示した上で, この関係式が成り立たないことを確認せよ.

集合: set, 写像: map, 部分集合: subset, 反例: counterexample

【次のページへ続く】

選択問題

情報学専攻

科目の番号

3　離散数学

【前のページから続く】

問 5． $L_0 = 2, L_1 = 1, L_{n+1} = L_n + L_{n-1} (n \geq 1)$ により定義される数列がある。n が正の整数であるとき，以下の等式が成り立つことを数学的帰納法を用いて証明せよ．

$$L_n^2 = L_{n-1}L_{n+1} + 5(-1)^n$$

数列：numerical sequence, 正の整数：positive integer, 等式：equation, 数学的帰納法：mathematical induction

【離散数学の問題はここまで】

選択問題

情報学専攻

科目の番号

4　計算機工学　[4-1]

1. 以下の問いに答えなさい。

   (1) 言語 $\{a^m b^n \mid m, n \geq 1\}$ に属する記号列を３つ書きなさい。また、この言語を生成する、プロダクションの数が６個の正則文法を書きなさい。

   (2) 言語 $\{a^m b^m \mid m \geq 1\}$ に属する記号列を３つ書きなさい。また、この言語を生成する、プロダクションの数が２個の文脈自由文法を書きなさい。

2. 入力記号が a, b であり、a と b からなる記号列のうち、連続する 2 個の b を含まない記号列を受理する 3 状態の有限オートマトンを M とする。このとき、以下の問いに答えなさい。（例えば、M は aba は受理するが、$abba$ は受理しない。）

   (1) M の状態遷移図を図示しなさい。その際、初期状態は矢印で指し、最終状態は二重丸で囲みなさい。

   (2) M の状態遷移関数を書きなさい。

3. 次の文法について、以下の問いに答えなさい。ただし、以下の文法の非終端記号は S, A, B、終端記号は a, b とし、開始記号は S とする。

$$S \to aA,\ S \to bB,\ B \to bB, B \to aA,$$
$$A \to aB, A \to bA,\ A \to a, B \to b$$

   (1) この文法が生成する終端記号列を３つ書きなさい。

   (2) この文法が生成する終端記号列には、a が偶数個含まれることを証明しなさい。

正則文法：regular grammar, 文脈自由文法：context-free grammar, 非終端記号：nonterminal symbol, 終端記号：terminal symbol, 開始記号：start symbol, 有限オートマトン：finite automaton, 状態遷移図：state transition diagram

選択問題

情報学専攻

科目の番号

4 計算機工学 [4-2]

1. 以下の問いに答えよ.

(1) 8 進数の小数$(2.22)_8$ を10 進数の分数で書け.

(2) 2 の補数で表された負数 111111111 の絶対値を 10 進数で書け.

(3) 16 ビットの値 x が与えられた時、6 ビット右ローテートの結果を変数y とする処理を、シフト演算とビット論理演算で表現せよ. ただし、x ビット右ローテートとは、x ビット右シフトであふれたx ビットを上位ビットに循環するものである.

2. a, b, c_{in} を入力とし、和s と桁上がりc_{out} を出力とする全加算器について、以下の問いに答えよ.

(1) 全加算器の桁上がり c_{out} をできるだけ簡単な積和型の論理式で表せ.

(2) 二つ半加算器を使った全加算器の回路図を描け.

3. 下表の性能を持つ計算機 A を使って 1000 命令のプログラムを実行した. なお、計算機 A のクロック周波数と 1 インストラクションあたりの命令数 (CPI) はビジネスモードとゲームモードで異なっている.

	クロック周波数	CPI
ビジネスモード	3GHz	30
ゲームモード	4GHz	?

(1) 計算機 A のビジネスモードにおけるプログラムの処理時間を答えよ.

(2) 計算機 A のゲームモードではビジネスモードに比べ処理時間を半減できる. 計算機 A のゲームモードの CPI を求めよ.

4. 計算機 B の設計において、キャッシュメモリのサイズを大きくすると、キャッシュミス率は下がるが、ヒット時間は増大する傾向にある. ただし、ミスペナルティ時間はヒット時間の 10 倍で、キャッシュサイズに依存しないとする. 以下の問いに答えよ.

(1) ヒット時間が 1ns、ミスペナルティ時間が 10ns のメモリシステムを考える. キャッシュミス率が 10%のときの平均メモリアクセス時間を求めよ.

(2) (1) のシステムにおいて、平均メモリアクセス時間を 1.5ns とするために、必要なキャッシュミス率を求めよ.

(3) 計算機 B のキャッシュサイズを 2 倍にすることによって、キャッシュミス率が 20%から 10%に改善できることがわかった. このとき平均メモリアクセス時間を短縮するためには、ヒット時間の増加はキャッシュサイズの変更前の何倍に抑えられている必要があるかを答えよ.

8 進数:octal number, 小数:decimal number, 10 進数:decimal number, 分数 fractional number, 2 の補数:2's complement, 負数:negative number, 絶対値:absolute value, 右ローテート:right rotate, 変数:variable, シフト演算:shift operation, ビット論理演算:bit-wise logical operation, 循環 circulattion, 和: summation, 桁上がり:carry, 全加算器:full adder, 積和型:sum of products, 論理式:logical formula, 半加算器:half adder, 回路図:circuit diagram, クロック周波数:clock frequency, CPI:cycles per instruction, ビジネスモード:business mode, ゲームモード:game mode, 処理時間:processing time, 半減:half reduced, キャッシュメモリ:cache memory, サイズ:size, キャッシュミス率:cache miss rate, ヒット時間:hit time, ミスペナルティ時間:miss penalty time, メモリシステム:memory system, 平均メモリアクセス時間:average memory access time, 抑えられている: suppressed