
問題1、2、3、4の中から2問を選んで解答せよ。選択した問題は（ ）と（ ）です。
※選択した問題の番号を、上記 下線部のカッコ内に記入 すること。

問題1 次の(1)～(3)の問いに答えなさい。

(1) 「整数」とは何かを例示しながら説明しなさい。

解答 $\dots -3, -2, -1, 0, 1, 2, 3 \dots$

(2) 「自然数」とは何かを例示しながら説明しなさい。

解答 0を含まない正の整数であり、日常で物を数えるとき使う1、2、3、4……

(3) 「 $xy - 4x - 3y + 6 = 0$ 」を満たす自然数 x と y の組をすべて求めなさい。

解答 与えられた関係式は

$$(x - 3)(y - 4) = 6$$

と変形できる。

従って、 $x - 3$ と $y - 4$ の値の組み合わせ $(x - 3, y - 4)$ は

$$(1, 6), (6, 1), (2, 3), (3, 2)$$

$$(-1, -6), (-6, -1), (-2, -3), (-3, -2)$$

のいずれかであり、 (x, y) の組み合わせは

$$(4, 10), (9, 5), (5, 7), (6, 6)$$

$$(2, -2), (-3, 3), (1, 1), (0, 2)$$

のいずれかであり、正の整数であることを考慮すると、

$$(x, y) = (4, 10), (9, 5), (5, 7), (6, 6), (1, 1)$$

の5通りであることがわかる。

受験科目名
数学



問題2 次の(1)～(2)の問いに答えなさい。

(1) 「 $f(x) = 2x^3 - 3x^2 - 4x + 5$ 」この関数を微分した結果を求めなさい。

解答 $f'(x) = 6x^2 - 6x - 4$

(2) 次の常微分方程式の一般解を求めなさい。

$$\frac{dy}{dx} = y(2 - y)$$

解答 これは変数分離型の方程式である。両辺を $y(2 - y)$ で割って整理すると、

$$\int \frac{dy}{y(2-y)} = \int dx$$

この式を積分して

$$\log \left| \frac{y}{2-y} \right| = x + C' \quad (C' \text{ は定数})$$

よって

$$\frac{y}{2-y} = \pm e^{x+C'} = \pm e^{C'} \cdot e^x$$

これを y について解くと

$$y = C \cdot e^x (2 - y) \quad (C \text{ は定数})$$

$$y(C \cdot e^x + 1) = 2C \cdot e^x$$

$$y = \frac{2C \cdot e^x}{C \cdot e^x + 1}$$

$$y = \frac{2e^x}{e^x + \frac{1}{C}}$$

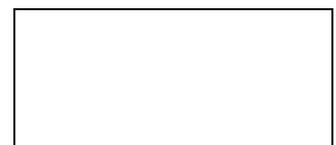
ここで、新たに C'' を導入すると、一般解は次のようになる。

$$y = \frac{2e^x}{e^x + C''} \quad (C'' \text{ は任意定数、} C'' = \frac{1}{C} \text{)}$$

受験科目名

数学

[2 / 4 頁]



問題3 次の行列Aに対して、行列Aの転置行列 A^T と、行列Aと A^T との積 AA^T を計算しなさい。

$$A = \begin{pmatrix} 2 & 0 & 2 & 4 \\ 0 & 8 & 0 & 2 \end{pmatrix}$$

解：

転置行列 A^T は以下、

$$A^T = \begin{pmatrix} 2 & 0 \\ 0 & 8 \\ 2 & 0 \\ 4 & 2 \end{pmatrix}$$

行列Aと転置行列 A^T の積 AA^T は以下、

$$AA^T = \begin{pmatrix} 2 & 0 & 2 & 4 \\ 0 & 8 & 0 & 2 \end{pmatrix} \begin{pmatrix} 2 & 0 \\ 0 & 8 \\ 2 & 0 \\ 4 & 2 \end{pmatrix} = \begin{pmatrix} 2*2+0*0+2*2+4*4 & 2*0+0*8+2*0+4*2 \\ 0*2+8*0+0*2+2*4 & 0*0+8*8+0*0+2*2 \end{pmatrix}$$

よって

$$AA^T = \begin{pmatrix} 24 & 8 \\ 8 & 68 \end{pmatrix}$$

受験科目名
数学



問題 4 京成大久保駅は、千葉県習志野市に位置する京成電鉄の駅です。この駅周辺には、大学や公園、商店街などがあり、学生や地元の人々に親しまれています。

ある日、あなたは最近付き合い始めた恋人と京成大久保駅の近くにあるカフェで待ち合わせをすることにしました。約束の時間は午後 3 時です。しかし、二人とも普段から少し遅れがちなので、実際には何分か遅れて到着するかもしれません。あなたがカフェに到着する時間は、午後 3 時から午後 3 時 10 分の間どこかで一様分布に従います。恋人が到着する時間も同じく午後 3 時から午後 3 時 10 分の間どこかで一様分布に従います。

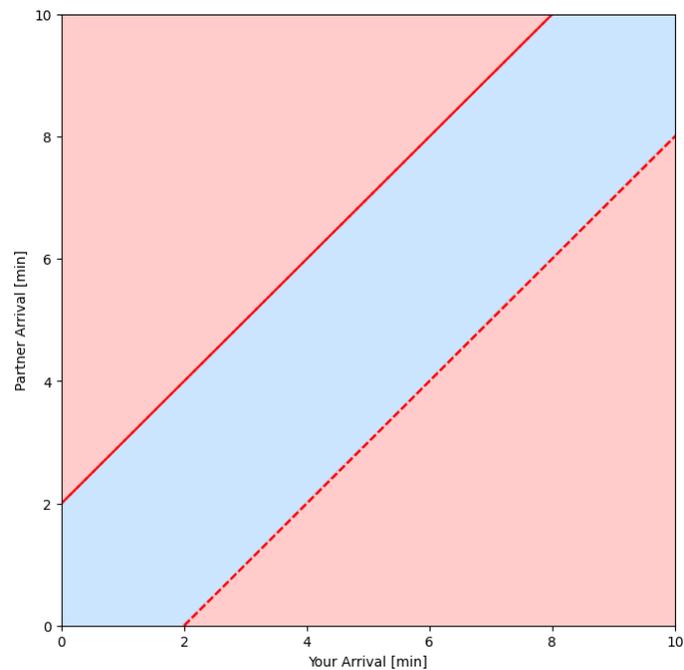
到着した時間差が 2 分以内であるとき、カフェのポイントカードに特別なスタンプをもらえるというルールがあります。このスタンプは貯めると、カフェでのドリンクが無料になる特典があるため、二人はこのルールに従って早めに到着しようとしてます。

さて、二人の到着時間差が 2 分以内である確率 [%] を求めなさい。

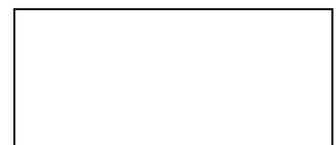
解：

あなたと恋人の到着時間 X と Y は、それぞれ 0 分から 10 分の間でランダムに分布している。

$|X-Y| \leq 2$ の条件を満たす領域は、平面上で対角線に挟まれた帯状の部分となる。正方形全体の面積は 100 で、条件を満たさない三角形の合計面積が 64、よって条件を満たす領域の面積は 36 となる。従って、条件を満たす確率は 36 (%) となる。



受験科目名
数学



問題 1, 2, 3, 4 の中から 2 問を選んで答えよ (右欄に選択した問題番号を記入)。【選択した問題：()と()】

問題 1

(1) 次の集合 A の真部分集合を全て列挙せよ。

$$A = \{a, b, c, d\}$$

$$\phi, \{a\}, \{b\}, \{c\}, \{d\}, \{a, b\}, \{a, c\}, \{a, d\}, \{b, c\}, \{b, d\}, \{c, d\}, \{a, b, c\}, \{a, b, d\}, \{a, c, d\}, \{b, c, d\}$$

(2) 次の集合 A の直積 $A \times A$ を書きなさい。

$$A = \{a, b, c, d\}$$

$$A \times A = \{(a, a), (a, b), (a, c), (a, d), (b, a), (b, b), (b, c), (b, d), (c, a), (c, b), (c, c), (c, d), (d, a), (d, b), (d, c), (d, d)\}$$

(3) 次の集合 A 上の関係 $R \subseteq A \times A$ が「半順序関係である」あるいは「半順序関係でない」ことを証明せよ。

$$A = \{a, b, c, d\}$$

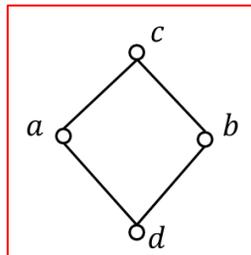
$$R = \{(a, a), (a, b), (a, d), (b, b), (b, d), (c, d), (d, d)\}$$

$(c, c) \notin R$ より、反射律を満たさないので R は半順序関係ではない。

(4) 次の半順序集合 (A, R) のハッセ図を書きなさい。(集合 A の要素とハッセ図の点の対応がわかるように記載せよ)

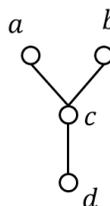
$$A = \{a, b, c, d\}$$

$$R = \{(a, a), (a, c), (b, b), (b, c), (c, c), (d, a), (d, b), (d, c), (d, d)\}$$



(5) 次のハッセ図で表される半順序集合 (A, R) について関係 R を書きなさい。

$$A = \{a, b, c, d\}$$



$$R = \{(a, a), (b, b), (c, a), (c, b), (c, c), (d, a), (d, b), (d, c), (d, d)\}$$

受験科目名
情報数学



問題 2

(1) 次の論理式を真理値分析せよ。

$\neg P \supset (Q \vee R)$

P	Q	R	$\neg P$	$Q \vee R$	$\neg P \supset (Q \vee R)$
1	1	1	0	1	1
1	1	0	0	1	1
1	0	1	0	1	1
1	0	0	0	0	1
0	1	1	1	1	1
0	1	0	1	1	1
0	0	1	1	1	1
0	0	0	1	0	0

(2) 命題論理におけるトートロジーとはどのような論理式か、具体例とその真理表を書いて説明せよ。

トートロジーとは原子式の真理値によらずに常に真となる論理式である。

例： $P \vee \neg P$

真理表：

P	$\neg P$	$P \vee \neg P$
1	0	1
0	1	1

(3) 下記の公理 1 と公理 2 と導出規則を使って論理式 $A \supset A$ を証明せよ。

公理 1： $(A \vee A) \supset A$

公理 2： $A \supset (A \vee B)$

導出規則： $A \supset B$ と $B \supset C$ から、 $A \supset C$ を導出してよい

※ただし、上記の A, B, C は任意の論理式を表している。

$A \supset (A \vee A)$ 公理 2 の B に A を代入する

$(A \vee A) \supset A$ 公理 1

$A \supset A$ 導出規則の B に $(A \vee A)$ 、 C に A を代入する

(4) 命題論理の公理系の完全性を説明せよ。

命題論理の公理系の完全性とは、全てのトートロジーが定理として証明されうることである。

(5) 命題論理の公理系の無矛盾性を説明せよ。また、公理系が無矛盾であることが望ましい理由を説明せよ。

命題論理の公理系の完全性とは、矛盾式 $D \wedge \neg D$ が定理として証明されえないことである。

公理系が無矛盾であることが望ましい理由は、矛盾式 $D \wedge \neg D$ が仮定なしで証明されると背理法によって全ての論理式が証明可能になって、公理系が実質的な意味を失うから。

受験科目名
情報数学



問題 3

(3.1) 以下の 8 ビットの 2 の補数表示された 2 進数の計算を行い、結果を 2 進数で示しなさい。なお、オーバーフローの場合は、解答欄にオーバーフローと記述しなさい。

(1) $00011000 - 00000110$

0	0	0	1	0	0	1	0
---	---	---	---	---	---	---	---

(2) $10100000 - 10110100$

1	1	1	0	1	1	0	0
---	---	---	---	---	---	---	---

(3.2) IEEE754 形式の浮動小数点表現 (単精度) に関して、以下の問いに答えなさい。

(1) 指数部 (8 ビット) に加える数値 (バイアス) を 10 進数で答えなさい。

(1) 解答欄

(2) (1) の数値 (バイアス) を加える理由を説明しなさい。

(2) 解答欄

符号を表すために 1 ビット使用すると、8 ビットで表せる数値の範囲が狭まってしまう。これを避けるために、10 進数で -127~128 の値を指数部で表現する際、直接 2 進数に変換せず、バイアスの数値を加えて 0~255 の値にしてから 2 進数に変換するのである。

(3) 指数部が 01111100 のとき、10 進数では 乗を意味する。空欄に入る数値を答えなさい。

(3) 解答欄

(3.3) 2-to-4 デコーダについて以下の問いに答えなさい。ただし、2-to-4 デコーダとは「 d_1, d_0 で表される 2 ビット符号なし 2 進数に対応する番号」の出力が 1 となる回路のことである。なお、 d_1 が上位ビット、 d_0 が下位ビットである。

表 3-3-1. 2-to-4 デコーダの真理値表

(1) d_1, d_0 で表せる 2 進数は 10 進数の 0~3 である。0, 1, 2, 3 に対応する出力がそれぞれ o_0, o_1, o_2, o_3 であるとき 2-to-4 デコーダの真理値表を作成したい。表 3-3-1 の空欄を埋めなさい。

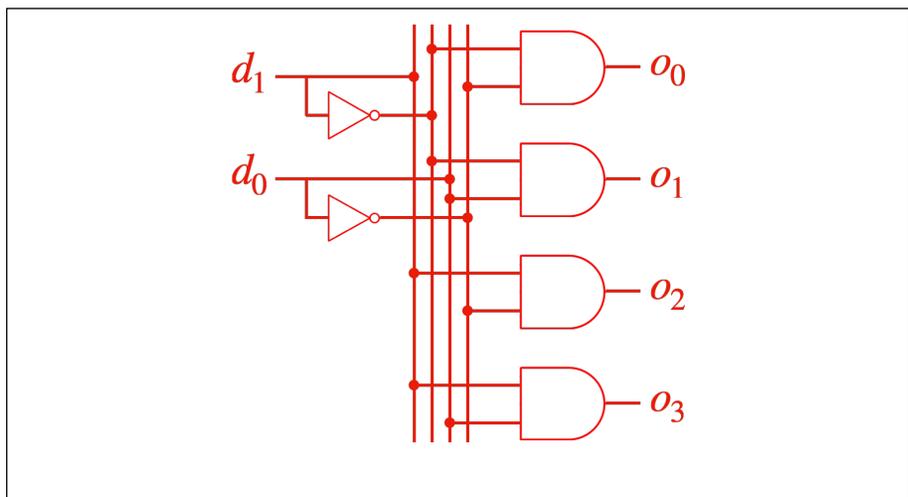
d_1	d_0	o_0	o_1	o_2	o_3
0	0	1	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	1

(2) o_0, o_1, o_2, o_3 の論理式をそれぞれ示しなさい。

$o_0 = \bar{d}_1 \cdot \bar{d}_0$, $o_1 = \bar{d}_1 \cdot d_0$, $o_2 = d_1 \cdot \bar{d}_0$, $o_3 = d_1 \cdot d_0$

※ (2) において「 \cdot 」は省略可。

(3) (2) の論理式に基づいて 2-to-4 デコーダの回路を設計し、右側の空欄に回路図として示しなさい。ただし、使用可能な論理ゲートは、NOT ゲート、2 入力 AND ゲートとする。



受験科目名
情報数学

問題 4

表 4-1 は MIPS アセンブリ言語表現の命令の一部である。メモリ及びレジスタのデータ幅 32 ビット、アドレス幅 32 ビットとするとき、以下の問いに答えなさい。

表 4-1. MIPS アセンブリ言語表現の命令の一部 (rd, rs, rt : レジスタ, PC : プログラムカウンタ, 命令幅 : 32 ビット)

命令	意味	補足
add rd, rs, rt	$rd = rs + rt$	レジスタの値の加算
addi rt, rs, immd	$rt = rs + SE(immd)$	immd は即値, SE(x) は x を 32 ビットに符号拡張
slt rd, rs, rt	if(rs < rt) rd = 1, else rd = 0	beq 及び bne 命令の直前で使用することが多い
slti rt, rs, immd	if(rs < SE(immd)) rt = 1, else rt = 0	slti は即値との比較 (slt はレジスタ同士の比較)
beq rs, rt, Label	if(rs == rt) PC = Label, else PC = PC + 4	PC = Label で, PC を Label のところにセット
bne rs, rt, Label	if(rs != rt) PC = Label, else PC = PC + 4	PC = PC + 4 で, PC は次の命令(4 バイト先)に進む
lw rt, x(rs)	$rt = mem[(rs) + x]_{32}$	(rs の値 + x) のメモリ番地のデータを rt にロード
sw rt, x(rs)	$mem[(rs) + x]_{32} = rt$	(rs の値 + x) のメモリ番地に rt の値をストア
j Label	PC = Label	PC を強制的に Label のところにセット

メモリ上に n 個の整数データ x_1, x_2, \dots, x_n の値が連続して書き込まれているとする。 x_1, x_2, \dots, x_n をメモリから順番に読み込んでいくとき、 x_1, x_2, \dots, x_n の 0 以上 9 以下の数値のみを合計したものを s_{09} とする。 s_{09} を求め、メモリに保存する MIPS アセンブリコードを書きなさい。なお、 $\$s0 \sim \$s7$ 及び $\$t0 \sim \$t9$ はレジスタで整数値やアドレスなどを保存でき、 $\$zero$ は定数 0 が常に入っている特別なレジスタである。これらのレジスタを適宜使用してコード作成をするものとする (必要なレジスタだけを使用すればよく、全部のレジスタを使う必要はない)。

コードを書く際の前提として、データ数 n の値が格納されているメモリアドレスはレジスタ $\$s7$ に、 x_1 のアドレスはレジスタ $\$s6$ に入っており、 **End** のラベルが付いた行の命令を実行後、正常終了するものとする。また、 s_{09} は n の値が格納されている領域の隣 ($\$s7$ に入っているアドレスに 4 加えた領域) に保存されるものとする。

(例) $\$s7$ に格納されているアドレスが $0x10010038$ で、 $[x_1, x_2, \dots, x_n] = [5, 11, 24, -2, 0, 3, 31, -5, 1, 9]$ の場合、 $s_{09} = 18$ となり、 $0x1001003C$ の領域に s_{09} が保存される。

【解答欄 (最初の命令と最後の命令は記載されているため、その他のコードを追記する)】

```

lw $s0, 0($s7)      # n の値をロード
add $s1, $zero, $zero # i = 0 に初期化
add $s2, $zero, $zero # s09 = 0 に初期化 (Endで$s2を使っているため$s2の必要あり)
Loop: slt $t0, $s1, $s0 # if(i < n) $t0 = 1, else $t0 = 0
      beq $t0, $zero, End # if($t0 == 0) PC = End
      lw $s3, 0($s6)      # x_iの値をロード
      slti $t0, $s3, 10   # if(x_i < 10) $t0 = 1, else $t0 = 0
      beq $t0, $zero, L1  # if($t0 == 0) PC = L1
      slti $t0, $s3, 0    # if(x_i < 0) $t0 = 1, else $t0 = 0
      bne $t0, $zero, L1  # if($t0 != 0) PC = L1
      add $s2, $s2, $s3   # s09 = s09 + x_i (Endで$s2を使っているため$s2に足し込む)
L1:   addi $s1, $s1, 1     # i++
      addi $s6, $s6, 4    # アドレスを4バイトずらして次のデータを参照できるようにする
      j Loop              # Loopのラベルにジャンプ
End:  sw $s2, 4($s7)      # s09 の値を n の隣の領域に書き込む (この行の実行後に正常終了)

```

受験科目名
情報数学



問題 1, 2, 3, 4 の中から 2 問を選んで答えよ。選択した問題は () と () です。

〔問題 1〕ある学校が教員募集した。担当してもらいたい科目 (S1~S8) と教員に応募してきた人物 (A~G) が担当可能な科目情報を表 1 に示す。表 1 において、ある教員応募人物が、ある科目を担当可能な場合は、丸印で表現している。例えば、A は科目 S1 の担当が可能であることを示している。分枝限定法を用いて、最小人数の採用で全科目の担当が可能になるように、採用する教員の集合を求めなさい。ただ解を出すだけでなく、分枝限定処理の中で限定処理 (解空間の枝刈り) が可能な箇所を説明して、変更して簡約化した表を示しながら解を求めなさい。この問題では、限定処理のみで分枝処理 (教員の採用・非採用の場合分け) は発生しないで、解を求めることができる。

表 1 教員応募人物の科目担当可能表

人物	S1	S2	S3	S4	S5	S6	S7	S8
A	○	○	○					
B			○			○		
C	○		○		○			
D		○		○				
E				○			○	
F						○		○
G							○	○

【解を求めるための説明】

S5 の科目を担当できるのは C のみであるので、C を解に入れる。解={C}

C は S1, S3 も担当できるので、表 1 から C, S1, S3, S5 の行と列を削除し簡約化したものを表(a)に示す。

表(a)において、F は B を、D は A を支配しているので、表(a)の A, B, E を削除すると、表(b)になる。

表(b)において、S8 は S7 を、S4 は S2 を支配しているので、S7 を担当する人物を選択すると必ず S8 を担当することができる、S2 を担当する人物を選択すると必ず S4 を担当することができる。

S8 と S4 を表(b)から削除すると、表(c)ができる。

表(c)において、S2, S6 を担当できる人物はそれぞれ 1 人だけなので、その人物である D, F, を解に入れる。

解= {C, D, F}

表(c)から D, F, S2, S6 の行と列を削除すると表(d)になる。

表(d)において、G は E を支配している (逆でも OK) のので、E を削除すると、表(e)になる。

表(e)において、S7 を担当できるのは G だけであるので、G を解に含める。解= {C, D, F, G}

表(e)から G と S7 を削除すると表はなくなる。

表(a)						表(b)						表(c)				表(d)		表(e)	
人物	S2	S4	S6	S7	S8	人物	S2	S4	S6	S7	S8	人物	S2	S6	S7	人物	S7	人物	S7
A	○					D	○	○				D	○			E	○	G	○
B			○			E		○		○		E			○	G	○		
D	○	○				F			○		○	F		○					
E		○		○		G				○	○	G			○				
F			○		○														
G				○	○														

【解】採用する教員の集合= { C, D, F, G }

受験科目名
アルゴリズムとプログラミング



[問題2] 個人情報である学生番号、氏名、年齢、成績をハッシュ表で管理しているソフトウェアを考える。ハッシュ表はチェーン法を採用しており、ハッシュ値が衝突したときには、線形リストを使用して管理している。C 言語で記述された関数 `chain_hash` は引数としてキーである学生番号の変数 `key` とハッシュ表である配列の先頭アドレスの変数 `hash_tbl` が入力され、学生番号が一致するデータをハッシュ探索し、その個人情報のアドレスを返却する機能を持つ。ただし、学生番号が `key` に一致するデータがない場合は `NULL` を返却する。`NULL` はどのアドレスも格納されていない場合に使用されるものであるとする。個人情報に対応する構造体が下記のように定義されているものとし、ハッシュ表の配列も下記のように型宣言されているものとする。またキーである学生番号からハッシュ値への変換は学生番号を 11 で割った剰余とする。C 言語で記述された関数 `tran_hash` は引数として学生番号の変数 `key` を入力し、ハッシュ値に変換して、そのハッシュ値を返却する機能を持つ。関数 `chain_hash` と関数 `tran_hash` のプログラムを完成させなさい。

```
struct personal_info          /* 個人情報 */
{
    int    id;                /* 学生番号 */
    char   name[20];          /* 氏名 */
    int    age;               /* 年齢 */
    int    score;            /* 成績 */
    struct personal_info *next; /* 後続ノードのアドレス */
};

struct personal_info * hash_tbl[11]; /* ハッシュ表 */

struct personal_info *chain_hash(int key, struct personal_info **hash_tbl)
{
    int i, struct personal_info *p;
    i = tran_hash(key);
    for (p = hash_tbl[i]; p != NULL; p=p->next)
    {
        if (p->id == key)
        {
            return p;
        }
    }
    return NULL;
}

int tran_hash(int key)
{
    return (key%11);
}
```

受験科目名
アルゴリズムとプログラミング



問題 3.

- (1) オブジェクト指向の三大要素を答えよ。
(2) 前問のオブジェクト指向の三大要素のうちいずれか1つを選択し、選択した要素を説明せよ。

解答例：

・カプセル化

データとそれを操作するメソッドを一つのオブジェクトにまとめ、その内部構造を外部から隠すことで、安全性と保守性を高める仕組み

・継承

既存のクラスの機能を、新しいクラスが引き継ぐ仕組み

・多態性 (ポリモーフィズム)

同じ名前のメソッドや関数が、呼び出されるオブジェクトのクラス (型) によって異なる振る舞いをする仕組み

受験科目名
アルゴリズムと プログラミング



問題 4.

(1) 最小二乗法を用いた回帰直線の導出について説明せよ。説明変数, 目的変数, 実測値, 予測値という単語を使うこと。

解答例:

最小二乗法を用いて回帰直線を導出するとは、実測値と予測値の差（残差）の二乗和が最小になるように、説明変数と目的変数の関係を表す一次式（回帰直線）を定めることである。

(2) 以下のデータについて、説明変数を「練習時間」、目的変数を「スコア」としたときの回帰直線を導出せよ。

プレイヤー	A	B	C	D	E
練習時間 (時間)	3	2	4	1	5
スコア (点)	240	420	600	180	800

解答例:

説明変数を x 、目的変数を y 、回帰直線を $y = ax + b$ とおく。

傾き a は、説明変数と目的変数の共分散を、説明変数の分散で割ることで求められる。

切片 b は、目的変数の平均から、傾きに説明変数の平均をかけたものを引くことで得られる。

表の値を用いて計算すると、 $a = 142$ 、 $b = 22$ となる。

したがって、回帰直線は

$$y = 142x + 22$$

である。

受験科目名

アルゴリズムと
プログラミング

{ 4 / 4 頁 }



問題 1～問題 4 の中から 2 つ選択して解答しなさい。

選択した問題：問題 () と問題 () ※選択した問題の番号をカッコ内に記入すること

AA	BB	CC	DD	EE	FF	01	02
03	04	05	06	08	00	45	00
00	2D	00	00	00	00	7F	06
00	00	0A	09	08	07	FA	2C
2C	02	10	00	00	50	00	00
10	01	70	00	20	01	50	10
20	00	00	00	00	00	47	45
54	20	2F	00	FE	90	72	CA

問題 1

(1-1) 右図はある TCP セグメントに対するイーサネットフレーム全体のデータである (16 進表記) .
 プリアンブルおよび SFD は省略されている。
 以下の数値を指定された形式で示しなさい

- (a) 始点物理アドレス(コロン区切り 16 進数表記) 01:02:03:04:05:06
- (b) 終点 IP アドレス(4 バイト 16 進数表記) FA 2C 2C 02
- (c) 始点 IP アドレス(dotted-decimal 形式) 10.9.8.7
- (d) 終点ポート番号(10 進数表記) 80
- (e) IP ヘッダ長 20 バイト, IP データグラム長 45 バイト, TCP ヘッダ長 20 バイト
 →TCP データ長 5 バイト, パディング 1 バイト

(1-2) この TCP セグメントはインターネット上のどのようなサービスに対応して送受信されたものか。
 また, クライアントとサーバのどちらから送信されたものかと考えられるか。

サービス名: WWW (web, HTTP も可) どちらから送信されたか クライアント (ブラウザも可)

参考資料

IP ヘッダフォーマット

版 (4b)	ヘッダ長 (4b)	サービスタイプ (8b)	データグラム長 (16b)
フラグメント識別子 (16b)		フラグ (3b)	フラグメントオフセット (13b)
TTL (8b)	プロトコル (8b)	ヘッダチェックサム (16b)	
始点 IP アドレス (32b)			
終点 IP アドレス (32b)			
オプション (あれば 32b)単位で追加			

TCP ヘッダフォーマット

始点ポート番号 (16b)		終点ポート番号 (16b)	
シーケンス番号(seq) (32b)			
受信確認番号(seq) (32b)			
データオフセット (4b)	予約 (6b)	制御フラグ (1b * 6)	ウィンドウ (16b)
チェックサム (16b)		緊急ポインタ (16b)	
オプション (あれば 32b)単位で追加			

イーサネットフレームフォーマット

プリアンブルおよび SFD (8B)	終点物理アドレス (6B)	始点物理アドレス (6B)	イーサネットタイプ (2B)	データ (46~1500B)	FCS (4 B)	次フレームとの間隔 (12B 以上)
--------------------	---------------	---------------	----------------	----------------	-----------	--------------------

1 バイト 16 進数⇄10 進数変換表

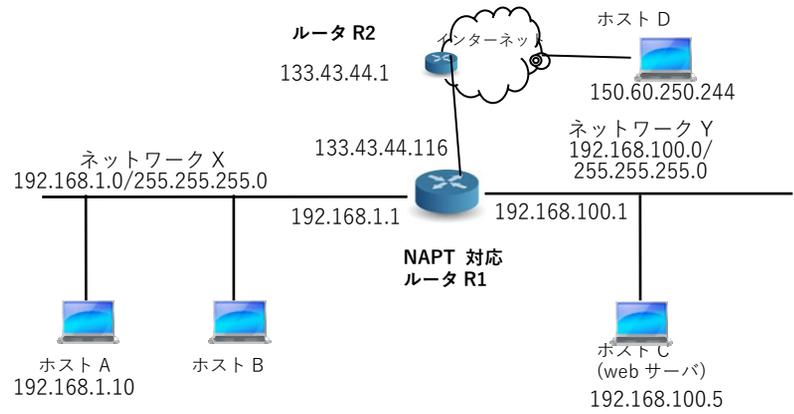
		下位															
		-0	-1	-2	-3	-4	-5	-6	-7	-8	-9	-A	-B	-C	-D	-E	-F
0-	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
1-	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	
2-	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	
3-	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	
4-	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	
5-	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	
6-	96	97	98	99	100	101	102	103	104	105	106	107	108	109	110	111	
上	112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127	
7-	128	129	130	131	132	133	134	135	136	137	138	139	140	141	142	143	
8-	144	145	146	147	148	149	150	151	152	153	154	155	156	157	158	159	
9-	160	161	162	163	164	165	166	167	168	169	170	171	172	173	174	175	
A-	176	177	178	179	180	181	182	183	184	185	186	187	188	189	190	191	
B-	192	193	194	195	196	197	198	199	200	201	202	203	204	205	206	207	
C-	208	209	210	211	212	213	214	215	216	217	218	219	220	221	222	223	
D-	224	225	226	227	228	229	230	231	232	233	234	235	236	237	238	239	
E-	240	241	242	243	244	245	246	247	248	249	250	251	252	253	254	255	
F-																	

受験科目名
情報ネットワーク

--

問題 2

右図のネットワーク構成では、
 2 個の LAN (ネットワーク X, Y) が
 ルータ R1 によって接続されている。
 X, Y はプライベートネットワークである。
 R1 は NATP 対応ルータであり、
 グローバル IP アドレスとして 133.43.44.116 を持つ。
 ホスト C はプライベートネットワーク内の web サーバ、
 ホスト D はインターネット上に存在する web サーバである。
 このとき、以下の間に答えよ。



192.168.1.xx の形式で
 xx は 2~254 のいずれか、かつ 10 以外

(1) ホスト B に設定する IP アドレスとして適切な値をひとつ答えよ (例) 192.168.1.20

(2) ネットワーク Y 上には、最大何台のマシンを接続させることができるか答えよ。

(既に接続済みのホスト C, R1 も含めた台数とする) 254 台

(3) ホスト A からホスト C に HTTP リクエストが送信されるとき、ネットワーク Y 上で R1 からホスト C に転送される IP データグラムの始点および終点 IP アドレスを答えよ

始点 IP アドレス: 192.168.1.10 終点 IP アドレス: 192.168.100.5

(4) ホスト A からホスト D に HTTP リクエストが送信されるとき、ホスト A から R1 に送信される IP データグラム、および R1 から R2 に転送される IP データグラムの始点および終点 IP アドレスを答えよ

ホスト A → R1

始点 IP アドレス: 192.168.1.10 終点 IP アドレス: 192.168.1.1

R1 → R2

始点 IP アドレス: 133.43.44.116 終点 IP アドレス: 133.43.44.1

(5) ホスト A の経路表を完成させよ。

ネットワークアドレス	サブネットマスク	インタフェース	ルータ	ホップ数
192.168.1.0	255.255.255.0	192.168.1.10	192.168.1.10	1
127.0.0.0	255.0.0.0	127.0.0.1	127.0.0.1	1
0.0.0.0	0.0.0.0	192.168.1.10	192.168.1.1	1

受験科目名
情報ネットワーク

--

問題 3

入力記号・出力記号の集合を $\{0, 1\}$ とする線形符号 $C = \{w_1, w_2, w_3, w_4\}$ を考える。
ただし、 $w_1 = 00000$, $w_2 = 11011$, $w_3 = 10101$, $w_4 = 01110$ とする。

(1) 符号 C の最小距離 d_{\min} を求めよ。

$$d_{\min} = 3$$

(2) 符号語 w_4 に対する半径 1 のハミング球 $S_1(w_4)$ を求めよ。

$$S_1(w_4) = \{01110, 11110, 00110, 01010, 01100, 01111\}$$

(3) 2つの受信語 $v_1 = 00011$ および $v_2 = 11110$ の送信語を推定せよ。

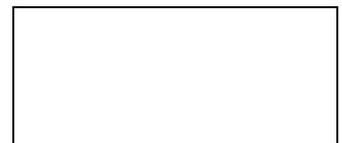
$v_2 \in S_1(w_4)$ より、 v_2 は w_4 と推定される。

v_1 はどのハミング球にも属さないので送信語を推定できない。

受験科目名

情報ネットワーク

[3 / 4 頁]



問題 4

Schnorr の認証法に関する次の問に答えなさい。

Schnorr の認証法

この認証法の安全性は \boxed{A} に基づいており、秘密鍵を秘密にしたまま秘密鍵を知っていることを証明している。このような個人認証法を \boxed{B} と呼ぶ。

登録: q を素数とし、証明者 A は秘密鍵 $s \in Z_q$ をランダムに選び、

$$v = g^{-s}$$

を計算し、 v, g, q を証明者 A の公開鍵とする。

認証:

- (a) 証明者 A は $r \in Z_q$ をランダムに選び、 $x = g^r$ を計算し、 x を検証者 B に送る。
- (b) 検証者 B は $c \in Z_q$ をランダムに選び証明者 A に送る。
- (c) 証明者 A は $y = r + sc \pmod q$ を計算し、 y を検証者 B に送る。
- (d) 検証者 B は

$$x = \boxed{C}$$

が成り立つかどうか検証し、成り立てば受理し、そうでなければ拒否する。

- (1) \boxed{A} と \boxed{B} と \boxed{C} に当てはまる語句や数式を書きなさい。

\boxed{A} : 離散対数問題

\boxed{B} : 零知識型認証法

\boxed{C} : $g^y v^c$

- (2) なりすましに成功するアルゴリズム A' が存在したと仮定すると、秘密鍵 s を効率的に求めることができるアルゴリズム M が存在することを示しなさい。

(i) M は A' を証明者、 M 自身が検証者として実行し、 (x, c_1, y_1) とする。

(ii) M は A' を初期状態にリセットし、再度実行し、 (x, c_2, y_2) とする。ただし、 c_2 はランダムに選びなおす。

(iii) $c_1 \neq c_2$ ならば、

$$x = g^{y_1} v^{c_1} = g^{y_2} v^{c_2}$$

より、

$$v = g^{(y_1 - y_2)/(c_1 - c_2)}$$

となり、

$$s = -(y_1 - y_2)/(c_1 - c_2) \pmod q$$

受験科目名

情報ネットワーク

[4 / 4 頁]

問題 1～4 から 2 問選択して答えなさい：選択した問題番号【 】【 】

【問題 1】

- A) 3 次元コンピュータグラフィックスの統合ソフトウェアを用いて金属の質感を持つ青い球のモデルを作成するための作業内容を説明しなさい。

【解答例】

まず、3 次元コンピュータグラフィックスの統合ソフトウェアにあらかじめ用意されたプリミティブの球をシーンに追加する。つぎに、球の色と質感を変更するために、球にマテリアルを追加する。マテリアルの拡散反射光 (diffuse) の色を青 (R=0, G=0, B=1) に近いものに変更すれば、青い球が作成できる。金属の質感を得るためには、鏡面反射光によるハイライトの強さと色が適切である必要がある。ハイライトの色が白であると、金属の質感は得られない。そのため、表面の「粗さ」や「メタリック」などのマテリアルのパラメータを調節することで、ハイライトの色を青に近い色に変更し、球の質感を金属に近づける。また、多面体 (ポリゴンの集まり) で作成した球面を滑らかに表示するために、サブディビジョンサーフェスによるポリゴンの分割とスムーズシェーディングの設定を行う。球は、多面体の代わりに NURBS 曲面を使用して作成することもできる。質感は、マテリアルのプレビュー表示を行ったり、ライトを設定した上でレンダリングしたりすることで確認する。

(具体的な統合ソフトウェアの使用例を用いた説明も可)

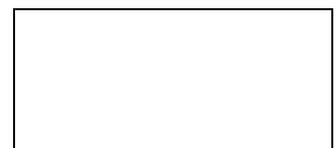
- B) 3 次元コンピュータグラフィックスと写真撮影における「フレーミング」と「カメラアングル」を説明しなさい。

【解答例】

3 次元コンピュータグラフィックスと写真撮影のどちらの場合も、被写体を画像内に収めるためにカメラの位置・向きと画角 (視野角) を設定する。「フレーミング」とは、仮想世界または現実世界のどこをどのように切り取るかを定める作業である。「フレーミング」においては、被写体を画面の中央に置くセンター配置、画面を分割した交点に置く三分割法、遠目から被写体を収めるロングショット、被写体を画面いっぱいに収めるフルショットなどが使い分けられる。被写体を大きく写すためには、カメラの画角を狭くするズームインか、被写体に近づくドリーインを用いる。「カメラアングル」は、カメラ (視点) と被写体 (注視点) の位置関係を表すカメラワークの用語である。注視点に対して視点を高く設定する場合をハイアングル、低く設定する場合をローアングルと呼び、目的に応じて使い分ける。アイレベルは、人間の目の高さに設定され、注視点が視点に近い高さにある場合であり、多くの状況で使用される。コンピュータグラフィックスにおいては、カメラの位置・角度と被写体の位置情報によっても、カメラアングルを確認できる。

受験科目名
コンピュータ グラフィックス

{ 1/4 頁}



【問題 2】

A) 3次元コンピュータグラフィックスにおけるレンダリングの処理の流れと手順および目的を説明しなさい。

【解答例】

3次元コンピュータグラフィックスにおけるレンダリングとは、仮想空間内の3次元モデルとライトやカメラなどの情報をもとに2次元画像を生成するための計算である。レンダリングの処理には、投影変換、クリッピング、隠面消去、シェーディングの各段階がある。投影変換の目的は、3次元モデル内の点（頂点など）が投影面（スクリーン）に投影された点の2次元座標値を得ることであり、この計算は4×4行列を用いて行われる。クリッピングとは、ビューボリュームの範囲外にある点の除外を目的とした処理であり、これによって投影変換を行う点の数が減り、計算量が削減できる。ビューボリューム内にあるオブジェクトであっても、手前にあるオブジェクトによって隠される見えない部分を除外すれば、レンダリングの計算量は少なくて済む。これを隠面消去という。シェーディングは、仮想世界の見えている部分を画像上に描画するための処理である。投影変換で計算された三角形の各頂点の2次元座標値を用いれば、その三角形を画像内に描くことができる。三角形全体を一定の色で表示する方法は、コンスタントシェーディング（フラットシェーディング）と呼ばれ、短時間で描画が行える。滑らかな曲面を描画する場合は、明るさや法線ベクトルを補間し、ポリゴン内の各画素の色を計算するスムーズシェーディングが用いられる。

B) 線形変換の計算方法と3次元コンピュータグラフィックスにおける応用例を説明しなさい。

【解答例】

線形変換は、一次変換とも呼ばれ、あるベクトルを別のベクトルに移す変換（写像）である。n次元ベクトルの線形変換は、n×n行列とベクトルの積によって計算できる。例えば、3×3行列を3次元列ベクトルに左からかけることで、別の3次元ベクトルが得られる。3次元コンピュータグラフィックスにおいては、ポリゴン（多角形）の頂点の座標値を表す位置ベクトルと行列の積によって、新しい頂点の座標値を計算し、ポリゴンの回転、鏡映、スケーリング（拡大・縮小）などが行える。具体的な計算式の例を以下に示す。線形変換の行列の逆行列は、もとの形状に戻す線形変換、行列式の絶対値は、変換前後の立体の体積比率をそれぞれ表している。

$$\text{スケーリング} \quad \begin{bmatrix} a & 0 & 0 \\ 0 & b & 0 \\ 0 & 0 & c \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} ax \\ by \\ cz \end{bmatrix} \quad \text{XY平面に関する鏡映} \quad \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} -x \\ y \\ z \end{bmatrix}$$

$$\text{X軸まわりの回転} \quad \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \theta & -\sin \theta \\ 0 & \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} x \\ y \sin \theta + z \cos \theta \\ -y \sin \theta + z \cos \theta \end{bmatrix} \quad \text{XY平面への平行投影} \quad \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} x \\ y \\ 0 \end{bmatrix}$$

複数の線形変換を続けて行う計算は、3×3行列同士の積を計算することで、1回の線形変換で行うことができる。このような計算を行うと、多くの頂点を持つ3Dモデルの線形変換（スケーリングと回転の合成変換など）が効率よく行える。

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \theta & -\sin \theta \\ 0 & \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} a & 0 & 0 \\ 0 & b & 0 \\ 0 & 0 & c \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} a & 0 & 0 \\ 0 & b \cos \theta & -c \sin \theta \\ 0 & b \sin \theta & c \cos \theta \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

受験科目名
コンピュータ グラフィックス



問題 3.

設計したシステムのユーザビリティを評価するために、どのような手法を用いるべきか代表的な手法を3つ挙げて説明し、それぞれの手法の利点と欠点を述べよ。

チェックリストによる評価

インタフェースの仕様をインタフェースの設計原則や設計ガイドライン、チェックリストのいずれかと照合し、設計原則にのっとっているか、所定の基準を達成しているかどうかを調べる

利点: リストさえあればコストがかからない方補で手軽にできる

欠点: 評価者によって偏りが生じること、作業が機械的になりやすく問題点を見落としやすいこと、チェックリストにのってないことは評価されないこと

インスペクションによる評価

仕様書を基に評価する方法であるが、複数の専門家が集まり、紙に書かれた画面例や簡単なモックアップを使いながら使い方を想定して問題点を見つける技法。ヒューリスティック法とウォークスルー法があり、2つの技法の違いは問題点を発見するアプローチにある。

利点: 実際のインタフェースが無くても問題点を洗い出すことができ、コストがかからない、設計段階で評価できる

欠点: 評価者にはインタフェースに関する深い知識と経験が求められ、初心者にはなかなか問題点がみつけれられない。

ユーザビリティテスト

製品やサービス (Web サイトやアプリなど) を、実際のユーザーに使ってもらい、その使いやすさ (ユーザビリティ) を評価・検証するテスト

利点: 実際のユーザーがアプリケーションを使用する際の具体的な問題点を発見できる。

欠点: 実施には時間とコストがかかる。

受験科目名

コンピュータ グラフィックス

{ 3 / 4 頁 }



問題 4.

Web デザインにおけるレスポンシブデザインとは何かを説明し、レスポンシブデザインを実現するための具体的な方法を 3 つ挙げよ。

レスポンシブデザインは、デバイスの画面サイズに応じてウェブサイトのレイアウトを柔軟に変える設計手法。ユーザーがどのデバイス（PC、タブレット、スマートフォンなど）を使用しても、快適に閲覧できるようにすることを目指す。

具体的な方法:

フルイドグリッドレイアウト: ピクセルベースではなく、相対的な単位（%など）を使用して要素の幅を指定する。

メディアクエリ: CSS のメディアクエリを使用して、特定の条件（画面の幅など）に応じてスタイルを変更する。

フレキシブルメディア: 画像や動画などのメディア要素を、コンテナの幅に応じて自動的にサイズを変更するように設定する。

受験科目名
コンピュータ グラフィックス

