知能システム学Ⅰ

次の3問すべてを解答せよ．問題毎に別の解答用紙を用いること．解答用紙の追加は認めない．

1

次の実数値関数 $y = y(t)$ に関する微分方程式を考える．ただし，$y_0 > 0$, $k \geq 0$ とする．以下の問い 1) ～ 3) に答えよ．

$$\frac{dy}{dt} = ky, \quad y(0) = y_0$$

1) 解を求めよ．ただし，k は定数とする．また，$y(T) = 2y_0$ となる時間 T を求めよ．

2) k が図1のように変化するとき，解を求めよ．また，$\lim_{t\to\infty} y(t)$ を求めよ．ただし，k'，L は定数とし，$k' > 0$，$L > 0$ とする．

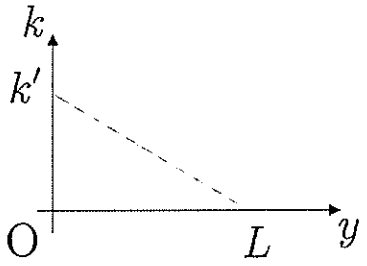

図1

3) 問い2) と同じ k を用いるとき，次の3つの条件に対応した，$y - t$ グラフの概形を描け．ただし，$L = 10$, $k' = 1$ とし，y 軸との交点の値を明記せよ．

a) $y_0 = 5$

b) $y_0 = 10$

c) $y_0 = 20$

2

以下の問い 1)〜6) に答えよ．ただし，$\boldsymbol{A}$，$\boldsymbol{B}$，$\boldsymbol{C}$ は $n \times n$ の実正方行列である．また，答えだけでなくそれぞれの導出過程も示すこと．

1) $\boldsymbol{A}$ の固有多項式を

$$\phi_A(t) = t^n + a_{n-1}t^{n-1} + \cdots + a_1 t + a_0, \qquad t,\ a_0, \cdots, a_{n-1} \in \mathbb{R}$$

とし，重複を含めた全ての固有値を λ_k $(k = 1, 2, \cdots, n)$ とする．係数 a_{n-1} と a_0 を λ_k を用いて表せ．

2) 1) の行列 $\boldsymbol{A}$ が正則行列であるための必要十分条件を，1) の結果を用いて表せ．

3) $\boldsymbol{B}$ と $\boldsymbol{C}$ が相似であり，1) の行列 $\boldsymbol{A}$ と $\boldsymbol{B}$ も相似であれば，$\boldsymbol{A}$ と $\boldsymbol{C}$ も相似であることを示せ．また，$\boldsymbol{C}$ のトレースと行列式を λ_k を用いて表せ．ただし，$\boldsymbol{A}$ と $\boldsymbol{B}$ が相似であるとは，$\boldsymbol{B} = \boldsymbol{P}^{-1}\boldsymbol{A}\boldsymbol{P}$ なる $n \times n$ の正則行列 $\boldsymbol{P}$ が存在することである．

4) $\mathbb{R}^3$ 上の線形変換

$$f\left(\begin{bmatrix} x \\ y \\ z \end{bmatrix}\right) = \begin{bmatrix} x - z \\ 2x + y + z \\ y + 3z \end{bmatrix}$$

を考える．この線形変換の，基底

$$\left\{ \boldsymbol{u}_1 = \begin{bmatrix} 1 \\ 1 \\ 2 \end{bmatrix}, \boldsymbol{u}_2 = \begin{bmatrix} -1 \\ 2 \\ 0 \end{bmatrix}, \boldsymbol{u}_3 = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix} \right\}$$

に関する表現行列 $\boldsymbol{F}$ を求めよ．ただし，

$$[f(\boldsymbol{u}_1)\ f(\boldsymbol{u}_2)\ f(\boldsymbol{u}_3)] = [\boldsymbol{u}_1\ \boldsymbol{u}_2\ \boldsymbol{u}_3]\boldsymbol{F}$$

を満たす $\boldsymbol{F}$ を，線形変換 f の基底 $[\boldsymbol{u}_1\ \boldsymbol{u}_2\ \boldsymbol{u}_3]$ に関する表現行列と呼ぶ．

5) 3×3 の正則行列 $\boldsymbol{Q}$ によって，4) の基底を $[\boldsymbol{u}_1\ \boldsymbol{u}_2\ \boldsymbol{u}_3]\boldsymbol{Q}$ のように取り替える．線形変換 f の，基底 $[\boldsymbol{u}_1\ \boldsymbol{u}_2\ \boldsymbol{u}_3]\boldsymbol{Q}$ に関する表現行列 $\boldsymbol{G}$ を，$\boldsymbol{Q}$ と $\boldsymbol{F}$ を用いて表せ．

6) 5) で求めた表現行列 $\boldsymbol{G}$ のトレースと行列式を求めよ．

3

以下の問い 1)，2)に答えよ．

1) プログラム１は基数ソートを行うC言語プログラムである．基数ソートのソート対象の配列はaradallで定義する．関数 counting_sort は引数 atemp のある基数に対し分布数え上げソートを行う．当該関数の引数 a は分布数え上げの対象配列，b は結果配列，c は累積度数の計算用の配列である．このプログラムに関連する以下の小問 a)〜c)に答えよ．

a) 基数ソートとクイックソートの平均時間計算量と平均領域計算量をそれぞれ要素数 n のオーダー表記で表せ．ただし，n は十分大きいものとする．
b) 8 行目では基数配列の累積度数を計算している．空欄［　1　］を埋めよ．
c) 10 行目と 11 行目では累積度数に従ってデータを配列 atemp から配列 b にコピーする作業を行っている．空欄［　2　］を埋めよ．

プログラム１

```
1  #include<stdio.h>
2  #define NUM 10
3  /* 関数のプロトタイプ宣言は省略している */
4  void counting_sort(int a[], int b[], int c[], int atemp[]) {
5      int i,j;
6      for (i=0; i<NUM; i++) c[i] = 0;
7      for (j=0; j<NUM; j++) c[a[j]] += 1;
8      for (i=1; i<NUM; i++) c[i] = [      1      ]
9      for (j=NUM-1; j>=0; j--) {
10         b[c[a[j]]-1] = atemp[j];
11         [      2      ]
12     }
13 }
14 int main() {
15     int i,j;
16     int aradall[NUM] = {56,42,333,60,1,71,92,235,53,19};
17     int aradcrt[NUM], b[NUM], c[NUM];
18     for (i=0; i<NUM; i++) aradcrt[i] = aradall[i]%10;
19     counting_sort(aradcrt, b, c, aradall);
20     for (i=0; i<NUM; i++) aradall[i] = b[i];
21     for (i=0; i<NUM; i++) aradcrt [i] = aradall[i]/10%10;
22     counting_sort(aradcrt, b, c, aradall);
23     for (i=0; i<NUM; i++) aradall[i] = b[i];
24     for (i=0; i<NUM; i++) aradcrt [i] = aradall[i]/100%10;
25     counting_sort(aradcrt, b, c, aradall);
26     for (i=0; i<NUM; i++) printf("%d ",b[i]);
27     printf("\n");
28 }
```

---次ページに続く---

---問題３の続き---

2) プログラム２は構造体 node によって定義された二分探索木の挿入・削除作業を行うＣ言語プログラムである．このプログラムについて，以下の小問 a), b)に答えよ．

a) 32～37 行目の deletemin 関数は二分探索木のノードを削除する際，ノードの順序を調整する関数である．関数の中の空欄 [A] を埋めよ．ただし，複数の文になってもよい．

b) 59～64 行目の関数 inorder は再帰を使用して二分探索木を通りがけ順でなぞる．当該関数を呼びだしている 69 行目と 72 行目を実行したときの標準出力結果を示せ．

プログラム２

```
1  #include<stdlib.h>
2  #include<stdio.h>
3  typedef int KEY;
4  typedef struct node {
5     KEY data;
6     struct node *left, *right;
7  } NODE;
8  /* 関数の宣言は省略している */
9  int keyequal(KEY key1, KEY key2) {
10    if(key1==key2) return 1;
11    else return 0;
12 }
13 int keylt(KEY key1, KEY key2) {
14    if(key1<key2) return 1;
15    else return 0;
16 }
17 NODE *insert(NODE **p, KEY key) {
18    NODE *new;
19    while (*p != NULL) {
20        if (keyequal(key, (*p)->data)) return NULL;
21        else if (keylt(key, (*p)->data)) p = &(*p)->left;
22        else p = &(*p)->right;
23    }
24    if((new = malloc(sizeof(NODE))) == NULL)
25        printf("Out of memory!!");
26    new->left  = NULL;
27    new->right = NULL;
28    new->data = key;
29    *p = new;
30    return new;
31 }
32 NODE *deletemin(NODE **p) {
33    NODE *x;
34    while((*p)->left != NULL) p = &(*p)->left;
35       A
36    return x;
37 }
38 int delete(NODE **p, KEY key) {
39    NODE *x;
40    while(*p!=NULL) {
41       if(keyequal(key, (*p)->data)) {
42             x = *p;
43             if(x->left == NULL && x->right == NULL) *p = NULL;
44             else if (x->left == NULL) *p = x->right;
45             else if (x->right == NULL) *p = x->left;
46             else {
47                *p = deletemin(&x->right);
48               (*p)->right = x->right;
```

---次ページに続く---

---問題３の続き---

```
49                  (*p)->left = x->left;
50              }
51              free(x);
52              return 1;
53          }
54          else if(keylt(key, (*p)->data)) p = &(*p)->left;
55          else p = &(*p)->right;
56      }
57      Return 0;
58  }
59  void inorder(NODE *p) {
60      if(p==NULL) return;
61      inorder(p->left);
62      printf("%d ", p->data);
63      inorder(p->right);
64  }
65  int main() {
66      NODE *root;
67      int i=0;
68      for(i=0; i<10; i++) insert(&root, i);
69      inorder(root);
70      printf("\n");
71      delete(&root, 3);
72      inorder(root);
73      printf("\n");
74      return 0;
75  }
```