平成22年度大学院前期課程入試問題

選択科目　b. システム制御

平成21年8月25日

注意事項

- 問題用紙は全部で９枚（但し，表紙を除く）あるので確認すること．
- 解答には必ず問題番号を書き，どの問題に解答したか分かるようにすること．
- 「制御工学」（問題１および２－１，２－２）は全員が解答せよ．
- 選択問題（問題３～７）から２分野を選択して解答せよ．選択しなかった問題の解答用紙には×印を記して選択した問題が明確に分かるようにせよ．
- 選択問題から３分野以上解答した場合には，選択問題の解答をすべて無効とするので注意せよ．
- 「電気機器」を選択する者は，問題３－１および３－２を解答せよ．
- 「パワーエレクトロニクス」を選択する者は，問題４－１および４－２を解答せよ．
- 「信号処理」を選択する者は，問題５－１および５－２を解答せよ．
- 「論理回路・計算機システム」を選択する者は，問題６を解答せよ．
- 「基本アルゴリズム・プログラミング」を選択する者は，問題７－１および７－２を解答せよ．
- 解答用紙は色分けしてあるので，問題番号と対応させて以下のように使い分けよ（間違わないように注意せよ）．

問題番号		解答用紙の色
1	…………	白
2	…………	赤
3	…………	青（紺）
4	…………	黄
5	…………	水（薄い青）
6	…………	桃
7	…………	緑

- 解答用紙の表に書き切れない場合は，裏を使用しても良い．
- 問題用紙は持ち帰っても良い．

制 御 工 学

1．図1のフィードバックシステムについて以下の問いに答えよ．ただし，$K_1, K_2, K_3, \alpha, \beta$ はいずれも非負の実数とする．

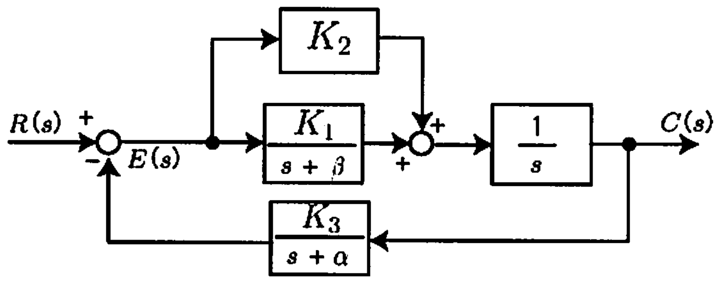


図1

(i) 入力 $R(s)$ から出力 $C(s)$ までの伝達関数を答えよ．

(ii) フィードバックシステムが安定となる $K_1, K_2, K_3, \alpha, \beta$ の条件を答えよ．

(iii) $K_1 = 7,\ K_2 = 5,\ K_3 = 4,\ \alpha = 5,\ \beta = 1$ の場合の，フィードバックシステムのボード線図（ゲイン特性，位相特性）の概形を示せ．なお，ゲイン特性は折点周波数を明示した折れ線で近似せよ．また，位相特性は $\omega \to +0$ と $\omega \to +\infty$ のときの漸近値を明示せよ．

(iv) 問 (iii) と同じく，$K_1 = 7,\ K_2 = 5,\ K_3 = 4,\ \alpha = 5,\ \beta = 1$ の場合の，フィードバックシステムのベクトル軌跡の概形を示せ．特に $\omega \to +0$ と $\omega \to +\infty$ のときの形状に注意して描け．

(v) $K_1 = 5,\ K_2 = 0,\ K_3 = 1,\ \alpha = 3,\ \beta = 1$ の場合の，ゲイン余裕を求めよ．なお，次の近似を用いてもよい．$\log 2 \fallingdotseq 0.30$, $\log 3 \fallingdotseq 0.48$.

(vi) 問 (v) と同じく，$K_1 = 5,\ K_2 = 0,\ K_3 = 1,\ \alpha = 3,\ \beta = 1$ の場合の，正弦波入力 $r(t) = \cos t$ に対し，時間が十分に経過した後の出力 $c(t)$ を示せ．

(vii) 問 (v) と同じく，$K_1 = 5,\ K_2 = 0,\ K_3 = 1,\ \alpha = 3,\ \beta = 1$ の場合の，単位ランプ入力 $(r(t) = t, t \geq 0)$ に対する定常偏差 $e(\infty)$ を答えよ．

2―1 状態方程式と出力方程式がそれぞれ

$$\frac{d\boldsymbol{x}(t)}{dt} = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{b}u(t)$$

$$y(t) = \boldsymbol{c}\boldsymbol{x}(t)$$

で表現される1入力1出力システムに対して，以下の問いに答えよ．ただし，

$$\boldsymbol{x}(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix}, \quad \boldsymbol{A} = \begin{bmatrix} -2 & 0 & a \\ 1 & 0 & 1 \\ 0 & 0 & 2 \end{bmatrix}, \quad \boldsymbol{b} = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}, \quad \boldsymbol{c} = \begin{bmatrix} 1 & 0 & 1 \end{bmatrix}$$

であり，a は実数値をとるパラメータとする．

(i) 入力 $u(t)$ から出力 $y(t)$ への伝達関数を求めよ．

(ii) このシステムが不可制御になるような a の値をすべて求めよ．

(iii) 任意の a の値に対して，このシステムが不可観測となることを示せ．

(iv) 入力 $u(t)$ を $u(t) = -k_1x_1(t) - k_2x_2(t) - k_3x_3(t)$ とする状態フィードバック制御を考える．パラメータ a の値が $a = 1$ のとき，状態フィードバックを施した閉ループシステムの特性方程式の根が -2, $-1 \pm j2$ となるように，フィードバック係数 k_1, k_2, k_3 の値を決定せよ．ただし，j は虚数単位を表す．

2―2 状態方程式と出力方程式がそれぞれ

$$\frac{d\boldsymbol{x}(t)}{dt} = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{b}u(t)$$

$$y(t) = \boldsymbol{c}\boldsymbol{x}(t)$$

で表現される1入力1出力システムに対して，以下の問いに答えよ．ただし，

$$\boldsymbol{x}(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}, \quad \boldsymbol{A} = \begin{bmatrix} 1 & 1 \\ 0 & 2 \end{bmatrix}, \quad \boldsymbol{b} = \begin{bmatrix} -1 \\ 1 \end{bmatrix}, \quad \boldsymbol{c} = \begin{bmatrix} 1 & 1 \end{bmatrix}$$

である．

(i) 行列指数関数 $e^{\boldsymbol{A}t}$ を求めよ．

(ii) $t = 0$ での初期状態を

$$\boldsymbol{x}(0) = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$$

とする．このとき，入力 $u(t) = e^{-t}$ $(t \geq 0)$ に対する出力 $y(t)$ $(t \geq 0)$ を求めよ．

電気機器

3－1　定格電圧 200 V，定格出力 1.7 kW，定格周波数 60 Hz の三相巻線形誘導電動機があり，無負荷で運転したところ，回転数が 1800 rpm で回り続けたものとする（機械損等の無い理想状態を仮定する）．次に，この電動機を全負荷で運転したところ，回転数は 1710 rpm となり，力率は 85%，一次電流は 7.0 A となった．ただし，二次回路は Y 結線されており，各相の巻線抵抗は 0.2Ω とする．以下の問いに答えよ．

(i) 固定子の極数を求めよ．

(ii) 全負荷時におけるすべりを求めよ．

(iii) 全負荷時における二次誘導起電力の周波数を求めよ．

(iv) 全負荷時における電動機の効率を求めよ．ただし，平方根を開く必要はない．

(v) 回転速度を 1440 rpm として定格トルクを得るためには，二次側に何Ωの抵抗を追加すればよいか．

3－2　非突極型（円筒型）三相同期発電機の一相分の等価回路を図 3 に示す．ただし，x_s を同期リアクタンス（電機子反作用リアクタンスと電機子漏れリアクタンスの和），r_a を電機子巻線抵抗，$\dot{E}_0$ を公称誘導起電力，$\dot{V}$ を発電機の端子電圧（相電圧），$\dot{I}$ を電機子電流とする．以下の問いに答えよ．

(i) 負荷に力率 $\cos\theta$ の遅れ負荷を接続したときのベクトル図を描け．ただし，$\dot{E}_0, \dot{V}, \dot{I}$ の関係が分かるように描き，力率角 θ と内部相差角 δ も図示せよ．

(ii) 問(i) で求めたベクトル図に基づき公称誘導起電力の大きさ E_0 を求めよ．ただし，端子電圧の大きさ V，電機子電流の大きさ I, x_s, r_a, θ を用いて表すこと．

(iii) 負荷に供給される有効電力（三相分）を，E_0, V, x_s, δ を用いて表せ．ただし，$r_a \ll x_s$ として r_a を無視するものとする．また，この有効電力が最大となる δ を求めよ．

(iv) 同期発電機の電機子巻線に三相交流電圧を印加すると電動機としても動作させることができる．この同期電動機を進み力率で動作させたとき，$\dot{E}_0, \dot{V}, \dot{I}, x_s, r_a, \theta, \delta$ を用いてベクトル図を描け．

(v) 全負荷および無負荷時における同期電動機の V 曲線を描け．ただし，遅れ力率となる領域および進み力率となる領域をそれぞれ示すこと．

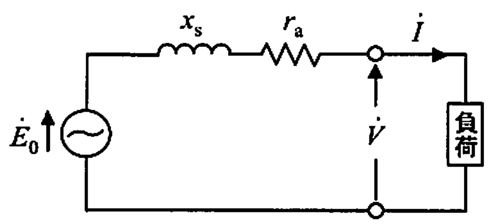


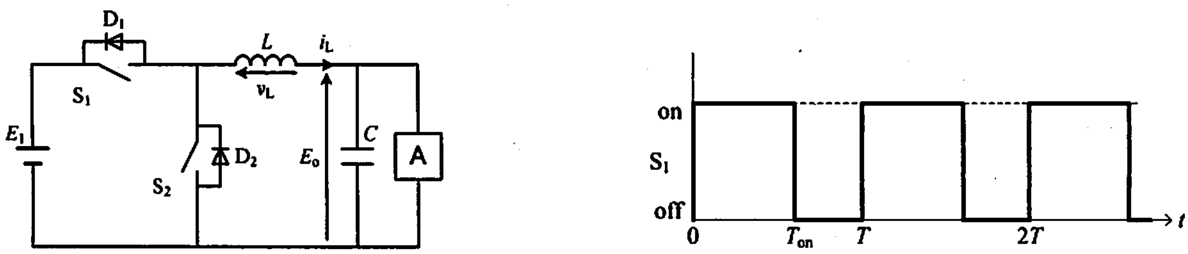
図 3

パワーエレクトロニクス

4－1　図4－1(a)に示す回路について考える．ただし，E_1は定電圧源とみなせるバッテリを示し，S_1，S_2は主スイッチ，D_1，D_2はダイオードを示す．各素子は理想的に動作するものとする．また，キャパシタCは出力電圧E_oの脈動が無視できるほど十分に大きく，回路動作はすべて周期定常状態にあるものとする（周期T）．以下の問いに答えよ．

(i) $\boxed{\text{A}}$の部分に純抵抗Rを接続した．S_2はオフの状態を保つものとする．S_1に図4－1(b)に示す動作をさせたとき，回路は電流連続モード動作となった．このとき，S_1がオンしている時間T_{on}と周期Tの比をデューティ比d($d = T_{on}/T$)として定義する．入力電圧E_1とE_oの関係を，dを用いて示すとともにその導出過程も示せ．また，インダクタ電圧v_Lとインダクタ電流i_Lの概形を描き，i_Lのリプル電流Δi_L(振動成分の振幅の1/2の値）を求めよ．またΔi_Lの最大値とそのときのdを求めよ．

(ii) $\boxed{\text{A}}$の部分にE_1と同極性の電圧源E_2を接続した（ただし，$E_2 < E_1$）．E_2からバッテリE_1を充電したい．どのようにスイッチング素子を動作させればよいか，簡単に述べよ．



(a)　　(b)

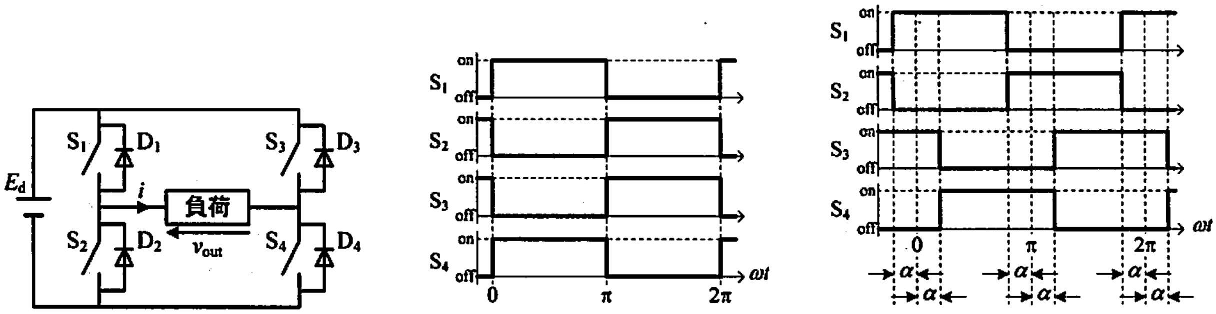
図4－1

4－2　図4－2(a)に示す回路について考える．E_dは直流電圧，S_1～S_4は主スイッチ，D_1～D_4はダイオードを示している．ただし，回路は周期定常状態にあるものとし（周期2π，角周波数ω），各素子は理想的に動作するものとする．以下の問いに答えよ．

(i) スイッチング素子S_1～S_4を，それぞれ図4－2(b)に示すように動作させた．負荷が誘導性（抵抗R，インダクタンスLの直列回路）である場合，負荷に印加される電圧v_{out}と負荷を流れる電流iの概形を描け．また，各時点で電流が流れているスイッチング素子の記号（S_1，D_1など）を図に併記せよ．

(ii) 問(i)と同じ動作状態において，位相$\omega t = 0$のときの電流値$i(0)$と半周期後の位相$\omega t = \pi$のときの電流値$i(\pi)$の関係を示し，$i(0)$の値を求めよ．

(iii) スイッチング素子S_1～S_4に，図4－2(c)に示す動作をさせた場合のv_{out}の概形を描き，その基本波の振幅V_1をE_dと位相αで表わせ．



(a)　　(b)　　(c)

図4－2

信号処理

5－1　離散時間信号に関する以下の問いに答えよ．

(i)　離散時間信号 $x[n]=\delta[n]$ について，N 点の離散フーリエ変換 DFT を求めよ．ただし $\delta[n]$ は離散時間の単位インパルス信号（クロネッカのデルタ）とする．

(ii)　離散時間信号 $x[n]=1$ ($n=0, 1, 2, \cdots, N-1$) について，N 点の離散フーリエ変換 DFT を求めよ．

(iii)　N 点の離散フーリエ変換 DFT を高速に解くアルゴリズムとして，高速フーリエ変換 FFT がある。FFT を用いて逆離散フーリエ変換 IDFT を求めるアルゴリズムを示せ。

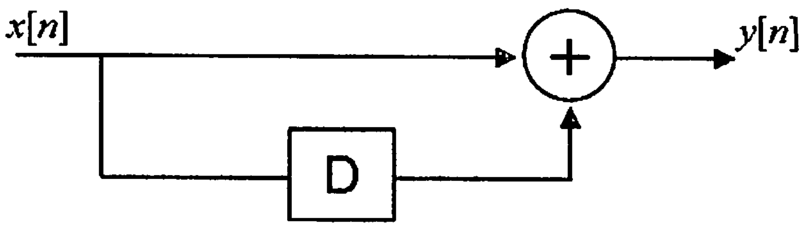
5－2　下図に示す離散時間線形時不変システムについて，以下の問いに答えよ．ただし，図中の $x[n]$，$y[n]$は，それぞれ入力信号および出力信号，D は単位時間の遅延素子である．

(i)　システムのインパルス応答 $h[n]$を求めよ．

(ii)　システムの周波数応答 $H(\Omega)$を求めよ．ここで，離散時間信号の周波数応答は，角周波数 Ω（Ωは$|\Omega|\leqq\pi$ の実数）の連続関数として $H(\Omega)=\sum_{n=-\infty}^{\infty}h[n]\,e^{-jn\Omega}$で定義される．

(iii)　システムの振幅特性（ゲイン）$|H(\Omega)|$と位相特性$\angle H(\Omega)$をそれぞれ求め，図示せよ．

(iv)　システムの振幅特性が最大値から 3dB 以内である帯域幅（3dB 帯域幅）を求めよ．



論理回路・計算機システム

6．10進数の各桁の数字を4桁の2進数で表すコードを2進化10進コード(BCDコード)という．一般に4桁の2進数は，0から15までの整数を表すことができるが，BCDコードではこのうちの最初の10個（0〜9）を有効な数値として扱う．例えば，10進数の11は，BCDコードでは0001 0001となる．以下では，BCDコードにより表現された2つの1桁の10進数 $X=(x_3x_2x_1x_0)$，$Y=(y_3y_2y_1y_0)$ を，下位からの桁上げ C_d^l を考慮して加算し，和 $Z=(z_3z_2z_1z_0)$ とその桁上げ C_d^u を出力する回路について考える．ただし，以下すべての4桁の2進数及びBCDコードにおいて，添字が小さい方を下位桁とする．X と Y の10進数の加算では，4桁の2進数の加算結果を，10進数の桁上げのために修正する必要がある．例えば，10進数の5と6の加算においては，$X=(0101)$ と $Y=(0110)$ を4桁の2進数として加算した結果である1011に対し，和 $Z=(0001)$ と桁上げ $C_d^u=1$ が出力されるような修正が必要となる．このようなBCDコードの加算を図1に示す回路によって実現するとき，以下の問いに答えよ．

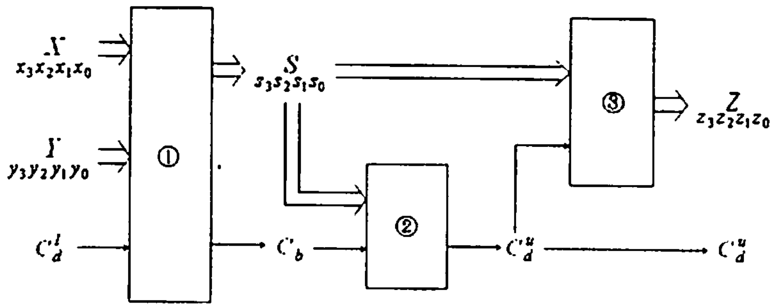


図1

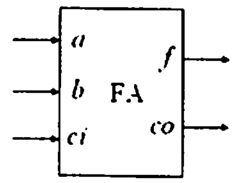


図2

(i) 下位からの桁上げを考慮した1桁の2進数の加算を行う回路は全加算器(FA)と呼ばれ，図2に示すように，1桁の2進数 a，b および下位からの桁上げ ci から，和 f と上位への桁上げ co が求められる．全加算器の真理値表を示すと共に，f と co を a，b，ci の最小積和形論理式で表せ．

(ii) 図1中①に示すように，$X=(x_3x_2x_1x_0)$，$Y=(y_3y_2y_1y_0)$ および下位からの桁上げ C_d^l を入力とし，順次桁上げ方式の4桁の2進数の加算を行う回路を，全加算器(FA)を複数個用いて構成し，その回路図を示せ．ただし，出力は和 $S=(s_3s_2s_1s_0)$，桁上げ C_b とし，全加算器(FA)は図2に示す記号で表すものとする．なお，解となる複数の回路構成が存在する場合はその1つを示せばよい．

(iii) 図1中②では，$S=(s_3s_2s_1s_0)$，C_b に基づき，10進数の加算により生じる桁上げを検出する．桁上げ C_d^u の論理式を，$S=(s_3s_2s_1s_0)$，C_b の最小積和形論理式で表し，その回路図を示せ．ただし，利用可能な論理ゲートは論理積(AND)，論理和(OR)，論理否定(NOT)とし，各ゲートの入力数は3以下とする．

(iv) $S=(s_3s_2s_1s_0)$ と C_d^u から $Z=(z_3z_2z_1z_0)$ を得るためには，S にどのような修正が必要となるか説明せよ．また，図1中③に示すように，$S=(s_3s_2s_1s_0)$ と C_d^u を入力としてそのような修正を実現し，$Z=(z_3z_2z_1z_0)$ を出力する回路を，全加算器FAを複数個用いて構成し，図示せよ．なお，全加算器は図2に示す記号で表し，解となる複数の回路構成が存在する場合はその1つを示せばよい．

基本アルゴリズム・プログラミング

7－1 バケットソートは，整列したいデータの取りうる値が m 種類であるとき，m 個のバケットを準備し，各値ごとに1個のバケットを対応づけることでソートを行う方法である．プログラムAは，取りうる値が $0,1,\ldots,7$ であるデータに対し，バケットソートを行うプログラムである．以下の問いに答えよ．

(i) プログラムAの空欄 $\boxed{\ (1)\ }$, $\boxed{\ (2)\ }$ を埋めよ．

(ii) プログラムAの /* α */ において j=3 のときの配列 a に格納されている値を示せ．

(iii) プログラムAで表されるバケットソートの時間計算量を求めよ．ただし，ソートするデータ数を n，必要なバケット数を m として，m, n に対するオーダとして示し，その導出の理由を示せ．さらに，バケットソートの利点および欠点を，適用可能なデータの種類や，時間計算量および記憶容量の観点から述べよ．

プログラムA

```
#define M 8          /* バケット数 */
#define N 10         /* データ数 */

int main(){
  int a[N]={2,3,5,6,7,4,2,0,7,1};                 /* ソート対象の配列 */
  int bucket[M], i, j, k;                      /* ソートに用いる bucket */

  for (j=0; j<M; j++) bucket[j] = 0;               /* bucket を初期化 */
  for (i=0; i<N; i++) bucket[ (1) ]++;             /* bucket に格納 */

  i=0;
  for (j=0; j<M; j++){                     /* bucket のデータを結合 */
    for (k= (2) ; k!=0; k--){
      a[i]=j;
      i++;
    }
    /* α */
  }
  return 0;
}
```

7－2　プログラムBは，クラスカルのアルゴリズムを用いて，与えられたグラフの最小木を求めるプログラムである．クラスカルのアルゴリズムは，最短の枝から順にその枝が最小木を構成するかどうかを調べる方法である．このプログラムでは，配列 a[ ] に枝長の短い順にソートされた各枝の情報が格納されている．また，配列 set[ ] は各節点が属する節点集合の番号を格納しており，２つの節点集合がマージされたときは，小さい方の番号がマージ後の集合につけられる．

図 7(a) は，与えられたグラフを示しており，円内の数字は各節点番号を，枝に付された数字は，その枝の長さを表す．図 7(b), (c) は，それぞれ，初期状態と反復 ite=1 が終了したときの「節点集合と選択された枝」を示したものである．点線で囲まれた部分は，節点集合を表し，太字の数字はその集合の番号を表す．このプログラムに関して以下の問いに答えよ．

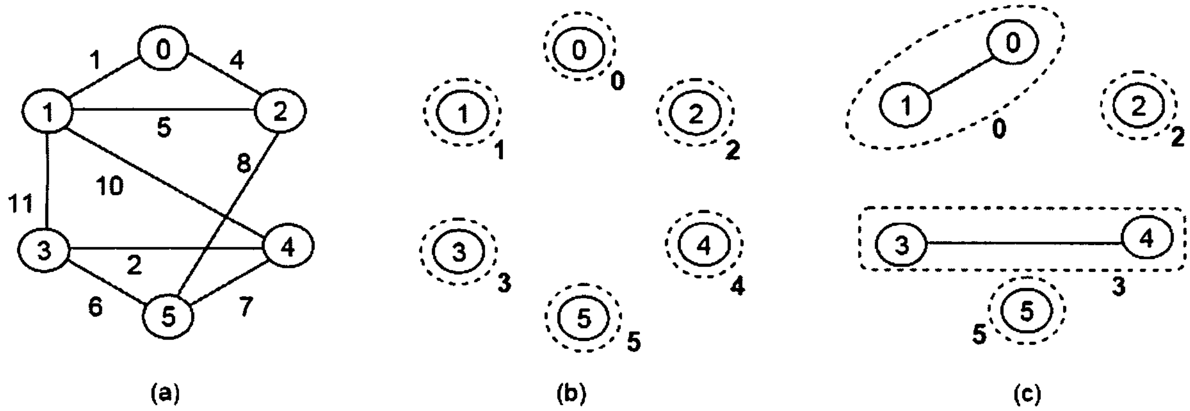


図 7

(i) クラスカルのアルゴリズムのように，局所的に最適な選択を続けることにより解を得る方法は一般的に何と呼ばれるか答えよ．

(ii) クラスカルのアルゴリズムでは，最短の枝から順にその枝が最小木の構成要素になりうるかを調べていく．ある枝が最小木の構成要素になりえない場合，その枝とすでに最小木の要素として選択されている枝集合の関係がどのようになっているか答えよ．

(iii) プログラムBの空欄 $\boxed{(3)}$ ～ $\boxed{(6)}$ を埋めよ．

(iv) プログラムBの ite=2 以降の各反復において，/* β */ での「節点集合と選択された枝」の状態を図 7(c) を参考にして図示せよ．

(v) 与えられたグラフの節点数を n，枝数を m としたとき，n および m に対するプログラムBの時間計算量のオーダを求め，その計算量の算出根拠を述べよ．

プログラムB

```
#define Node_num 6
#define Arc_num  9

typedef struct arc_tag { /* 枝情報を格納する構造体 */
   int length;           /* 枝長 */
   int end_pt1;          /* 枝の端点 */
   int end_pt2;          /* 枝の端点 */
   int selected;         /* 1: 枝が最小木に含まれる -1: 含まれない */
} Arc;

void swap(int *a, int *b){ int tmp; tmp=*a; *a=*b; *b=tmp; }
void merge_trees(int n1, int n2, int *set){      /* 2つの節点集合をマージ */
  int k;
  if(set[n2] > set[n1]) swap(&n1, &n2);
  for(k=0; k<Node_num; k++)
    if(set[k] == set[n1]) [ (3) ] = [ (4) ];        /* 2つの節点集合の番号の */
                                       /* 小さいほうの値をマージ後の節点集合につける */
}

int main(){
  Arc a[Arc_num]={ {1, 0, 1, -1}, {2, 3, 4, -1}, {4, 0, 2, -1},
   {5, 1, 2, -1}, {6, 3, 5, -1}, {7, 4, 5, -1}, {8, 2, 5, -1},
   {10, 1, 4, -1}, {11, 1, 3, -1} };                      /* 枝情報の初期化 */
  int set[Node_num]={0, 1, 2, 3, 4, 5};            /* 各節点に初期番号を付加 */
  int ite, arc_num=0, n1, n2;

  for(ite=0; ite<Arc_num; ite++){               /* 枝長が短い順に枝をチェック */
    n1=a[ite].end_pt1; n2=a[ite].end_pt2;
    if( [ (5) ] != [ (6) ]){   /* 枝 a[ite] を最小木の要素として選択できるか？ */
      a[ite].selected = 1;                                       /* 枝を選択 */
      arc_num++;
      merge_trees(n1, n2, set);                           /* 節点集合のマージ */
    }
    /*  β  */
    if( arc_num == Node_num -1 ) break;               /* 最小木ができれば終了 */
  }
  return 0;
}
```