平成18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17年8月23日

注意事項

- 問題用紙は全部で10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）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（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を解答せよ．
- 「信号処理」を選択する者は，問題５を解答せよ．
- 「コンピュータ工学（ハードウェア）」を選択する者は，問題６（６－１および６－２）を解答せよ．
- 「コンピュータ工学（ソフトウェア）」を選択する者は，問題７（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のフィードバック制御系について以下の問いに答えよ．ただし，K_1, K_2, T は定数であり，$U(s)$, $E(s)$, $D(s)$, $Y(s)$ はそれぞれ入力 $u(t)$，偏差 $e(t)$，外乱 $d(t)$，出力 $y(t)$ のラプラス変換である．

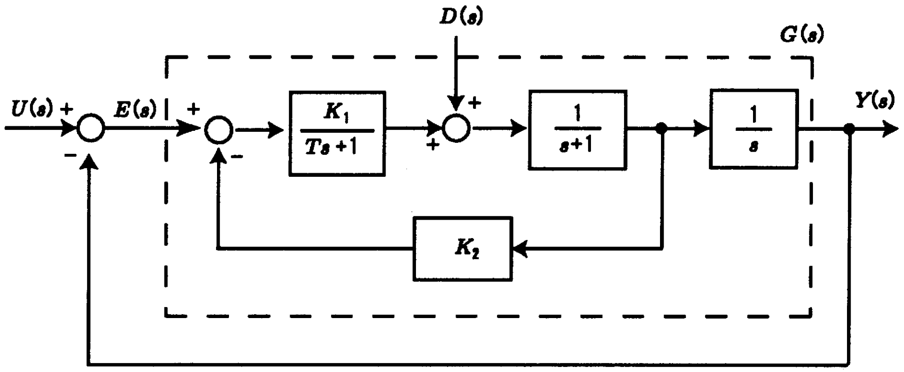

図1

(i) 外乱がない場合（$D(s)=0$）の $U(s)$ から $Y(s)$ までの伝達関数を求めよ．また，入力がない場合（$U(s)=0$）の $D(s)$ から $Y(s)$ までの伝達関数を求めよ．

(ii) $U(s)=0$，$K_1=0.5$，$K_2=1$，$T=0.5$ で，インパルス状の外乱 $D(s)=1$ が発生したときの出力 $y(t)$ を求めよ．

(iii) $U(s)$ から $Y(s)$ の閉ループ系が安定となる K_1, K_2, T の条件を求めよ．ただし，$T>0$ とする．

(iv) 外乱なし ($D(s)=0$)，$K_1=1$，$K_2=2$，$T=0$ の場合の，ステップ入力 ($u(t)=1$)，ランプ入力 ($u(t)=t$)，放物線入力 ($u(t)=t^2$) に対する定常偏差 $e(\infty)$ をそれぞれ求めよ．

(v) $T=0$ で外乱がない場合 ($D(s)=0$)，図1のフィードバック系の破線部の伝達関数 $G(s)$ が以下のようになった．

$$G(s)=\frac{\omega_n^2}{s^2+2\zeta\omega_n s}$$

ω_n と ζ を K_1，K_2 の関数としてあらわせ．

(vi) 伝達関数 $G(s)$ のナイキスト軌跡（ベクトル軌跡）の概形を示し，位相余裕を ζ を用いて表せ．

2．図２のフィードバック制御系に対して，以下の問いに答えよ．

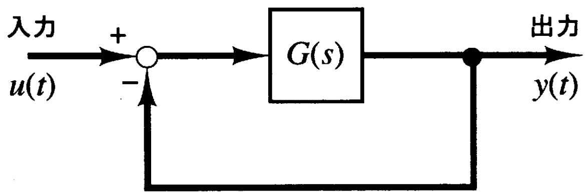

図 **2**

(i) 前向き伝達関数 $G(s)$ が

$$G(s) = \frac{{\omega_n}^2}{s(s+2\zeta\omega_n)}, \qquad (\zeta > 0,\ \omega_n > 0)$$

のように与えられるとき，閉ループ伝達関数を求めよ．単位ステップ関数入力に対して，出力 $y(t)$ にオーバーシュート（行き過ぎ量）が生じるための減衰係数 ζ に関する条件を答えよ．

(ii) 問 (i) と同じ $G(s)$ を考える．閉ループ系の周波数応答のゲイン特性が極大値を持つための減衰係数 ζ に関する条件を求めよ．また，極大値を持つときの周波数（共振周波数）を求めよ．

(iii) 前向き伝達関数 $G(s)$ として

$$G(s) = \frac{K(s+1)}{s(s-2)}$$

を考える．K の値を 0 から増加させたときの閉ループ系の根軌跡の概要を描け．ただし，軌跡の開始点，実軸からの分岐点，虚軸を横切る点の座標，複素数根を持つときの軌跡の方程式や $K \to \infty$ での解の挙動は正確に記述すること．

(iv) 問 (iii) の $G(s)$ において，$K = 8$ の場合を考える．閉ループ系の特性根を求めよ．また，単位ステップ関数入力に対する出力 $y(t)$ の概形を描け．出力に，オーバーシュートが生じる場合は，最大オーバーシュートとピーク時間（行き過ぎ時間）を求め，生じない場合は，その理由を述べよ．

(v) 問 (iv) と同じ $G(s)$ を考える．閉ループ系の周波数応答のゲイン曲線の概要を折れ線近似を用いて描け．

(vi) 問 (iv) と同じ $G(s)$ に対して，閉ループ系の周波数応答で共振が生じる（ゲイン特性が極大値を持つ）かどうか答えよ．生じる場合は，共振周波数を求め，生じない場合は，その理由を述べよ．

(vii) 以上の結果をもとに，伝達関数の零点（zero）がステップ応答や周波数応答に及ぼす影響について考察せよ．

電気機器

３－１．定格が，出力 5 kW，端子電圧 100 V，電流 50 A，回転速度 1800 rpm の他励式直流発電機がある．このとき，界磁電圧は 40 V，界磁電流は 7 A，電機子抵抗は 0.2 Ωで効率は 85％である．ブラシの電圧降下は無視する．この直流発電機について以下の問いに答えよ．

(i) 運転時の端子電圧を V, 電流を I_a, 電機子抵抗を R_a として，無負荷時の端子電圧 E_0 を式で表せ．また，無負荷時の端子電圧を計算して求めよ．

(ii) 定格負荷時の機械入力は何ワットか．

(iii) 鉄損，機械損などの固定損はいくらか．

(iv) 界磁電流と回転数を一定に保った状態で，負荷抵抗を種々調整して得られる端子電圧 V と電流 I_a の関係曲線を外部特性曲線という．この関係曲線を実測したところ，電流が大きい範囲において問(i)で求めた関係式と若干の相違が生じたという．どのような相違が生じるか，その原因とともに簡単に説明せよ．

３－２．4 極，周波数 60 Hz，端子電圧 220 V の三相かご形誘導電動機があり，その巻数比(N_1 / N_2, N_1：1 次側巻数，N_2：2 次側巻数)は 2 で，

1 次側のインピーダンス：	$r_1 + jx_1 = 0.3 + j\,0.2$ [Ω]
2 次側のインピーダンス：	$r_2 + jx_2 = 0.04 + j0.05$ [Ω]
励磁アドミタンス：	$\dot{Y}_0 = 0.006 - j0.03$ [S]

である(すべて 1 相あたりの値)．この電動機が 1728 rpm で回転しているとき，以下の問いに答えよ．ただし，1 次，2 次ともに星型（Y）結線とする．

(i) すべり s [%]を求めよ．

(ii) 1 次側換算 2 次抵抗 r_2'，1 次側換算 2 次リアクタンス x_2'を求め，1 次側に換算した 1 相あたりの簡易等価回路を求めよ．

(iii) 機械的出力 P_o (3 相分)を，簡易等価回路 1 次側印加電圧を V_1 として，変数 s，r_1，r_2'，x_1，x_2'を用いて式で表せ．また、計算して求めよ．

パワーエレクトロニクス

4．図 4－1 に主回路を示す自励式インバータにおいて図 4－2 に示すような制御（図中の記号はその期間中当該トランジスタにオン信号が与えられていることを示す）を行った．このとき、以下の各問に答えよ．

（i）負荷の相電圧 v_{UN} および負荷線間電圧 v_{UV} の波形を示せ．電圧のレベルおよびレベルの変化のタイミングと制御信号との位相関係が分かるように示すこと．

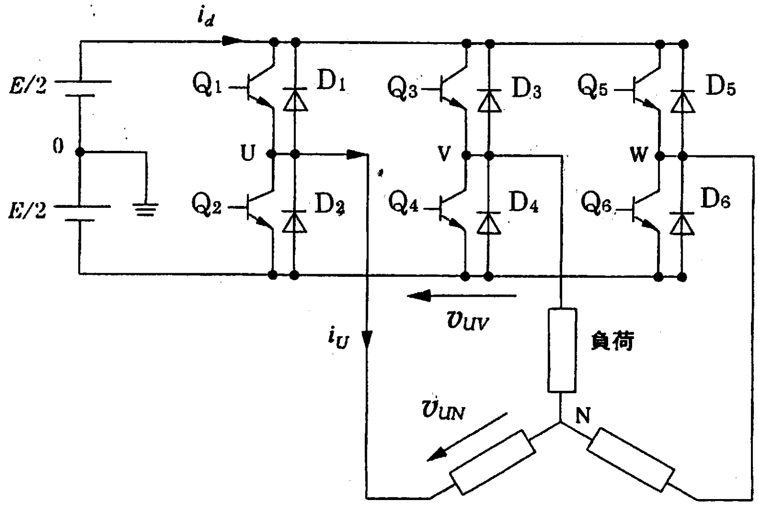

図 4－1　主回路

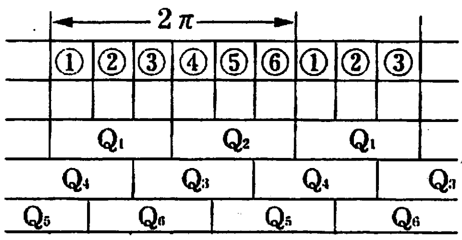

図 4－2　制御方式

（ii）問（i）で示した負荷線間電圧 v_{UV} の基本波実効値，v_{UV} の実効値，および総合ひずみ率（＝全高調波実効値／基本波実効値，THD; Total Harmonic Distortion）を求めよ．

（iii）$E = 200\mathrm{V}$ であるとき，各トランジスタに加わる電圧の最大値を求めよ．

（iv）$i_U = 100\sqrt{2}\sin(\omega t)$ で負荷力率は 0.5（遅れ）であるときトランジスタ Q_1 に流れる電流の波形を示し（負荷の相電圧 v_{UN} との位相関係が分かるように示すこと），その最大値および実効値を求めよ．

（v）図 4－1 に示したダイオードの役割を記せ．

（vi）自励式インバータに用いられる半導体スイッチング素子の名称を 2 種類アルファベットの略語とフルネームで示せ．

以上

信号処理

5．以下の空欄を埋め，問いに答えよ．

離散時間信号 $x[n]$, (n は整数) に対する離散時間フーリエ変換 (DTFT: Discrete Time Fourier Transform) は，Ω を角周波数を表す実数変数，j を虚数単位とするとき，

$$X(\Omega) = \sum_{n=-\infty}^{\infty} x[n] \exp(-j\Omega n)$$

で定義される．$X(\Omega)$ は $\boxed{(あ)}$ と呼ばれ，周期 $\boxed{(い)}$ の周期関数となる．この DTFT をコンピュータなどを用いて実際に計算するには (a)問題がある ことも知られている．

そこで，実際には次に示す (b)離散フーリエ変換(DFT: Discrete Fourier Transform) を用いる．DFT は長さ N の離散時間信号（データ系列）$x[n], (n = 0, ..., N-1)$ と長さ N の $\boxed{(あ)}$ 列 $X[k], (k = 0, ..., N-1)$ との変換（N 点 DFT と呼ぶ）を定義したもので，

$$X[k] = \boxed{(う)} W_N^{nk}, \quad 但し, W_N = \exp(\boxed{(え)})$$

である．N 点 DFT を直接計算するには高々，$\boxed{(お)}$ 回の複素乗算と $\boxed{(か)}$ 回の複素加算で十分である．

高速フーリエ変換 (FFT: Fast Fourier Transform) は，DFT を計算する際の乗加算回数を減らすために考案されたアルゴリズムで，ある問題を部分問題に分割し，それらを解き，解を統合することによって元の問題の解を与える $\boxed{(き)}$ 法に基づくものである．具体的には，N 点 DFT(ここでは N を 2 のべき乗とする) を 2 回の $\boxed{(く)}$ 点 DFT に分割し，さらに $\boxed{(く)}$ 点 DFT を 2 回の $\boxed{(け)}$ 点 DFT に分割するという操作を繰り返し，最終的に $\boxed{(こ)}$ 点 DFT に帰着させるのが FFT の考え方である．

FFT のアルゴリズムとして (c)時間間引き FFT と周波数間引き FFT が知られており，何れの場合も N 点 DFT を高々，$\boxed{(さ)}$ 回の複素乗算と $\boxed{(し)}$ 回の複素加算で実行することができる．また (d)逆離散フーリエ変換 (IDFT : Inverse DFT) も FFT を用いて高速に計算すること が可能である．

(i) 空欄（あ）〜（し）に当てはまるものを示せ．ただし，記号が同じ空欄には同じ字句が入るので注意すること．

(ii) 下線部 (a) について，どのような問題があるのか具体的に述べよ．

(iii) 下線部 (b) について，離散時間フーリエ変換 DTFT と離散フーリエ変換 DFT の関係，すなわち $X(\Omega)$ と $X[k]$ の関係について説明せよ．

(iv) 下線部 (c) について，時間間引き FFT あるいは周波数間引き FFT を用いて，4 点 DFT を計算する FFT の信号流れ図（バタフライ演算を基本計算単位とする）を図示せよ．また，データ系列 $(x[0], x[1], x[2], x[3]) = (0, 8, 2, 4)$ の 4 点 DFT を求めよ．

(v) 下線部 (d) について，FFT による IDFT のアルゴリズムを記述せよ．また，問い (iv) における 4 点 DFT の計算結果にそのアルゴリズムを適用し，元のデータ系列に戻ることを示せ．

コンピュータ工学（ハードウェア）

6 − 1　整数部 n 桁，小数部 m 桁の 2 進数 A を固定小数点表現で

$$A = a_n 2^n + \sum_{i=-m}^{n-1} a_i 2^i \quad (0 \le a_i \le 1)$$

と表す．最上位桁 $a_n 2^n$（$n+1$桁目）を符号部と呼び，正数のとき a_n=0，負数のとき a_n=1 とし，残りを数値部 $|A|$ と呼ぶ．以下の問いに答えよ．

(i) 整数部 n 桁，小数部 m 桁で表される負数 A の 2 の補数表現 $\overline{\overline{A}}$ の数値部 $|\overline{\overline{A}}|$ に関して，$|\overline{\overline{A}}| = 2^n - |A|$ が成り立つことを示せ．

(ii) 10 進数の −18.242 の固定小数点 2 進数表現（整数部 5 桁，小数部 3 桁）の (a) 符号ー絶対値表現，(b) 1 の補数表現，(c) 2 の補数表現をそれぞれ示せ．なお，2 進数 A の 1 の補数表現を $(\overline{A})_{c1}$，2 の補数表現を $(\overline{\overline{A}})_{c2}$ と表すものとする．

(iii) 1 の補数で表現された整数の負数 A, B を加算する際には，加算結果において，最上位ビットの桁上がり 2^{n+1} を無視して，1 を減算すればよい．小数部がなく，またオーバーフローは起こらないとして，このことが成り立つことを示せ．

(iv) 2 進数の除算に関する以下の文章中の空欄に当てはまるものを示せ．なお，同じ記号の欄には同じ内容が入る．

2 進数の除算も 10 進数の場合と同様に被除数から除数を引くという操作を各桁について行えばよい．いま，2 進数の正の整数 A, B について，$A \div B$ を実行する場合を考え，計算途中で除数 B を減算した結果を R とする．この場合，減算後の次の操作は R に関する条件により次のように変わる．ただし、2 は左に 1 ビットシフトする操作を表す．また，空欄ア、ウには R に関する条件式が入る．

(a) $2R - B$　　（$\boxed{\text{ア}}$）
(b) $2(\boxed{\text{イ}}) - B$　（$\boxed{\text{ウ}}$）

また，上記 (b) の操作を単純化して計算の高速化を図る手法があり，その場合，上記 (b) の操作は以下のようになる．

(b') $\boxed{\text{エ}}$　　（$\boxed{\text{ウ}}$）

たとえば，$(55)_{10} \div (3)_{10}$ をした場合，(b)を用いた場合の減算回数は $\boxed{\text{オ}}$ 回であり，(b')を用いた場合の減算回数は $\boxed{\text{カ}}$ 回である．

6－2 以下の演算を，命令フェッチ(IF)，命令デコード(D)，オペランドフェッチ(OF)，命令の実行(EX)，結果の格納(S)の 5 ステージ命令パイプライン制御により実行する．なお，命令語，データ，アドレスの長さはいずれも 1 語であり，主記憶装置へのデータバスは 1 本であるとする．以下の問いに答えよ．

1. (B) = (A) + R1　　アドレス A の主記憶の内容とレジスタ R1 の内容を加算し，主記憶のアドレス B に格納する．
2. (D) = (C) – R2　　アドレス C の主記憶の内容からレジスタ R2 の内容を減算し，主記憶のアドレス D に格納する．

(i) これらの演算を以下の機械命令順で実行する場合について，表 1 のような表を解答用紙に作成し，各クロックサイクルにおいて実行されるステージの名前を，表 1 中の命令 1 の例のように，上記の記号を用いて記入し，表 1 を完成させよ．なお，表は必要に応じて大きくし，またクロックサイクルとの関係が明確に分かるように記述すること．

1. (A) → R3　　主記憶アドレス A の内容(A)をレジスタ R3 に格納する
2. Add R3, R1, R4　　レジスタ R3 とレジスタ R1 の内容を加算し，結果をレジスタ R4 に格納する．
3. [R4] → B　　レジスタ R4 の内容[R4]を主記憶アドレス B に格納する．
4. (C) → R5　　主記憶アドレス C の内容(C)をレジスタ R5 に格納する．
5. Sub R5, R2, R6　　レジスタ R5 の内容からレジスタ R2 の内容を減算し，結果をレジスタ R6 に格納する．
6. [R6] → D　　レジスタ R6 の内容[R6]を主記憶アドレス D に格納する．

表 1

		クロックサイクル								
		1	2	3	4	5	6	7	....	
1	(A) → R3	IF	D	OF	EX	S				
2	Add R3, R1, R4									
3	[R4] → B									
4	(C) → R5									
5	Sub R5, R2, R6									
6	[R6] → D									

(ii) 一般に，機械命令の実行順序を工夫することで，パイプラインの乱れ（ストール）を減らし，実行効率を向上させることが可能である．必要なクロックサイクルが最小になるように，問い(i)における命令の実行順序(1～6)を変更して，表 2 を完成させよ．ただし，表 2 にあるように，1 番目に (A) → R3，3 番目に Add R3, R1, R4 を実行する命令実行順序だけを考えよ．

表 2

		クロックサイクル								
		1	2	3	4	5	6	7	....	
1	(A) → R3	IF	D	OF	EX	S				
2										
3	Add R3, R1, R4									
4										
5										
6										

コンピュータ工学（ソフトウェア）

7－1．図7－1のように，親の保持するデータ値が子の保持するデータ値以下となる２分木に対して，図7－2に示す配列による実現をヒープと呼ぶ．プログラムＡは，図7－2の配列heapを，データ値の大きさに応じてソートするプログラムである．以下の問いに答えよ．

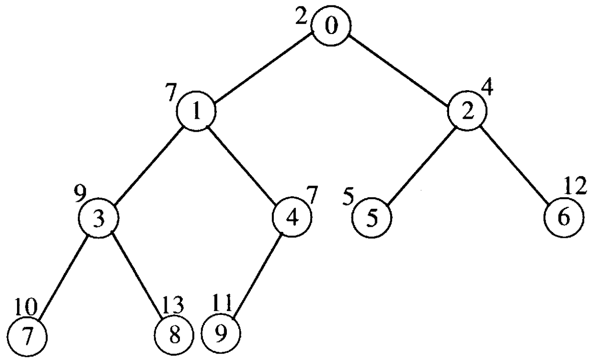

図7－1　　ヒープの条件を満たす２分木
（円内は節点番号，肩の数字はデータ値）

2	7	4	9	7	5	12	10	13	11

heap[0] ························ heap[9]

図7－2　　図7－1に示す２分木のヒープによる実現

プログラムＡ

```
#include <stdio.h>
#define N 10
void deletemin(int heap[ ], int *n)
{
     int p,q,min,tmp;
     if(*n < 0){ printf("Empty ¥n"); }
     else{
             min=heap[0]; heap[0]=heap[*n]; *n = *n-1; p=0;
             while(p*2+1 <= *n){
                 q=p*2+1;
                 if(p*2+2 <= *n && [  (1)  ] > heap[p*2+2]){ q= [  (2)  ] ; }
                 if([  (3)  ] ) break;
                 else{ tmp=heap[p]; heap[p]=heap[q]; heap[q]=tmp; }
                 p=q;
             }
             heap[*n+1]=min;
      }
}
int main()
{
     int k,n,heap[N]={2,7,4,9,7,5,12,10,13,11};
     n=N-1;
     for(k=0;k<N-1;k++){
         deletemin(heap, &n);
               < α >
     }
}
```

(ⅰ) 図7－1において，親の節点番号を i とすると，その左の子と右の子の節点番号は，i を用いてどのように表されるか述べよ．但し，i=0,1,2,3 とする．

(ⅱ) プログラムAの空欄(1)，(2)，(3)を埋めよ．

(ⅲ) プログラムAのmain関数における＜ α ＞の位置において,k=2 のときの heap[0], heap[1] ・・・・ heap[9]の値を示せ．

(ⅳ) プログラムAの計算量のオーダは，データ数 N を用いて $O(N\log N)$ となることを示せ．

7－2．プログラムBは，図7－3のグラフにおいて，始点となる節点0から他のすべての節点までの最短経路を求めるダイクストラ法のプログラムである．このプログラムは，節点0から最短距離の未探索節点を探索済み節点集合へ逐次加える．以下の問いに答えよ．

(ⅰ) プログラムBの空欄(4)，(5)，(6)を埋めよ．但し，(5)は2カ所あるが，同一の内容が入るので注意すること．

(ⅱ) プログラムBの下から3行目のprintf文に関して,k=2 の時の出力結果を示せ．

(ⅲ) プログラムBを変更し，図7—4の有向グラフに対する最短経路を求めるプログラムを答えよ．但し，解答にはプログラム全体ではなく，すべての変更箇所について変更前と変更後を対比させる形で記せ．

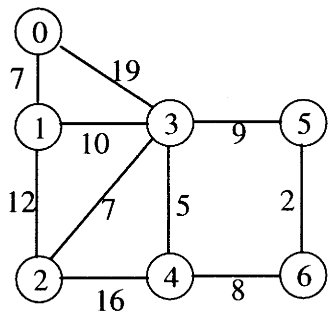

図7－3　対象とする無向グラフ
(円内は節点番号，枝に付された数字は節点間の距離)

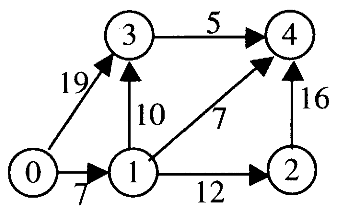

図7－4　対象とする有向グラフ

プログラムB

```
#include <stdio.h>
#define BN 100000
#define N 7
#define START 0
void sub(int *next, int *min, int [ (4) ], int distance[], int visited[], int prev[])
{   int i,j;
    i=*next; visited[i]=1; *min=BN;
    for(j=0;j<N;j++){
      if(visited[j]) continue;
      if(a[i][j] < BN && [ (5) ] < distance[j]){
          distance[j] = [ (5) ] ;
          prev[j] = i;
      }
      if(distance[j]<(*min)){ *min=distance[j]; [ (6) ] ;}
    }
}
int main()
{  int i,k,next,min,distance[N],visited[N],prev[N];
   int a[N][N] = { 0, 7,BN,19,BN,BN,BN,
                   7, 0,12,10,BN,BN,BN,
                  BN,12, 0, 7,16,BN,BN,
                  19,10, 7, 0, 5, 9,BN,
                  BN,BN,16, 5, 0,BN, 8,
                  BN,BN,BN, 9,BN, 0, 2,
                  BN,BN,BN,BN, 8, 2, 0};
   for(i=0;i<N;i++){ visited[i]=0; distance[i]=BN; }
   distance[START]=0; next=START;

   for(k=0;k<N-1;k++){
       sub(&next, &min, a, distance, visited, prev);
       printf("k=%d  %d %d %d¥n",k,next,min,prev[next]);
   }
}
```