

計算機数学入門とタイピングプログラム

兼 山 瓊 典

AN INTRODUCTION OF THE COMPUTER MATHEMATICS AND THE TYPING PROGRAMME

Tamafumi KANEYAMA

Summary

Nowadays the personal computer is commonly used for information retrieval and processing. Word processors are also found in almost any workplace and any home. However, most of those who work with them hardly have any training, though they believe it necessary to develop comprehensive computer literacy. Students in our math courses use computers for programming, and therefore must master basic typing and keyboard skills. For that reason I offer a course designed to develop those typing and keyboard skills.

Received Apr. 30. 1994.

1. は じ め に

今の時代は情報化時代といわれ、いろいろな情報が我々の周りにある。多くのメディアを通して、情報を得ることが出来る。その中でも、コンピュータによる情報の検索、取得は重要な位置を占めている。また個人でもパソコンを利用して仕事上のデータや個人的なデータを入力している。データの処理だけでなく、プログラムを作成して、役立てている人たちもいる。家庭や職場においてパソコンを使わない人々であっても、ワープロを使っていない人は少ないのではなかろうか。毎日使っている人は、あまりタイピングに興味がないかも知れない。しかし、キーは打てるが、自由に打つことが出来ないと思っても、タイピングの練習をしようとは思わない。

大学に入ってくる学生の場合はどうであろうか。家庭でのワープロや、小学校、中学校、高校でのパソコンの普及により、それらに触れる機会が多くなり、ある程度のタイピングが出来るようになってきている。しかしタイピングは基礎が大切であり、早い時期にそれをマスターすることが望まれる。それにより、ワープロや、コンピュータの操作が早く楽になることはいうまでもない。

これらのことにより、岐阜教育大学でも新しい入学生にダイビングの練習をすることが望まれ、行われるようになった。ここにダイビングの練習が出来るプログラムを作成しましたのであわせて発表します。

2. 入学生のタイプ経験

今年入学した学生は今までにどのくらいパソコンやワープロをあつかったことがあるだろうか。身近なところにあっても実際に使うとなると少ないものである。次の表は、今年の数学科の学生に対する調査である。

質問 今までにパソコン、ワープロを使ったことがありますか。

両方ともない	40.3%
パソコンだけである	19.3%
ワープロだけである	24.6%
両方ある	15.8%

この結果は予想したよりも使用経験が少ないものであった。さらにどのように使用したかを聞いてみると、パソコンの場合はゲームや何かのデータを入力したり、データの検索程度で簡単なものであった。またワープロについては簡単な文章や、手紙、年賀状といったものである。

指使いについては、使用経験が少ないためにひとつずつ探しながら、人差し指だけで打つというものが多い。

質問 タイピングについて

探しながら打つ	81.7%
キーを見ないで打てる	11.6%
すべての指を使って打てる	5.0%
指の使い方を知っている	1.7%

指使いが、探しながら打っているためになかなか早く打てない。そのためにキーを打つときに英字で打つか、カタカナで打つかという人の割合は、同じくらいであった。これはキーを捜すときは英字でもカタカナでも同じくらいであるという理由であろう。

今まで、パソコンやワープロにまったく触ったことのない学生の場合は、実際に操作してみたいという興味が多く、タイピングを練習してみたい学生がいる。しかし、ワープロをある程度使用した学生は、指使いや、タイプの速さの必要性を感じて、タイピングを練習して、ブラインドタッチで早くタイプできるようになりたいと望んでいる。

3. タイピング

学生の経験からみて、今の時期にタイピングを練習することが望ましい。早い時期にタイプの正しい方法を身につけることがよい。この練習がなく、我流でタイプをすると、ある程度速くタイプできるようになった段階で、正しい方法を身につけようとするのは非常に大変である。大学ではいろいろな学科があり、将来的にパソコンやワープロの利用方法は異なっているが、早い時期でのタイピングの練習は多いに役立つことであろうと思われる。数学科では2年目からプログラミングの実習を行うが、このタイピングの練習による、ブラインドタッチの修得はプログラミングの学習を効果的にすると思われる。

現在はタイピングの練習をしないでプログラミングの実習を行っているが、プログラム作成のアルゴリズムを考えるよりも入力することに多くの時間が取られている状態である。入力が速く出来るようになれば、授業の効率も大幅によくなることであろう。このため実際に扱うよりも前にタイピングの練習をすることが大切である。

4. プログラム

プログラムは、C言語(PC9801シリーズTurbo C 2.0)で書いてあります。プログラムの中にコメントが入っていますので、簡単に関数などについての説明をします。

type_ex ()	タイプ練習の入力部分
enter ()	タイプ練習の一行分の入力
key_disp_code ()	正しいキーの表示をコードで行なう。
key_disp ()	キーの表示を行なう。
result ()	結果を表示する。この中には速度、正解率、タイプ数、時間、キー別表示、総合表などがある。
read_file ()	練習問題を入力する。練習問題は、別のファイルになっているため、自由に変えることが出来る。
read_keyboard ()	キーボードのデータを読み込んで表示する。(データは別に用意する)

5. 最 後 に

タイピング練習のプログラムは数多くあるが、この機会に作成してみました。実際に使ってみて不満足なところが出てくるかも知れません。そのときは、少しずつでも直していき、より良いものにしたいと思っています。このプログラムを使って多くの人がブラインドタッチが出来るようになり、コンピューターやワープロが自由に使えるようになることを期待します。

```

/*
 *
 *      タイピング練習のプログラム (スーパータイプ)
 *
 *      S_TYPE.C  1994. 4. 17      By T. KANEYAMA
 *
 */

```

```

#include <stdio. h>
#include <conio. h>
#include <graphics. h>
#include <fcntl. h>
#include <sysstat. h>
#include <bios98. h>
#include <time. h>

typedef union {unsigned char c [20] ;
    struct {int    n_type;           /* 総打数           */
           int    n_correct;        /* = l. n_cc        */
           int    n_error;          /* = l. n_ee        */
           int    year;
           int    month;
           int    day;
           int    l_time;           /* = l. level_time  */
           int    t_time;          /* コース別合計時間 */
           int    count;           /* 練習回数         */
    } d;
} data;

typedef struct {int    level;        /* コース           */
               int    max;          /* コースの最大     */
               int    n_cc;         /* 正解打数         */
               int    n_ee;         /* 不正解打数       */
               unsigned char n_c [250] ; /* 文字別正解打数   */
               unsigned char n_e [250] ; /* 文字別不正解打数 */
               long level_time;     /* 時間             */
    } now;

int    f_course= 1;
long   t;
time_t t1, t2;
char   f_g [800] , f_w [800] , name [40] ;
data   d [30] ;
now    l;

main ()

```

```

{
    int g_driver, g_mode;

    detectgraph (&g_driver, &g_mode) ;
    initgraph (&g_driver, &g_mode, "");
    textmode (0) ;
    clrscr () ;
    cleardevice () ;
    gotoxy (28,12) ; printf ("データを読み込んでいます") ;

    read_keyboard () ;
    data_read (0) ;
    do {
        switch (disp_title () )
            {case 1 : type_ex () ; break;          /* f. 1 練習 */
             case 5 : result () ; break;         /* f. 5 成績 */
             case 6 : course () ; break;        /* f. 6 コース */
             case 10 : finish () ; break;       /* f. 10 終了 */
             default: break;
            }
        } while (1) ;
}

type_ex ()
{
    int i, f_len, f_index, f_lend, start;
    long lg;
    unsigned char c [3000] ;
    char cc [10] [7] = {"", "", "", "", "", "",
                      "", "", "", "", "", ""} ;

    clrscr () ;
    function_key_disp (cc) ;

    l_clear () ;
    f_course = 0;                               /* コース途中 */
    f_len = read_file (l.level, c) ;
    f_index = f_lend = start = 0 ;
    gotoxy (63, 2) ; printf ("コース %d", l.level + 1) ;
    bar (60, 134, 580, 136) ;
    t = 0 ;
    do {
        f_index = f_lend;
        line (60, 135, 60 + 520 L*f_lend/f_len, 135) ; /* どこまで進んだか表示 */
        file_disp (c, &f_lend) ;
        if (enter (c, f_index, f_lend, start ++ ) == 1) return;
        if (c [f_lend] == 0x0d) {

```

```

        if (c [f_lend + 1] == 0x0a) f_lend += 2;
        else f_lend++;
    }
} while (f_lend < f_len-1);
timer (0); /* 時間を止める */
l. level_time = t; /* コースの時間 */
f_course = 1; /* コース終了 */
line (60, 135, 580, 135);
data_write ();
gotoxy (10, 2); printf ("終了：ファンクションキーを押してください");
result ();
if (inc_level ()) {
    l. level ++;
    data_read(1);
    if (l. level > l. max) l. max = l. level;
}
}

int enter (unsigned char *c, int f_index, int f_lend, int start)
{
    int h, k, k_miss, index, k_scan, d_s;
    unsigned char scan, b_scan, inner, shift, b_shift, d;
    char mes1 [35], mes2 [35], mes3 [35];
    char cc [10] [7] = {" 中断  ", " ", " ", " ", " ", " ",
                      "コース終", " ", " ", " ", " ", " "};

    strcpy (mes1, "「カナ」キーが押されています");
    strcpy (mes2, "「カナ」キーが押されていません");
    strcpy (mes3, "スペースキーで開始です");
    b_scan = 0;
    b_shift = 0x20;
    if (start == 0)
    {
        d = c [f_index];
        if ((d > 160) && (d < 224)) h = 4; else h = 0, /*カタカナの時 h = 4 */
        do {
            key_shift (&shift);
            if (b_shift != shift)
            {
                shift_disp (b_shift, 3);
                b_shift = shift;
                shift_disp (shift, 1);
                gotoxy (10, 2);
                if (shift & 4) printf ("%s", mes1); else printf ("%s", mes2);
            }
        } while ((shift & 4) != h);
    }
}

```

```

gotoxy (10, 2) ; printf ("%s", mes3) ;
do {
    key (& scan, & inner, & shift) ;
    if (scan == 0x67) return (1) ;           /* f. 6 コース終 */
    } while (scan! = 0x34) ;                /* space キー */

gotoxy (10, 2) ; printf ("%35s", "") ;
function_key_disp (cc) ;
timer (1) ;
}
index = 10 ;
textcursor (DISP_CURSOR) ;
do {
    d = c [f_index ++] ;
    k_miss = 0 ;
    d_s = key_disp_code (d, 2) ;           /* key の指示 */
    do {
        gotoxy (index, 6) ;
        k = key (& scan, &inner, &shift) ;
        shift & = 0x1e ;
        if (b_shift! = shift)
            {key_disp_code (d, 2) ;
             shift_disp (b_shift, 3) ;
             b_shift = shift ;
             shift_disp (shift, 1) ;
             gotoxy (10, 2) ;
             if (shift & 4) printf ("%s" , mes1) ; else printf ("%35s" , "") ;
             gotoxy (index, 6) ;
            }
        if (k) {
            if (inner == d) {
                key_disp (b_scan, 8) ;
                break ;
            }
            k_miss = 1 ;                    /* 間違いあり */
            key_disp (b_scan, 8) ;
            if (d_s == b_scan) key_disp_code (d, 2) ;
            b_scan ;
            key_disp (scan, 4) ;           /* 間違いのキーの表示 */
            putch (7) ;
        }
        if (scan == 0x62) pause () ;       /* f. 1 中断 */
        if (scan == 0x67) return (1) ;     /* f. 6 コース終 */
    } while (scan! = 0xfe) ;
    l.n_c [d] ++ ;

```

```

        l.n_cc ++ ;
        if (k_miss == 1) {
            l.n_e [d] ++ ;
            l.n_ee ++ ;
        }
        key_disp_code (d,8);          /* key の指示を消す */
        printf ("%c", d);
        index ++ ;
    } while (f_index < f_lend);
    return (2);
}
int pause ()
{   unsigned char scan, inner, shift ;

    timer (0);
    gotoxy (4, 25); cprintf (" 再開 ");
    gotoxy (42, 25); cprintf (" ");
    gotoxy (10, 2); printf ("中断中です");
    textcursor (NODISP_CURSOR);
    do {
        key (& scan, & inner, & shift);
        } while (scan != 0x62);      /* f. 1 で再開 */
    gotoxy ( 4, 25); cprintf (" 中断 ");
    gotoxy (42, 25); cprintf ("コース終");
    gotoxy (10, 2); printf (" ");
    textcursor (DISP_CURSOR);
    timer (1);
}

timer (int k)
{   int i;

    if (k) {time (&t1); return;}    /* k=1 時間を計り始める */
    time (& t2);                    /* k=0 時間を止め t に */
    t += difftime (t2, t1);
}

int inc_level ()                    /* コースを上げるか */
{   int k;

    k=1. level;
    if (k== 29) return (0);
    if (!d [k] . d.n_correct) return (0);
    if (d [k] . d.n_correct < 4*d [k] . d.n_error) return (0); /* 正打率で判断 */
    if (6*d [k] . d.n_correct < 8L*d [k] . d.l_time) return (0); /* 速度で判断 */
}

```



```

        return (1) ;
    }
l_clear ()
{
    int i;
    l.n_cc = l.n_ee = 0;
    for (i = 0; i < 250; i++) l.n_c [i] = l.n_e [i] = 0;
    l.level_time = 0;
}
int disp_title ()
{
    unsigned char scan, inner, shift;
    char c [10] [7] = { "  練習  ", "      ", "      ", "      ", "      ", " 成績  ",
                      " コース ", "      ", "      ", "      ", "      ", " 終了  " };

    clrscr () ;
    function_key_disp (c);
    keyboard_copy (1, 0) ;
    disp_waku (1) ;
    putch (7);
    textcursor (NODISP_CURSOR) ;
    gotoxy (63, 2) ; printf ("コース %d", l.level + 1) ;
    gotoxy (10, 2) ; printf ("ファンクションキーを押してください") ;
    do {
        key (& scan, & inner, & shift);
        if (scan == 0x62) return (1) ;           /* l. 1 */
        if (scan == 0x66) return (5) ;           /* f. 5 */
        if (scan == 0x67) return (6) ;           /* f. 6 */
        if (scan == 0x6b) return (10) ;          /* f. 10 */
    } while (1) ;
}

shift_disp (unsigned char s, int col)
{
    if ((s & 0x01) != 0) (k_d (84, 88, 60, col) ; k_d (408, 88, 40, col) ;) /* * SHIFT */
    if ((s & 0x02) != 0) k_d (108, 64, 24, col); /* * CAPS */
    if ((s & 0x04) != 0) k_d (132, 112, 24, col) ; /* * カナ */
    if ((s & 0x08) != 0) k_d (156, 112, 24, col) ; /* * GRPH */
    if ((s & 0x10) != 0) k_d (84, 64, 24, col) ; /* * CTRL */
}

int key_disp_code (unsigned char d, int col)
{
    int w, s;
    KEY_INFO ky;

    static char c [256] = {0,
                           0,0,0,0,0,0,0,0,0,0,          0,0,28,0,0,0,0,0,0,0,0,
    };

```

```

0,0,0,0,0,0,0,60,59,58,      61,52,1,2,3,4,5,6,7,8,
9,39,38,48,11,49,50,10,1,2,   3,4,5,6,7,8,9,39,38,48,
11,49,50,26,29,45,43,31,18,32, 33,34,23,35,36,37,47,46,24,25,
16,19,30,20,22,44,17,42,21,41 27,13,40,12,51,0,29,45,43,31,

18,32,33,34,23,35,36,37,47,46, 24,25,16,19,30,20,22,44,17,42,
21,41,27,13,40,26,0,0,0,0,   0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,   0,0,0,0,0,0,0,0,0,0,
49,27,40,48,50,10,3,18,4,5,   6,7,8,9,41,13,3,18,4,5,
6,20,33,34,39,45,42,31,19,25, 43,16,29,41,17,30,22,23,1,48,

36,32,44,2,12,11,35,46,40,50, 47,7,8,9,24,37,49,38,51,10,
21,26,27,0,0,0,0,0,0,0,   0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,   0,0,0,0,0,0,0,0,0,0,

```

/* ASCII を SCAN CODE になおす */

```

r [256] = {0,
    0,0,0,0,0,0,0,0,0,0, 0,0,0,0,0,0,0,0,0,0,
    0,0,0,0,0,0,0,0,0,0, 0,0,4,4,4,4,4,1,1,1,
    1,1,1,0,0,0,0,0,0,0, 0,0,0,0,0,0,0,0,0,1,
    1,1,1,0,5,5,5,5,5,5, 5,5,2,2,2,2,2,2,2,2,
    5,5,5,5,2,5,5,5,2,5, 0,0,0,0,1,1,6,6,6,6,

    6,6,6,6,3,3,3,3,3,3, 3,3,6,6,6,6,3,6,6,6,
    3,6,1,1,1,1,0,0,0,0, 0,0,0,0,0,0,0,0,0,0,
    0,0,0,0,0,0,0,0,0,0, 0,0,0,0,0,0,0,0,0,0,
    1,1,1,1,1,1,4,4,4,4, 1,1,1,1,4,0,0,0,0,0,
    0,0,0,0,0,0,0,0,0,0, 0,0,0,0,0,0,0,0,0,0,

    0,0,0,0,0,0,0,0,0,0, 0,0,0,0,0,0,0,0,0,0,
    0,0,0,0,0,0,0,0,0,0, 0,0,0,0,0,0,0,0,0,0,
    0,0,0,0,0,0,0,0,0,0, 0,0,0,0,0,0,0,0,0,0,

```

/* SHIFT KEY 0=なし 1 = 必ず左 2 = 左 3 = CAPS のとき左
4 = 必ず右 5 = 右 6 = CAPS のとき右 */

```

s = c [d] ;
key_disp (s, col);
ky. cmmd=2;
w = bios98key (&ky) &2;
switch (r [d])
{case 1:      l_shift (2) ; r_shift (3) ; break;
  case 2: if (w) {l_shift (3) ; r_shift (3) ;}
          else {l_shift (2) : r_shift (3) ;} break;

```

計算機数学入門とタイピングプログラム

```

        case 3: if (w) {l_shift (2) ; r_shift (3) ;}
                else {l_shift (3) ; r_shift (3) ;} break;
        case 4:      l_shift (3) ; r_shift (2) ; break;
        case 5: if (w) {l_shift (3) ; r_shift (3) ;}
                else {l_shift (3) ; r_shift (2) ;} break;
        case 6: if (w) {l_shift (3) ; r_shift (2) ;}
                else {l_shift (3) ; r_shift (3) ;} break;
        default:    l_shift (3) ; r_shift (3) ;} break;
    } ;
    hand (s, col) ;
    return (s) ;
}
l_shift (int col)
{
    if (col == 3) {k_d (84, 88, 60, 8) ; finger (1, 7) ;}
    else {k_d (84, 88, 60, col) ; finger (1, col) ;}
}
r_shift (int col)
{
    if (col == 3) {k_d (408, 88, 40, 8) ; finger (10, 7) ;}
    else {k_d (408, 88, 40, col) ; finger (10, col) ;}
}
hand (int s, int col)
{
    int l;

    switch (s) {
        case 1 : case 0x10 : case 0x1d : case 0x29 : case 0x0f : l = 1 ; break;
        case 2 : case 0x11 : case 0x1e : case 0x2a : l = 2 ; break;
        case 3 : case 0x12 : case 0x1f : case 0x2b : l = 3 ; break;
        case 4 : case 0x13 : case 0x20 : case 0x2c :
        case 5 : case 0x14 : case 0x21 : case 0x2d : l = 4 ; break;
        case 6 : case 0x15 : case 0x22 : case 0x2e:
        case 7 : case 0x16 : case 0x23 : case 0x2f : l = 7 ; break;
        case 8 : case 0x17 : case 0x24 : case 0x30 : l = 8 ; break;
        case 9 : case 0x18 : case 0x25 : case 0x31 : l = 9 ; break;
        case 0x34 : finger (6, col) ; l = 5 ; break;
        default : if (s <0x33) l = 10 ; else l = 0;
    }
    finger (l, col);
}
key_disp (int s, int col)
{
    int w;

    w = 24;

```

```

switch (s) {
case 0x0e : k_d (420, 16, 28, col) ; break;           /*BS */
case 0x0f : k_d (84, 40, 36, col) ; break;          /*TAB */
case 0x1c : k_d (408, 40, 40, col);
            k_d (420, 64, 28, col) ;
            k_d (532, 112, 24, col) ; break;        /*CR */
case 0x34 : k_d_sp (col) ; break;                   /*SPACE*/
case 0x51 : case 0xa5 : case 0xb5 :
            k_d (180, 112, 36, col) ; break;        /*NFER */
case 0x35 : case 0xa1 : case 0xb1:
            k_d (372, 112, 36, col) ; break;        /*XFER */
case 0x36 : case 0x37 : case 0x38 : case 0x39 : break;
            /*ROLL_UP ROLL_DOWN INS DEL */
case 0x3a : case 0x3b : case 0x3c : case 0x3d : break; /*矢印キー*/
default : if (s <= 0x0e) {k_d (84+24*s, 16, w, col) ;
                    break;
                }
            if (s <= 0x1b) {k_d (120+24* (s-16) , 40, w, col) ;
                    break;
                }
            if (s <= 0x28) {k_d (132+24* (s-29) , 64, w, col);
                    break;
                }
            if (s <= 0x33) {k_d (144+24* (s-41) , 88, w, col) ;
                    break;
                }
            if (s <= 0x50) {k_d (460+ ((s-62) %4) *24, 16+ ((s-62)/4) *24, w, col);
                    break;
                }
}
}
k_d (int xl, int yl, int w, int col)
{
    int    x, y, k, h;
    char b [810] ;           /*60*24 = 806*/

    if (col == 8) {setactivepage (1) ;               /*col = 8戻す*/
        x = xl ; y = yl+150;
        getimage (x, y, x+w-1, y+24, b) ;
        setactivepage (0);
        putimage (x, y, b, 0) ;
        return;
    }
    setactivepage (1) ;
    x = xl ; y = yl;
    getimage (x, y, x+w-1, y+24, b);
}

```

計算機数学入門とタイピングプログラム

```

setactivepage (0) ;
h = (w+7)/8;
k = 0;
x =xl ; y = yl+150;
if (col & 1) {
    putimage (x, y, b, k) ;
    k = 2;
}
if (col & 2) {
    b [h] = b [0] ;
    b [h+2] = b [2] ;
    putimage (x, y, b+h, k) ;
    k = 2;
}
if (col & 4) {
    b [2*h] = b [0] ;
    b [2*h+2] = b [2] ;
    putimage (x, y, b+2*h,k) ;
    k = 2;
}
}

k_d_sp (int col)
{
    int    p [8] ;

    if (col == 8) col = 7;
    p [0] = p [6] = 217;
    p [1] = p [3] = 263;
    p [2] = p [4] = 371;
    p [5] = p [7] = 285;
    setcolor (col);
    setfillstyle (1, col) ;
    fillpoly (4, p) ;
    putpixel (216, 273, 0) ;
    putpixel (371, 273,0) ;
    putpixel (216, 285, 0) ;
    putpixel (371, 285, 0) ;
}
result()
{
    do {
        switch (result_title())
        {case 1 : cleardevice () ; return;           /*戻る      */
        case 2 : result_speed () ; break;          /*速度      */
        case 3 : result_rate () ; break;           /*正解率    */

```

```

        case 4 : result_stroke () ; break;          /* タイプ数 */
        case 5 : result_table () ; break;         /* キー別 */
        case 6 : result_level_disp () ; break;   /* 解説 */
        case 7 : result_total () ; break;        /* 総合表 */
        case 9 : result_print () ; break;        /* 印刷 */
        default : break;
    }
} while (1) ;
}
result_line ()
{
    int i, p [8] ;

    gotoxy (30, 23) ; cprintf ("名前 %s", name) ;
    gotoxy (10, 20) ; printf ("コース") ;
    gotoxy (20, 20) ; printf ("5") ;
    gotoxy (30, 20) ; printf ("10") ;
    gotoxy (40, 20) ; printf ("15") ;
    gotoxy (50, 20) ; printf ("20") ;
    gotoxy (60, 20) ; printf ("25") ;
    gotoxy (70, 20) ; printf ("30") ;

    setfillstyle (1, 7) ;
    p [0] = p [6] = 75 ;
    p [1] = p [3] = 50 ;
    p [2] = p [4] = 565 ;
    p [5] = p [7] = 300 ;
    fillpoly (4, p) ;

    setfillstyle (1, 3) ;
    setcolor (2) ;
    bar (75, 45, 565, 50) ; bar (75, 300, 565, 305) ; /* 横線 */
    bar (75, 50, 80, 300) ; bar (560, 50, 565, 300) ; /* 縦線 */
    for (i = 0 ; i < 30 ; i++) bar (96+i*16, 50, 96+i*16, 300) ; /* 縦線 */
    for (i = 0 ; i < 4 ; i++) line (80, 250-i*50, 560, 250-i*50) ; /* 横線 */
    for (i = 0 ; i < 5 ; i++) line (160+i*80, 55, 160+i*80, 300) ; /* 縦線 */
}
result_speed ()
{
    int i, y, yb;
    long x, t;

    cleardevice () ;
    clrscr () ;
    textattr (0xe5) ;
    gotoxy (30, 2) ; cprintf ("1分間のタイプ速度") ;
    gotoxy (7, 3) ; printf ("回/分") ;

```

計算機数学入門とタイピングプログラム

```
gotoxy (7, 4) ; printf ("250") ;
gotoxy (7, 7) ; printf ("200") ;
gotoxy (7, 10) ; printf ("150") ;
gotoxy (7, 13) ; printf ("100") ;
gotoxy (7, 16) ; printf (" 50") ;
gotoxy (7, 19) ; printf (" 0") ;
result_line () ;

setcolor (4) ;
t = d [0] . d . l_time;
if (x) x = 60L*d [0] . d . n_correct/t ; else x=0;
yb = 300-x;
if (yb <50) yb = 50;
for (i = 1 ; i <30 ; i++)
    {t = d [i] . d . l_time;
    if (t) x = 60L*d [i] . d . n_correct/t; else x = 0;
    y = 300-x;
    if (y <50) y = 50;
    line (80+i*16, yb, 96+i*16, y) ;
    yb = y;
    }
}

result_rate ()
{    int i, k, x, y, yb;

cleardevice () ;
clrscr () ;
textattr (0xe5);
gotoxy (36, 2) ; cprintf ("正解率") ;
gotoxy (7, 3) ; printf ("%") ;
gotoxy (7, 4) ; printf ("100") ;
gotoxy (7, 7) ; printf (" 80") ;
gotoxy (7, 10) ; printf (" 60") ;
gotoxy (7, 13) ; printf (" 40") ;
gotoxy (7, 16) ; printf (" 20") ;
gotoxy (7, 19) ; printf (" 0") ;
result_line () ;

setcolor (4) ;
k = d [0] . d . n_correct;
if (k) x = 250L*k/(d [0] . d . n_error+k) ; else x = 0;
yb= 300-x;
if (yb <50) yb = 50;
    {t = d [i] . d . l_time;
```

```

        if (t) x = 60L*d [i] . d. n_correct/t; else x = 0;
        y = 300-x;
        if (y <50) y = 50;
        line (80+i*16, yb, 96+i*16, y) ;
        yb = y;
    }
}

for (i = 1 ; i <30 ; i++)
cleardevice () ;
clrscr () ;
textattr (0xe5);
gotoxy (36, 2) ; cprintf("正解率") ;
gotoxy (7, 3) ; printf ("%") ;
gotoxy (7, 4) ; printf ("100") ;
gotoxy (7, 7) ; printf (" 80") ;
gotoxy (7, 10) ; printf (" 60") ;
gotoxy (7, 13) ; printf (" 40") ;
gotoxy (7, 16) ; printf (" 20") ;
gotoxy (7, 19) ; printf (" 0") ;
result_line () ;

setcolor (4) ;
k = d [0] . d. n_correct;
if (k) x = 250L*k/(d [0] . d. n_error+k) ; else x = 0;
yb= 300-x;
if (yb <50) yb = 50;
for (i = 1 ; i <30 ; i++)
    {k = d [i] . d. n_correct;
    if (k)x = 250L*k/(d [i] . d. n_error+k) ; else x = 0;
    y = 300-x;
    if (y <50) y = 50;
    line (80+i*16, yb, 96+i*16, y) ;
    yb = y;
    }
}

result_stroke ()
{ int i, x, y, yb;

    cleardevice () ;
    clrscr () ;
    textattr (0xe5);
    gotoxy (20, 2) ; cprintf ("タイプ数 (折れ線) と練習時間 (棒) ") ;
    gotoxy (5, 3) ; printf (" 回") ;

```


計算機数学入門とタイピングプログラム

```
gotoxy (5, 4) ; printf ("5000") ;
gotoxy (5, 7) ; printf ("4000") ;
gotoxy (5, 10) ; printf ("3000") ;
gotoxy (5, 13) ; printf ("2000") ;
gotoxy (5, 16) ; printf ("1000") ;
gotoxy (5, 19) ; printf (" 0") ;

gotoxy (72, 4) ; printf ("50分") ;
gotoxy (72, 7) ; printf ("40") ;
gotoxy (72, 10) ; printf ("30") ;
gotoxy (72, 13) ; printf ("20") ;
gotoxy (72, 16) ; printf ("10") ;
gotoxy (72, 19) ; printf ("0") ;
result_line () ;
setcolor (4) ;
yb = 300-d [0] . d. n_type/20;
if (yb <50) yb = 50;
for (i = 1 ; i <30 ; i++)
    {y = 300-d [i] . d. n_type/20;
    if (y <50) y = 50;
    line (80+i*16, yb, 96+i*16, y) ;
    yb = y;
    }
setfillstyle (1, 2) ;
for (i = 0 ; i <30 ; i++)
    {y = 300L-d [i] . d. t_time/12;
    if (y <50) y = 50;
    bar (94+i*16, y, 98+i*16, 300) ;
    }
}

result_table ()
{
    int i, x, y, p [8] ;

    cleardevice () ;
    clrscr () ;
    i = l. level;
    gotoxy (10, 1) ;cprintf ("%s さんのコース %d の最終練習結果", name, i+1) ;
    gotoxy (45, 2) ;
    if (!d [i] . d. count) printf ("このコースはまだ終了していません") ;
    else {if (f_course)
        printf ("最終練習は 19%d 年 %d 月 %d 日です。",
            d [i] . d. year, d [i] . d. month, d [i] . d. day) ;
        else printf ("このコースの途中で終わった結果です") ;
    }
}
```

```

setfillstyle (1, 7) ;
p [0] =p [6] = 2;
p [1] =p [3] = 30;
p [2] =p [4] = 637;
p [5] =p [7] = 370;
fillpoly (4, p) ;

setfillstyle (1, 2) ;
setcolor (2) ;
for (i = 0 ; i <22 ; i++) line (4, i*16+32, 635, i*16+32) ; /*横線 */
for (i = 0 ; i <16 ; i++) line (i*42+4, 32, i*42+4, 368) ; /*縦線 */
for (i = 0 ; i <6 ; i++) bar (i*126+4, 32, i*126+5, 368) ; /*縦太線*/
for (i = 0 ; i <5 ; i++)
    {put_kanji ("文字" , 8+126*i, 33, 0, 0) ;
     put_kanji ("正解数" , 53+126*i, 33, 1, 0) ;
     put_kanji ("不正解数" , 90+126*i, 33, 1, 0) ;
    }
if (kana ()) return; /*カタカナ*/
for (i = 33 ; i <126 ; i++)
    {position (i, &x, &y);
     moji_disp (i, x, y);
    }
}
result_total ()
{
    int i, x, y, p [8] ;

    cleardevice () ;
    clrscr () ;
    gotoxy (20, 2) ; cprintf ("%s さんの総合練習結果" , name) ;
    setfillstyle (1, 7) ;
    p [0] =p [6] = 12;
    p [1] =p [3] = 60;
    p [2] =p [4] = 627;
    p [5] =p [7] = 340;
    fillpoly (4, p) ;

    setfillstyle (1, 2) ;
    setcolor (2) ;
    bar (14, 62, 625, 64) ; bar (14, 95, 625, 97) ; bar (14, 336, 625,338) %
                                                /*横太線*/
    bar (14, 62, 16, 338) ; bar (319, 62, 320, 338) ; bar (623, 62, 625, 338) ;
                                                /*縦太線*/
    for (i = 0 ; i <15 ; i++) line (16, i+16+96, 623, i*16+96) ; /*横線 */
    for (i = 0 ; i <2 ; i++) {line (i*304+64, 64, i*304+64, 336);/*縦線 */
                             line (i*304+94, 64, i*304+94, 336);
    }
}

```

```

        line (i * 304 + 160, 64, i * 304 + 160, 336);
        line (i * 304 + 224, 64, i * 304 + 224, 336);
    }
    for (i = 0 ; i < 2 ; i++)
        {put_kanji ("コース", 16 + 304 * i, 72, 0, 0) ;
         put_kanji ("練習", 64 + 304 * i, 64, 0, 0) ;
         put_kanji ("回数", 64 + 304 * i, 80, 0, 0) ;
         put_kanji ("総打数", 104 + 304 * i, 72, 0, 0) ;
         put_kanji ("総時間", 168 + 304 * i, 64, 0, 0) ;
         put_kanji (" (分) ", 176 + 304 * i, 80, 0, 0) ;
         put_kanji ("最終正解率", 232 + 304 * i, 64, 0, 0) ;
         put_kanji (" (%) ", 256 + 304 * i, 80, 0, 0) ;
        }

    textattr (0x1) ;
    textcolor (0) ;
    for (i = 0 ; i < 15 ; i++)
        {gotoxy (5, 7+i) ; cprintf ("%2d" , i+1) ;
         x = d [i] . d. count;
         if (x) {gotoxy (10, 7+i) ; cprintf ("%2d" , x) ;
                  gotoxy (15, 7+i) ; cprintf ("%4d" , d [i] . d. n_type) ;
                  gotoxy (23, 7+i) ; cprintf ("%4d" , d [i] . d. t_time/60) ;
                  x = d [i] . d.n_correct;
                  y = d [i] . d. n_error;
                  gotoxy (35, 7+i) ; cprintf ("%3d" , 100L * x / (x + y)) ;
                  }
         else {gotoxy (11, 7+i) ; cprintf ("-") ;
                gotoxy (17, 7+i) ; cprintf ("--") ;
                gotoxy (25, 7+i) ; cprintf ("--") ;
                gotoxy (36, 7+i) ; cprintf ("--") ;
                }
        }
    for (i = 15 ; i < 30 ; i++)
        {gotoxy (43, i-8) ; cprintf ("%2d" , i+1) ;
         x = d [i] . d. count;
         if (x) {gotoxy (48, i-8) ; cprintf ("%2d" , x) ;
                  gotoxy (53, i-8) ; cprintf ("%4d" , d [i] . d. n_type) ;
                  gotoxy (61, i-8) ; cprintf ("%4d" , d [i] . d. o_time/60) ;
                  x = d [i] . d. n_correct;
                  y = d [i] . d. n_error;
                  gotoxy (73, i-8) ; cprintf ("%3d" , 100L * x / (x + y)) ;
                  }
         else {gotoxy (49, i-8) ; cprintf ("-") ;
                gotoxy (55, i-8) ; cprintf ("--") ;
                gotoxy (63, i-8) ; cprintf ("--") ;
                }
        }

```

```

        gotoxy (74, i-8) ; cprintf ("--") ;
    }
}
result_print ()
{
    int i, j, x, y;
    TIME_INFO time_info;

    fprintf (stdprn, "%s さんの総合練習結果", name) ;
    time_info. cmmnd = 0;
    bios98time (&time_info) ;                /* 現在の時刻を読む */

    fprintf (stdprn, "          19%d 年 %d 月 %d 日 %r%r%r%r",
             (time_info.year/16) * 10+ (time_info.year%16),
             time_info. month week/16,
             (time_info. day/16) * 10+ (time_info. day%16)) ;

    fprintf (stdprn, "コース, 練習回数, 総打数, 総時間 (分), ") ;
    fprintf (stdprn, "最終正解率 (%), 最終速度 (回/分) %r%r%r%r" ) ;

    for (i = 0 ; i <30 ; i++)
        {if (i%5 == 0) {
            for (j = 0 ; j <75 ; j++) fprintf (stdprn, "--") ;
            fprintf (stdprn, "%r%r" ) ;
        }
        fprintf (stdprn, "%4d", i+1) ;
        x = d [i] . d. count;
        if (x) {
            fprintf (stdprn, "%9d", d [i] . d. count);
            fprintf (stdprn, "%10d", d [i] . d. n_type);
            fprintf (stdprn, "%10d", d [i] . d. t_time/60);
            x = d [i] . d. n_correct;
            y = d [i] . d. n_error;
            fprintf (stdprn, "%15d", 100L*x/(x+y)) ;
            fprintf (stdprn, "%18d%r%r", 60L*x/d [i] . d. l_time) ;
        }
        else {
            fprintf (stdprn, "          --          --" ) ;
            fprintf (stdprn, "          --          --%r%r" ) ;
        }
    }
    for (j = 0 ; j <75 ; j++) fprintf (stdprn, "--" ) ;
    fprintf (stdprn, "%r%r%r%r" ) ;
}
position (int i, int *x, int *y)                /* 文字別の表示の位置 */

```

計算機数学入門とタイピングプログラム

```

{
    if ((i > 96) && (i < 117)) { *x = 4; *y = 3+i-96; return; } /* a-t */
    if ((i > 116) && (i < 123)) { *x = 20; *y = 3+i-116; return; } /* u-z */
    if ((i > 64) && (i < 79)) { *x = 20; *y = 9+i-64; return; } /* A-N */
    if ((i > 78) && (i < 96)) { *x = 36; *y = 3+i-78; return; } /* O-Z */
    if ((i > 47) && (i < 64)) { *x = 52; *y = 3+i-47; return; } /* 0-9-? */
    if ((i > 32) && (i < 48)) { *x = 67; *y = 3+i-32; return; } /* !- */
    if ((i > 122) && (i < 126)) { *x = 36; *y = 20+i-122; }
}

```

```

moji_disp (int d, int x, int y) /* タイプの結果を文字別に表示 */
{
    int m;

```

```

    textattr (0xel);
    if (l.n_c [d] )
        {textcolor (0x20);
         gotoxy (x, y) ; cprintf ("%c" , d) ;
         gotoxy (x+4, y) ; cprintf ("%3d" , l.n_c [d] ) ;
        }
    else {textcolor (0) ;
         gotoxy (x, y) ; cprintf ("%c" , d) ;
         gotoxy (x+5, y) ; cprintf ("--") ;
        }
    gotoxy (x+11, y) ;
    if (l.n_e [d])
        {textcolor (0x40) ;
         gotoxy (x+9, y) ; cprintf ("%3d" , l.n_e [d]) ;
        }
    else {textcolor (0) ;
         gotoxy (x+10, y) ; cprintf ("--") ;
        }
}

```

```

int kana ()
{
    int i, x;

    x = 0;
    for (i = 161 ; i < 224 ; i++) x += l.n_c [i] ;
    if (!x) return (0) ;
    textattr (0xel) ;
    for (i = 177 ; i < 197 ; i++) moji_disp (i, 4, i-173) ;
    for (i = 197 ; i < 217 ; i++) moji_disp (i, 20, i-193) ;
    for (i = 217 ; i < 224 ; i++) moji_disp (i, 36, i-213) ;
    for (i = 161 ; i < 166 ; i++) moji_disp (i, 36, i-150) ;
    for (i = 166 ; i < 177 ; i++) moji_disp (i, 52, i-162) ;
}

```

```

    return (1) ;
}

put_kanji (unsigned char *c, int x, int y, int l, int col)
    /* (x, y) の位置に col 色で文字列 c を表示 l = 0 : 全角 l = 1 : 半角 */
{
    int i, j, k, h, w;
    unsigned int code;
    unsigned char f [34] ;

    h = 0;
    if (l) w = 2; else w = 1 ;          /* w = 2 半角 w = 1 全角 */
    while (c [h])
        {code = 0x100*c [h]+c [h+1] ;
         h+= 2;
         getfont (code, f) ;
         for (i = 0 ; i <16 ; i++) for (j = 0 ; j <2 ; j++) for (k = 0 ; k <8 ; k++)
             if (((f [2*i+j+2] <<k) &0x80) == 0x80) putpixel (x+(8*j+k)/w,
                 y+i, col) ;
         x+= (16/w) +2*l;
         if (x> 640) {x = 0 ; y+= 8 ;}
        }
}

result_level_disp ()
{
    inp k, x, y;

    cleardevice () ;
    clrscr () ;
    k = 1. level;
    gotoxy (10, 2) ; cprintf ("%s さんのコース %d の結果 ¥n¥n" , name, k+1) ;
    gotoxy (1, 4) ;
    if (!d [k] . d. count) printf ("このコースはまだ終了していません。") ;
    else {
        if (!f_course) printf ("途中で終わりました。¥n¥n 前回までの") ;
        printf ("このコースの練習回数は %d 回で時間は合計" , d [k] . d. count) ;

        y =d [k] . d. t_time;
        x = y/60;
        if (x> 59) printf ("%d 時間 %d 分 %d 秒です。 ¥n¥n" , x/60, x%60, y-x*60) ;
            else printf ("%d 分 %d 秒です。 ¥n¥n" , x, y-x*60);

        printf ("最終練習は 19%d 年 %d 月 %d 日です。" ,
            d [k] . d. year, d [k] . d. month, d [k] . d. day) ;
        x = d [k] .d.l_time;
        printf ("時間は %d 分 %d 秒です。 ¥n¥n" , x/60, x%60) ;
    }
}

```

計算機数学入門とタイピングプログラム

```

x = d [k] . d. n_correct;
y = d [k] . d. n_error;
printf ("総打数 %d で正しい打数の割合は %d パーセントです。¥n¥n",
        x+y, 100L*x/(x+y));
y = d [k] . d. l_time;
if (y) printf ("ストロークの速さは毎分 %ld です。¥n¥n", 60L*x/y);
if (inc_level ()) printf ("次のレベルへ進めます。");
    else printf ("もう少しこのレベルを練習しましょう");
    }
}

int result_title ()
{
    unsigned char scan, inner, shift;
    char c [10] [7] = {" 戻る ", " 速度 ", " 正解率", " 打数 ", " キー別",
                    " 解説 ", " 総合表", "          ", " 印刷 ", "          "};

    function_key_disp (c);
    textcursor (NODISP_CURSOR);
    do {
        key (&scan, &inner, &shift);
        if (scan == 0x62) return (1);      /*f. 1*/
        if (scan == 0x63) return (2);      /*f. 2*/
        if (scan == 0x64) return (3);      /*f. 3*/
        if (scan == 0x65) return (4);      /*f. 4*/
        if (scan == 0x66) return (5);      /*f. 5*/
        if (scan == 0x67) return (6);      /*f. 6*/
        if (scan == 0x68) return (7);      /*f. 7*/
        if (scan == 0x6a) return (9);      /*f. 9*/
    } while (1);
}

function_key_disp (char c [10] [7])
{
    int i;

    textattr (0xe5);
    gotoxy (4, 25);
    for (i = 0; i < 10; i++)
        {cprintf ("%s", c [i]);
         if (i == 4) printf (" "); else printf (" ");
        }
}

course ()
{
    int n;

```

```

do {
    clrscr ();
    gotoxy (10, 2) ;
    printf ("コース番号 (1から %d) を入力してください", l. max+1) ;
    gotoxy (10, 4) ; my_scan (&n) ;
    if ((n > 0) && (n < l. max+2))
        {l. level = n-1;
         data_read (1) ;
         f_course = 1;
         return;
        }
    if (n == 0) return;
} while (1) ;
}
my_scanf (int *n)
{
    int i, l;

    l = 0;
    textattr (0xe1) ;           /*putch は <conio. h> */
    do {
        i = getch () ;
        if (i == 13) *n = 1 ;
        if ((i > 47) && (i < 58)) {putch (i);
                                l = l*10+i-48;
                                }
        if ((i == 8) && (l > 0)) {printf ("%c %c" , i, i) ;
                                l = l/10;
                                }
    } while (i != 13) ;           /*BS key*/
}

init (int fp)
{
    int i, j;

    clrscr () ;
    for (i = 0 ; i < 40 ; i++) name [i] = 0;
    gotoxy (10, 2) ;
    printf ("名前を入力してください。終わりは CR キー") ;
    j = 0;
    do {
        gotoxy (10+j, 4) ;
        i = getch () ;
        if (i == 13) {name [j] = 0 ; break;}
        if ((i == 8) && (j > 0))
            {printf ("%c %c" , i, i);

```


計算機数学入門とタイピングプログラム

```

        j--;
        continue;
    }
    if (i < 32) continue;
    name [j++] = i;
    putchar (i) ;
    if (((i > 0x80) && (i < 0xa0)) || ((i > 0xdf) && (i < 0xf0)))
        {i = getch () ;
         name [j++] = i;
         putchar (i) ;
         } /* 漢字の処理 */
    if (j > 37) {name [j] = 0; break;}
} while (i != 13) ;
lseek (fp, 0L, 0) ; write (fp, name, 40) ;
}

data_read (int m)
{
    int i, fp;
    long lg;
    char c [2] ;

    if ((fp = open ("TEXT¥¥RESULT. DAT" , O_RDWR | O_BINARY)) == -1)
        result_make (&fp) ;
    if (!m) {
        read (fp, name, 40) ;
        if (!name [0])init (fp) ;
        lseek (fp, 40L, 0) ;
        read (fp, c, 2) ;
        l.level = c [0] ;
        l.max = c [1] ;
        for (i = 0 ; i < 30 ; i++) read (fp, d [i] . c, 20) ;
    }
    lg = 642 + l.level * 500;
    lseek (fp, lg, 0) ;
    read (fp, l.n_c, 250);
    read (fp, l.n_e, 250) ;
    close (fp);
}

data_write ()
{
    int fp, m;
    long lg;
    char c [2] ;
    TIME_INFO time_info;

```

```

c [0] = m = l. level;
c [1] = l. max;
d [m] . d. n_type = d [m] . d. n_type+l. n_cc+l. n_ee;
d [m] . d. n_correct = l. n_cc; d [m] . d. n_error = l. n_ee;
d [m] . d. l_time = l. level_time;          /* コースの時間 */

if (inc_level ()) {c [0]++;                /* 次のレベルにする */
                  if (c [0]>c [1]) c [1]=c [0]; /* 最大レベルを変更 */
                  }

time_info. cmmd = 0;
bios98time (&time_info);                  /* 現在の時刻を読む */
d [m] . d. year = (time_info. year/16) *10+(time_info. year%16);
d [m] . d. month = time_info. month_week/16;
d [m] . d. day = (time_info. day/16) * 10+(time_info. day%16);
d [m] . d. t_time+=l.level_time;          /* 合計時間 */
d [m] . d. count++;                       /* 練習回数 */

if ((fp = open ("TEXTYYRESULT. DAT" , O_RDWR | O_BINARY)) == -1)
    {clrscr ();
      cleardevice ();
      printf ("RESULT. DAT がありません");
      exit (0);
    }
lseek (fp, 40L, 0);
    _write (fp, c, 2);
lg = 42+20*m;
    lseek (fp, lg, 0);
    _write (fp, d [m] . c, 20);
lg = 642+m*500;
    lseek (fp, lg, 0);
    _write (fp, l. n_c, 250);
    _write (fp, l. n_e,250);
close (fp);
}

result_make (int *fp)
{
    int i;
    char c [100];

    for (i = 0;i <100;i++) c [i] = 0;
    if ((*fp = open ("TEXTYYRESULT. DAT" , O_CREAT | O_TRUNC |
                    O_BINARY, S_IREAD | S_IWRITE)) == -1)
        {clrscr ();
          cleardevice ();

```

計算機数学入門とタイピングプログラム

```

        printf ("RESULT. DAT がありません");
        exit (0);
    }
    for (i = 0; i < 156; i++) write (*fp, c, 100);
    c [97] = 0x0d;
    c [98] = 0x0a;
    c [99] = 0x1a;
    write (*fp, c, 100);
    close (*fp);
    if ((*fp = open ("TEXT¥¥RESULT. DAT", O_RDWR | O_BINARY)) == -1)
        {clrscr ();
        cleardevice ();
        printf ("RESULT. DAT がありません");
        exit (0);
        }
}
file_disp (unsigned char *c, int *f_lend)                /* 文章を表示する */
{
    int i, x;
    gotoxy (10, 4); printf ("%80s", "");
    gotoxy (10, 6); printf ("%80s", "");
    gotoxy (10, 4);
    i = 0;
    while ((x = c [*f_lend]) != 0x0d)
        {if (x == 0x1a) return;
        if (x == 0x1f) {
            textattr (0xe5);                /* 反転表示 */
            putch (0x1f);                /* 下向きの矢印 */
            c [( *f_lend) ++] = 0x0d;
            return;
        }
        /* CR キー表示 */
        if (((x > 0x80) && (x < 0xa0)) || ((x > 0xdf) && (x < 0xf0)))
            {
                printf ("%c%c", x, c [( *f_lend) + 1]);
                ( *f_lend) += 2;
                i++;
            }
            /* 漢字処理 */
        else printf ("%c", c [( *f_lend) ++]);
        if ((++i) > 60) return;
        }
}
int read_file (int l, unsigned char *c)                /* 文章を入力する */
{
    int j, fp;
    char t [15];

    l++;

```

```

strcpy (t, "TEXT¥¥TEXT.");
t [10] = 48;
t [11] = 1/10+48;
t [12] = 1%10+48;
t [13] = 0;
if ((fp = open (t, O_RDONLY | O_BINARY)) == -1)
    {clrscr ();
     cleardevice ();
     printf ("TEXT. ??? がありません");
     exit (0);
    }
j = read (fp, c, 3000);
close (fp);
return (j);
}
read_keyboard ()
{
    int i, fp;
    char c [1062];

    setvisualpage (0);
    setactivepage (0);
    cleardevice ();
    setactivepage (1);
    cleardevice ();

    if ((fp = open ("KEY_DISP. DAT", O_RDONLY | O_BINARY)) == -1)
        {clrscr ();
         printf ("KEY_DISP. DAT がありません。");
         exit (0);
        }
    for (i = 0; i < 75; i++) {
        read (fp, c, 1062);
        putimage (60, i*4, c, 0);
    }

    close (fp);
    if ((fp = open ("FINGER. DAT", O_RDONLY | O_BINARY)) == -1)
        {clrscr ();
         printf ("FINGER. DAT がありません。");
         exit (0);
        }
    read (fp, f_g, 800);
    read (fp, f_w, 800);
    close (fp);
    clrscr ();
    keyboard_copy (1, 0);
    disp_waku (1);
}

```

```

}
keyboard_copy (int l, int m)
{
    int i;
    char c [1310] ;
    for (i = 0;i > 30;i++)
        {setactivepage (1) ;
         getimage (60, 150+i*5, c) ;
         setactivepage (m) ;
         putimage (60, 150+i*5, c, 0);
        }
    setactivepage (0) ;
}
disp_waku (int k)
{
    int i;
    setfillstyle (1, 5) ;
    for (i = 0;i < 4;i++) bar (60, 5+i*32, 580, i*32+10) ;
    bar (60, 5, 65, 106) ;
    bar (575, 5, 580, 106) ;
    bar (480, 5, 485, 40) ;
    if (k == 1) for (i = 1;i < 11;i++) finger (i, 7) ;
}
finger (int l, int col)
/* l = 1-10 指を指定 col = 色 */
{
    char c [10] = {40, 55, 60, 55, 30, 30, 55, 60, 55, 40} ; /* 指の長さ */
    int d [10] = {124, 148, 172, 196, 220, 268, 292, 316, 340, 364} ;
/* 指の位置 */

    if (col == 2) {
        f_g [2] = c [l-1] ;
        putimage (d [l-1] , 370-c [l-1] , f_g, 0) ;
    }
    else {
        f_w [2] = c [l-1] ;
        putimage (d [l-1] , 370-c [l-1] , f_w, 0) ;
    }
}
int key_shift (unsigned char *shift)
{
    KEY_INFO Ky;
    ky. cmmd = 2;
    *shift = bios98key (&ky) ;
}
int key (unsigned char *scan, unsigned char *inner, unsigned char *shift)
{
    int k;
    static unsigned char b_shift;
    KEY_INFO ky;

```

```
do {
    ky. cmmd = 1;
    k = bios98key (&ky) ;
    *scan = ky. scan_code;
    *inner = ky. inter_code;
    if (!k) *scan = 0xff;
    ky. cmmd = 2;
    *shift = bios98key (&ky) ;
    if (b_shift != *shift)
        {b_shift = *shift;
         if (k) return (1) ; else return (0) ;
        }
    if (k) {
        ky. cmmd = 0;
        bios98key (&ky) ;
        return (1) ;
    }
} while (!k);
}
finish ()
{
    cleardevice () ;
    clrscr () ;
    exit (0) ;
}
```