
補足資料

人工知能パートナーシステム(AIPS)を支える

デジタル回路の世界

(ISBN978-4-88359-339-2)

著者名 萩原良昭 (hagihara-yoshiaki@aiplab.com)

<http://www.aiplab.com>

発行所名 青山社 (info@seizansha.co.jp)

<http://www.seizansha.co.jp>

●Appendix(0-1-2) P進法数をQ進法数に変換する。

この Program (a.c) はP進法数をQ進法数に変換します。まず自分のパソコンの環境下で実行 File(a.exe) を生成してください。そして実行してみましょう。まず、以下の Option の値を入力して、その後に各 Option の値に対応するP進法数を入力してください。

Type 0 to stop Program

Type 1 to Binary to Deci System

Type 2 to Decimal to BCD System

Type 3 to Binary to Hex System

Type 4 to Deci to Binary System

Type 5 to Hex to Binary System

Type 6 to P to 2 to 10 System

まず自分のパソコン環境で実行 File(a.exe) を生成してください。

●演習問題(0-1-2)

この Program (a.c) は入力数字のけた数に限界があります。大きなけた数を入力すると Error となります。何けた迄正常な値が出るか調べてみましょう。さらに Program を改良して、100けたでも、1000けたでも計算できるように変更しましょう。

以下に実際に実行した場合の Output File (B.txt)の例とその C 言語のソース Program File(a.c) の例を示します。この program はまだまだ不完全で効率が悪く改善の余地がたくさん残っています。この計算アルゴリズムを理解し、ソース Program を改良し、使いこなしてください。

0-1-2(B.txt)

Type 0 to stop Program
Type 1 to Binary to Deci System
Type 2 to Decimal to BCD System
Type 3 to Binary to Hex System
Type 4 to Deci to Binary System
Type 5 to Hex to Binary System
Type 6 to P to 2 to 10 System

Option = 1

***** Binary to Decimal *****

Give Binary Number = 100111

Decimal Number = 39

Option = 2

***** Decimal to BCD *****(HEX to BDH)*****

Input = 124578

BCD Code = 0001 0010 0100 0101 0111 1000

Option = 2

***** Decimal to BCD *****(HEX to BDH)*****

Input = AAABB99

BCD Code = 1010 1010 1010 1011 1011 1001 1001

Option = 3

***** Binary to Hex *****

Give Binary Number = 100111001101

Hexadecimal Number = 9CD

Option = 4

***** Decimal to Binary *****

Give Decimal Number = 2345689

m=21 Binary Number = 1000111100101011011001

Option = 5

***** Hexa to Binary *****

Give Hexadecimal Number = FFAA99A

m=27 Binary Number = 111111110101010100110011010

Option = 6

Give Number System p = 13

Input Number = ABC991B

na = 7

+ B x 1
+ 1 x 13
+ 9 x 169
+ 9 x 2197
+ C x 28561
+ B x 371293
+ A x 4826809

Decimal Number = 52716363

0-1-2 (B. txt)
Binary Number = 11001001000110001101001011

Option = 0

```

/*****
このProgramはまずOptionの値を聞いてきます。
*****/
Type 0 to stop Program
Type 1 to Binary to Deci System
Type 2 to Decimal to BCD System
Type 3 to Binary to Hex System
Type 4 to Deci to Binary System
Type 5 to Hex to Binary System
Type 6 to P to 2 to 10 System
*****/

#include <stdio.h>

FILE *fpB; char *BBB="B.txt";

/***** Option 1 *****/

int btodeci(void) { int n=1, i, m, k, Deci, D[200], B[200]; char j;

    printf("\n\n***** Binary to Decimal *****\n\n");
    fprintf(fpB, "\n\n***** Binary to Decimal *****\n\n");

    printf(" Give Binary Number = ");
    fprintf(fpB, " Give Binary Number = ");

    MOREC: j=getchar(); fprintf(fpB, "%c", j);

    if ( j == '1' ) goto NEXTSTEP1;
    if ( j == '0' ) goto NEXTSTEP0;goto FINALD;

    NEXTSTEP0: D[n]=0;goto NEXTSTEP;
    NEXTSTEP1: D[n]=1;
    NEXTSTEP: n=n+1;goto MOREC;

    FINALD: m=n-2; for (i=0; i<=m; i++) B[m-i]=D[i+1];Deci=B[0];k=1;

    for (i=1; i<=m; i++) { k=2*k;Deci=Deci+k*B[i]; }

    printf(" \n Decimal Number = %d", Deci);
    fprintf(fpB, "\n Decimal Number = %d", Deci);

    printf(" \n\n*****\n\n");
    fprintf(fpB, "\n\n*****\n\n");

    return 0 ;}

/***** Option 2 *****/

int decitoBCD(void) { int i, n=0; char c, cc[100], XXX;

```

0-1-2(a.c)

```

printf(      "\n\n***** Decimal to BCD ****( HEX to BDH )*****\n\n");
fprintf(fpB, "\n\n***** Decimal to BCD ****( HEX to BDH )*****\n\n");

printf(      "\n Input D = ");

```

```

MORE: c = getchar();

```

```

if ( c == '0' ) goto GETC;
if ( c == '1' ) goto GETC;
if ( c == '2' ) goto GETC;
if ( c == '3' ) goto GETC;
if ( c == '4' ) goto GETC;
if ( c == '5' ) goto GETC;
if ( c == '6' ) goto GETC;
if ( c == '7' ) goto GETC;
if ( c == '8' ) goto GETC;
if ( c == '9' ) goto GETC;
if ( c == 'A' ) goto GETC;
if ( c == 'B' ) goto GETC;
if ( c == 'C' ) goto GETC;
if ( c == 'D' ) goto GETC;
if ( c == 'E' ) goto GETC;
if ( c == 'F' ) goto GETC;

```

```

goto NEXT1;

```

```

GETC: n=n+1;cc[n]=c; goto MORE;

```

```

NEXT1: printf(      "\n\n Input = ");
       fprintf(fpB, "\n\n Input = ");

```

```

       for (i=0;i<n;i++) { printf(      "%c", cc[i+1]);
                          fprintf(fpB, "%c", cc[i+1]); }

```

```

       printf(      "\n\n BCD Code = ");

```

```

       fprintf(fpB, "\n\n BCD Code = ");

```

```

for (i=0;i<n;i++) {

```

```

if ( cc[i+1] == '0' ) { printf("0000 ");fprintf(fpB, "0000 ");}
if ( cc[i+1] == '1' ) { printf("0001 ");fprintf(fpB, "0001 ");}
if ( cc[i+1] == '2' ) { printf("0010 ");fprintf(fpB, "0010 ");}
if ( cc[i+1] == '3' ) { printf("0011 ");fprintf(fpB, "0011 ");}
if ( cc[i+1] == '4' ) { printf("0100 ");fprintf(fpB, "0100 ");}
if ( cc[i+1] == '5' ) { printf("0101 ");fprintf(fpB, "0101 ");}
if ( cc[i+1] == '6' ) { printf("0110 ");fprintf(fpB, "0110 ");}
if ( cc[i+1] == '7' ) { printf("0111 ");fprintf(fpB, "0111 ");}
if ( cc[i+1] == '8' ) { printf("1000 ");fprintf(fpB, "1000 ");}
if ( cc[i+1] == '9' ) { printf("1001 ");fprintf(fpB, "1001 ");}
if ( cc[i+1] == 'A' ) { printf("1010 ");fprintf(fpB, "1010 ");}
if ( cc[i+1] == 'B' ) { printf("1011 ");fprintf(fpB, "1011 ");}
if ( cc[i+1] == 'C' ) { printf("1100 ");fprintf(fpB, "1100 ");}
if ( cc[i+1] == 'D' ) { printf("1101 ");fprintf(fpB, "1101 ");}

```

```

                                0-1-2(a.c)
if ( cc[i+1] == 'E' ) { printf("1110 ");fprintf(fpB, "1110 ");}
if ( cc[i+1] == 'F' ) { printf("1111 ");fprintf(fpB, "1111 ");}

printf("%n%n "); fprintf(fpB, "%n%n ");

XXX=getchar(); if (XXX == 'S') goto FINAL;

FINAL:

    printf(    "%n%n*****\n");
    fprintf(fpB, "%n%n*****\n");

return 0;}

```

/***** Option 3 *****/

```

int btohex(void) { int n, i, m, h, H[200], B[200];char j;

    printf(    "%n%n***** Binary to Hex *****\n");
    fprintf(fpB, "%n%n***** Binary to Hex *****\n");

n=1;

    printf(    "    Give Binary Number = ");
    fprintf(fpB, "    Give Binary Number = ");

MOREC: j=getchar();fprintf(fpB, "%c", j);

if ( j == '1' ) goto NEXTSTEP1;if ( j == '0' ) goto NEXTSTEP0;goto FINALD;

NEXTSTEP0: H[n]=0;goto NEXTSTEP;
NEXTSTEP1: H[n]=1;
NEXTSTEP: n=n+1;goto MOREC;

FINALD: m=n-2;for (i=0; i<=m; i++) B[m-i]=H[i+1];
B[m+1]=0;B[m+2]=0;B[m+3]=0;h=m/4;

    printf(    "%n%n    Hexadecimal Number = ");
    fprintf(fpB, "%n%n    Hexadecimal Number = ");

for (i=0; i<=h; i++) H[i]=B[i*4+3]*8+B[i*4+2]*4+B[i*4+1]*2+B[i*4];

for (i=0; i<=h; i++) {

if( H[h-i]==0 ) { printf("0");fprintf(fpB, "0");}
if( H[h-i]==1 ) { printf("1");fprintf(fpB, "1");}
if( H[h-i]==2 ) { printf("2");fprintf(fpB, "2");}
if( H[h-i]==3 ) { printf("3");fprintf(fpB, "3");}
if( H[h-i]==4 ) { printf("4");fprintf(fpB, "4");}
if( H[h-i]==5 ) { printf("5");fprintf(fpB, "5");}
if( H[h-i]==6 ) { printf("6");fprintf(fpB, "6");}
if( H[h-i]==7 ) { printf("7");fprintf(fpB, "7");}
if( H[h-i]==8 ) { printf("8");fprintf(fpB, "8");}
if( H[h-i]==9 ) { printf("9");fprintf(fpB, "9");}

```

```

                                0-1-2 (a. c)
if ( H[h-i]==10 ) { printf("A");fprintf(fpB, "A");}
if ( H[h-i]==11 ) { printf("B");fprintf(fpB, "B");}
if ( H[h-i]==12 ) { printf("C");fprintf(fpB, "C");}
if ( H[h-i]==13 ) { printf("D");fprintf(fpB, "D");}
if ( H[h-i]==14 ) { printf("E");fprintf(fpB, "E");}
if ( H[h-i]==15 ) { printf("F");fprintf(fpB, "F");}

printf(      "\n\n*****\n\n");
fprintf(fpB, "\n\n*****\n\n");
return 0 ;}

```

/****** Option 4 *****/

```

int DecitoBinary(void) {
int n, nn, i, m, k, A[25], C[25], D[25], E[25], B[200];char j;

printf(      "\n\n***** Decimal to Binary *****\n\n");
fprintf(fpB, "\n\n***** Decimal to Binary *****\n\n");

n=1;

printf(      "\n Give Decimal Number = ");
fprintf(fpB, "\n Give Decimal Number = ");

MOREC: j=getchar();fprintf(fpB, "%c", j);

if ( j == '1' ) goto NEXTSTEP1; if ( j == '2' ) goto NEXTSTEP2;
if ( j == '3' ) goto NEXTSTEP3; if ( j == '4' ) goto NEXTSTEP4;
if ( j == '5' ) goto NEXTSTEP5; if ( j == '6' ) goto NEXTSTEP6;
if ( j == '7' ) goto NEXTSTEP7; if ( j == '8' ) goto NEXTSTEP8;
if ( j == '9' ) goto NEXTSTEP9; if ( j == '0' ) goto NEXTSTEP0;

goto FINALD;

NEXTSTEP0: D[n]=0;goto NEXTSTEP;
NEXTSTEP1: D[n]=1;goto NEXTSTEP;
NEXTSTEP2: D[n]=2;goto NEXTSTEP;
NEXTSTEP3: D[n]=3;goto NEXTSTEP;
NEXTSTEP4: D[n]=4;goto NEXTSTEP;
NEXTSTEP5: D[n]=5;goto NEXTSTEP;
NEXTSTEP6: D[n]=6;goto NEXTSTEP;
NEXTSTEP7: D[n]=7;goto NEXTSTEP;
NEXTSTEP8: D[n]=8;goto NEXTSTEP;
NEXTSTEP9: D[n]=9;NEXTSTEP:n=n+1;goto MOREC;

FINALD:n=n-1;m=-1; MNEXT:E[1]=D[1];

for (i=1;i<n;i++) {C[i]=E[i]/2;A[i]=E[i]-2*C[i];E[i+1]=10*A[i]+D[i+1];}
                C[n]=E[n]/2;A[n]=E[n]-2*C[n];m=m+1;B[m]=A[n];

```

0-1-2 (a. c)

```

k=0; for (i=1;i<=n;i++) { D[i]=C[i];if(D[i]!=0) k=1; }
    if(k==1) goto MNEXT;
FINAL: printf(    "%n m=%d Binary Number = ",m);
    fprintf(fpB,"%n m=%d Binary Number = ",m);

    for (i=0;i<=m;i++) { printf("%d",B[m-i]);fprintf(fpB,"%d",B[m-i]);}

    printf(    "%n\n*****\n");
fprintf(fpB,"%n\n*****\n");

return 0 ;}

```

/***** Option 5 *****/

```

int HextoBinary(void) { int n, i, m, k, D[25], B[200];char j;

    printf(    "%n\n***** Hexa to Binary *****\n");
fprintf(fpB,"%n\n***** Hexa to Binary *****\n");

n=1;

    printf(    " Give Hexadecimal Number = ");
fprintf(fpB," Give Hexadecimal Number = ");

MOREC: j=getchar();fprintf(fpB,"%c",j);

if ( j == 'A' ) goto NEXTSTEPA; if ( j == 'B' ) goto NEXTSTEPB;
if ( j == 'C' ) goto NEXTSTEP3; if ( j == 'D' ) goto NEXTSTEPD;
if ( j == 'E' ) goto NEXTSTEP5; if ( j == 'F' ) goto NEXTSTEPF;
if ( j == '1' ) goto NEXTSTEP1; if ( j == '2' ) goto NEXTSTEP2;
if ( j == '3' ) goto NEXTSTEP3; if ( j == '4' ) goto NEXTSTEP4;
if ( j == '5' ) goto NEXTSTEP5; if ( j == '6' ) goto NEXTSTEP6;
if ( j == '7' ) goto NEXTSTEP7; if ( j == '8' ) goto NEXTSTEP8;
if ( j == '9' ) goto NEXTSTEP9; if ( j == '0' ) goto NEXTSTEP0;goto FINALD;

NEXTSTEPA: D[n]=10;B[4*n-1]=0;B[4*n-2]=1;B[4*n-3]=0;B[4*n-4]=1;goto NEXTSTEP;
NEXTSTEPB: D[n]=11;B[4*n-1]=1;B[4*n-2]=1;B[4*n-3]=0;B[4*n-4]=1;goto NEXTSTEP;
NEXTSTEP3: D[n]=12;B[4*n-1]=0;B[4*n-2]=0;B[4*n-3]=1;B[4*n-4]=1;goto NEXTSTEP;
NEXTSTEPD: D[n]=13;B[4*n-1]=1;B[4*n-2]=0;B[4*n-3]=1;B[4*n-4]=1;goto NEXTSTEP;
NEXTSTEP5: D[n]=14;B[4*n-1]=0;B[4*n-2]=1;B[4*n-3]=1;B[4*n-4]=1;goto NEXTSTEP;
NEXTSTEPF: D[n]=15;B[4*n-1]=1;B[4*n-2]=1;B[4*n-3]=1;B[4*n-4]=1;goto NEXTSTEP;
NEXTSTEP0: D[n]=0;B[4*n-1]=0;B[4*n-2]=0;B[4*n-3]=0;B[4*n-4]=0;goto NEXTSTEP;
NEXTSTEP1: D[n]=1;B[4*n-1]=1;B[4*n-2]=0;B[4*n-3]=0;B[4*n-4]=0;goto NEXTSTEP;
NEXTSTEP2: D[n]=2;B[4*n-1]=0;B[4*n-2]=1;B[4*n-3]=0;B[4*n-4]=0;goto NEXTSTEP;
NEXTSTEP3: D[n]=3;B[4*n-1]=1;B[4*n-2]=1;B[4*n-3]=0;B[4*n-4]=0;goto NEXTSTEP;
NEXTSTEP4: D[n]=4;B[4*n-1]=0;B[4*n-2]=0;B[4*n-3]=1;B[4*n-4]=0;goto NEXTSTEP;
NEXTSTEP5: D[n]=5;B[4*n-1]=1;B[4*n-2]=0;B[4*n-3]=1;B[4*n-4]=0;goto NEXTSTEP;

```

0-1-2 (a. c)

```
NEXTSTEP6: D[n]=6;B[4*n-1]=0;B[4*n-2]=1;B[4*n-3]=1;B[4*n-4]=0;goto NEXTSTEP;  
NEXTSTEP7: D[n]=7;B[4*n-1]=1;B[4*n-2]=1;B[4*n-3]=1;B[4*n-4]=0;goto NEXTSTEP;  
NEXTSTEP8: D[n]=8;B[4*n-1]=0;B[4*n-2]=0;B[4*n-3]=0;B[4*n-4]=1;goto NEXTSTEP;  
NEXTSTEP9: D[n]=9;B[4*n-1]=1;B[4*n-2]=0;B[4*n-3]=0;B[4*n-4]=1;
```

```
NEXTSTEP:n=n+1;goto MOREC;
```

```
FINALD:n=n-1;m=n*4-1;CHECK:if (B[0]!=0) goto FINAL;  
m=m-1;for (i=0;i<=m;i++) B[i]=B[i+1];goto CHECK;
```

```
FINAL: printf(    "%n m=%d    Binary    Number = ",m);  
        fprintf(fpB,"%n m=%d    Binary    Number = ",m);  
        for (i=0;i<=m;i++) { printf("%d",B[i]);fprintf(fpB,"%d",B[i]);}  
  
        printf(    "%n\n*****\n");  
        fprintf(fpB,"%n\n*****\n");  
  
return 0 ;}
```

/***** Option 6 *****/

```
int  pto2to10(void) {  
int  p, i, na, n=-1, d=0, dd, ddd, s=1, bb[100];char  a, aa[100], XXX;  
  
        printf(    "%n\n*****\n");  
        fprintf(fpB,"%n\n*****\n");  
  
        printf(    "%n Give    Number System p = ");  
        fprintf(fpB,"%n Give    Number System p = ");  
  
        scanf("%d",&p); fprintf(fpB,"%d",p);  
  
        printf("%n          Input Number ? = ");  
  
        MOREA: a = getchar(); if( a == '?' ) goto NEXTA;  
                n=n+1;aa[n]=a; goto MOREA;  
  
        NEXTA: printf(    "%n\n          Input Number = ");  
                fprintf(fpB,"%n\n          Input Number = ");  
  
                for (i=0;i<n;i++) { printf("%c",aa[i+1]); fprintf(fpB,"%c",aa[i+1]);}  
  
                na=n;
```

0-1-2 (a. c)

```

printf(      "\n\n          na = %d\n", na);
fprintf(fpB, "\n\n          na = %d\n", na);

for (i=0; i<na; i++) {

if ( aa[na-i]== '1' ) d=d+s;   if ( aa[na-i]== '2' ) d=d+s*2;
if ( aa[na-i]== '3' ) d=d+s*3; if ( aa[na-i]== '4' ) d=d+s*4;
if ( aa[na-i]== '5' ) d=d+s*5; if ( aa[na-i]== '6' ) d=d+s*6;
if ( aa[na-i]== '7' ) d=d+s*7; if ( aa[na-i]== '8' ) d=d+s*8;
if ( aa[na-i]== '9' ) d=d+s*9; if ( aa[na-i]== 'A' ) d=d+s*10;
if ( aa[na-i]== 'B' ) d=d+s*11; if ( aa[na-i]== 'C' ) d=d+s*12;
if ( aa[na-i]== 'D' ) d=d+s*13; if ( aa[na-i]== 'E' ) d=d+s*14;
if ( aa[na-i]== 'F' ) d=d+s*15;

printf(      "\n          + %c x %d ", aa[na-i], s);
fprintf(fpB, "\n          + %c x %d ", aa[na-i], s);

s=s*p;}

printf(      "\n\n Decimal Number = %d \n", d);
fprintf(fpB, "\n\n Decimal Number = %d \n", d);

i=1; dd=d; MOREB: ddd=dd/2; bb[i]=dd-2*ddd;

if(ddd == 0) goto NEXT; dd=ddd; i=i+1; goto MOREB;

NEXT:n=i;

printf(      "\n Binary Number = ");
fprintf(fpB, "\n Binary Number = ");

for ( i=0; i<n; i++) { printf("%d", bb[n-i]); fprintf(fpB, "%d", bb[n-i]);}

printf("\n"); fprintf(fpB, "\n");

printf(      "\n\n*****\n");
fprintf(fpB, "\n\n*****\n");

XXX=getchar();

if(XXX == 'S') return 0;return 0;}

int main(void) { char c, d, XXX;

fpB=fopen(BBB, "w");

printf("\n\n *****\n ");

```

0-1-2 (a. c)

```

fprintf(fpB, "\n\n *****\n\n ");

printf("\n Type 0 to stop Program ");
printf("\n Type 1 to Binary to Deci System ");
printf("\n Type 2 to Decimal to BCD System ");
printf("\n Type 3 to Binary to Hex System ");
printf("\n Type 4 to Deci to Binary System ");
printf("\n Type 5 to Hex to Binary System ");
printf("\n Type 6 to P to 2 to 10 System ");

fprintf(fpB, "\n Type 0 to stop Program ");
fprintf(fpB, "\n Type 1 to Binary to Deci System ");
fprintf(fpB, "\n Type 2 to Decimal to BCD System ");
fprintf(fpB, "\n Type 3 to Binary to Hex System ");
fprintf(fpB, "\n Type 4 to Deci to Binary System ");
fprintf(fpB, "\n Type 5 to Hex to Binary System ");
fprintf(fpB, "\n Type 6 to P to 2 to 10 System ");

printf("\n\n *****\n\n ");
fprintf(fpB, "\n\n *****\n\n ");

MORE:

printf("\n\n Option = ");
fprintf(fpB, "\n\n Option = ");

c = getchar(); d = getchar();

fprintf(fpB, "%c", c);

if ( d == '0' ) goto FINAL;
if ( c == '0' ) goto FINAL;

if ( c == '1' ) btodeci();
if ( c == '2' ) decitoBCD();
if ( c == '3' ) btohex();
if ( c == '4' ) DecitoBinary();
if ( c == '5' ) HextoBinary();
if ( c == '6' ) pto2to10();

goto MORE;

FINAL: fclose(fpB);return 0; }

```