

Basicの基礎・文字列処理

0. 目次

1 0. 文字列処理

1 0. 1 文字の識別

- (1) 文字列を1文字ずつに分解せよ。
- (2) 英字、数字、その他の文字が混じった文字列で、英大文字、英小文字、数字、その他の文字を識別せよ。
- (3) 英字、数字、その他の記号(, . = 等)文字、全角文字(1 a A 漢)が混じった文字列で、英大文字、英小文字、数字、その他の記号文字、全角文字を識別せよ。

1 0. 2 文字列の順序

1. 文字の識別

(1) 文字列を1文字ずつに分解せよ。

考え方1: Mid\$関数を使い、1文字ごとに分解する。

半角文字 (1, a, A, =)、全角文字 (1, a, A, 漢)、それぞれ1文字として扱われる。

関数	機能
Mid\$	Mid\$(S\$, m, n) 変数S\$のm文字目からn文字分を返す。

●プログラム (K1011a. bas)

```

1  ' << K1011a. bas >>
2  ' 文字列を1文字に分解
3  '
4  Do
5  ' 文字列の読み込み。
6  Read S$
7  If S$ = "" Then Exit Do
8  ' 確認のため実行画面に表示。
9  Print"データ:";S$
10 '
11 ' 文字列から1文字ずつ取り出す。
12 For I=1 To LEN(S$)
13 ' T$: 取り出した1文字。
14   T$=Mid$(S$, I, 1)
15   Print"|";T$;"|"
16 Next I
17 Loop
18 End
19 '
20 ' データ。
21 Data "abc","漢字",""
```

実行結果

```

データ : abc
|a|
|b|
|c|
データ : 漢字
|漢|
|字|
OK
```

考え方2：Left\$関数とRight\$関数を使い、先頭の1文字と残りの文字列に分解する。

関数	機能
Left\$	Left\$(S\$, n) 変数S\$の左端からn文字分を返す。
Right\$	Right\$(S\$, n) 変数S\$の右端からn文字分を返す。

●プログラム (K1011b. bas)

```

1  ' << K1011b. bas >>
2  ' 文字列を1文字に分解
3  '
4  Do
5  ' 文字列の読み込み。
6  Read S$
7  If S$ = "" Then Exit Do
8  ' 確認のため実行画面に表示。
9  Print"データ:";S$
10 '
11 ' 文字列から1文字ごと取り出す。
12 While S$ <> ""
13 ' T$: 取り出した1文字。
14 T$=Left$(S$,1)
15 Print"|";T$;"|"
16 S$=Right$(S$,Len(S$)-1)
17 Wend
18 Loop
19 End
20 '
21 ' データ。
22 Data "abc","漢字",""
```

実行結果

```

データ : abc
|a|
|b|
|c|
データ : 漢字
|漢|
|字|
OK
```

- (2) 英字、数字、その他の記号(, . = 等)文字が混じった文字列で、英大文字、英小文字、数字、その他の記号文字を識別せよ。

文字や記号はコンピュータの内部では対応する特定の番号(コード)で表され、アスキーコードという。

英大文字(A-Z)が番号(65-90)、英小文字(a-z)が番号(97-122)、数字(0-9)が番号(48-57)に対応する。

番号(32~127)が文字に対応し、番号(0~31)は種々のコントロールコードに対応し文字ではない。

関数	機能
Asc	Asc(X\$) 文字列X\$の先頭1文字のアスキーコードを返す。

●文字とアスキーコードの対応

```

1 Print "アスキーコード(32~127)と文字の対応"
2 For I=32 to 120 Step 8
3   For J=0 to 7
4     Print Using"####";I+J;
5     Print " ";Chr$(I+J);
6   Next J
7   Print
8 Next I
9 End

```

実行結果

```

アスキーコード(32~127)と文字の対応
 32   33 !   34 "   35 #   36 $   37 %   38 &   39 '
 40 (   41 )   42 *   43 +   44 ,   45 -   46 .   47 /
 48 0   49 1   50 2   51 3   52 4   53 5   54 6   55 7
 56 8   57 9   58 :   59 ;   60 <   61 =   62 >   63 ?
 64 @   65 A   66 B   67 C   68 D   69 E   70 F   71 G
 72 H   73 I   74 J   75 K   76 L   77 M   78 N   79 O
 80 P   81 Q   82 R   83 S   84 T   85 U   86 V   87 W
 88 X   89 Y   90 Z   91 [   92 ¥   93 ]   94 ^   95 _
 96 `   97 a   98 b   99 c 100 d 101 e 102 f 103 g
104 h 105 i 106 j 107 k 108 l 109 m 110 n 111 o
112 p 113 q 114 r 115 s 116 t 117 u 118 v 119 w
120 x 121 y 122 z 123 { 124 | 125 } 126 ~ 127
OK

```

考え方1：Mid\$関数を使い、1文字ごとに分解し、アスキーコードで分類する。

●プログラム (K1012a. bas)

```

1  ' << K1012a. bas >>
2  ' 文字の識別
3  '
4  Do
5  ' 文字列の読み込み。
6  Read S$
7  If S$ = "" Then Exit Do
8  ' 確認のため実行画面に表示。
9  Print"データ：";S$
10 '
11 ' 文字列から1文字ごと取り出す。
12 For I=1 To LEN(S$)
13 ' T$: 取り出した1文字。
14 T$=Mid$(S$, I, 1)
15 Select Case Asc(T$)
16 ' その他の記号。
17 Case 0 To 47
18   Print T$;" その他の記号"
19 ' 数字。
20 Case 48 To 57
21   Print T$;" 数字"
22 ' その他の記号。
23 Case 58 To 64
24   Print T$;" その他の記号"
25 ' 英大文字。
26 Case 65 To 90
27   Print T$;" 英大文字"
28 ' その他の記号。
29 Case 91 To 96
30   Print T$;" その他の記号"
31 ' 英小文字。
32 Case 97 To 122
33   Print T$;" 英小文字"
34 ' その他の記号。
35 Case 123 To 127
36   Print T$;" その他の記号"
37 End Select
38 Next I
39 Loop
40 End
41 '
42 ' データ。
43 Data "a1B,2=c 45", ""

```

実行結果

```

データ : a1B,2=c 45
a 英小文字
1 数字
B 英大文字
, その他の記号
2 数字
= その他の記号
c 英小文字
  その他の記号
4 数字
5 数字
OK

```

考え方2 : Left\$関数とRight\$関数を使い、先頭の1文字と残りの文字列に分解する。

●プログラム (K1012b. bas)

```

1  ' << K1012b. bas >>
2  ' 文字の識別
3  '
4  Do
5  ' 文字列の読み込み。
6  Read S$
7  If S$ = "" Then Exit Do
8  ' 確認のため実行画面に表示。
9  Print"データ : ";S$
10 '
11 ' 文字列から1文字ごと取り出す。
12 While S$ <> ""
13   ' T$: 取り出した1文字。
14   T$=Left$(S$,1)
15   Select Case Asc(T$)
16     ' その他の記号。
17     Case 0 To 47
18       Print T$;" その他の記号"
19     ' 数字。
20     Case 48 To 57
21       Print T$;" 数字"
22     ' その他の記号。
23     Case 58 To 64
24       Print T$;" その他の記号"
25     ' 英大文字。
26     Case 65 To 90
27       Print T$;" 英大文字"
28     ' その他の記号。
29     Case 91 To 96

```

```
30     Print T$;" その他の記号"  
31     ' 英小文字。  
32     Case 97 To 122  
33     Print T$;" 英小文字"  
34     ' その他の記号。  
35     Case 123 To 127  
36     Print T$;" その他の記号"  
37     End Select  
38     S$=Right$(S$, Len(S$)-1)  
39     Wend  
40 Loop  
41 End  
42 '  
43 ' データ。  
44 Data "a1B,2=c 45", ""
```

実行結果

```
データ : a1B,2=c 45  
a 英小文字  
1 数字  
B 英大文字  
, その他の記号  
2 数字  
= その他の記号  
c 英小文字  
  その他の記号  
4 数字  
5 数字  
OK
```

- (3) 英字、数字、その他の記号(、. = 等)文字、全角文字(1 a A 漢)が混じった文字列で、英大文字、英小文字、数字、その他の記号文字、全角文字を識別せよ。

コンピュータ内では、文字は特定の番号(コード)で表されるが、漢字等を扱おうとすると、アスキーコードでは番号が足りず処理できない。

そこで、言語に係わらず、すべての文字に独立した番号(コード)を与えるユニコード(Unicode)が開発された。番号Nの範囲は、 $0 \leq N < 65535$ 。

関数	機能
AscU	AscU(x\$) 文字列x\$の先頭のUnicode文字の番号を返す。 0~127は普通のアスキーコードと一致する。
ChrU	ChrU(X) 番号X($0 \leq X < 65535$)に対応する文字を返す。

●ユニコードと文字

```

1 Print "ユニコードと文字の対応"
2 '
3 Print"漢";" ";AscU("漢")
4 Print"字";" ";AscU("字")
5 For I=28450 to 28458
6   For J =0 to 7
7     Print Using"#####";I+J;
8     Print " ";ChrU(I+J);
9   Next J
10  Print
11 Next I
12 Print
13 '
14 For I=32 to 120 Step 8
15   For J=0 to 7
16     Print Using"#####";I+J;
17     Print " ";ChrU(I+J);
18   Next J
19   Print
20 Next I
21 End

```

実行結果

```

漢  28450
字  23383
28450 漢 28451 漣 28452 漣 28453 漣 28454 漣 28455 漣 28456 漣 28457 漣
28451 漣 28452 漣 28453 漣 28454 漣 28455 漣 28456 漣 28457 漣 28458 漣
28452 漣 28453 漣 28454 漣 28455 漣 28456 漣 28457 漣 28458 漣 28459 漣
28453 漣 28454 漣 28455 漣 28456 漣 28457 漣 28458 漣 28459 漣 28460 漣
28454 漣 28455 漣 28456 漣 28457 漣 28458 漣 28459 漣 28460 漣 28461 漣
28455 漣 28456 漣 28457 漣 28458 漣 28459 漣 28460 漣 28461 漣 28462 漣
28456 漣 28457 漣 28458 漣 28459 漣 28460 漣 28461 漣 28462 漣 28463 漣

```



```

28457 漩 28458 漪 28459 漫 28460 瀆 28461 滌 28462 濂 28463 灑 28464 瀾
28458 漪 28459 漫 28460 瀆 28461 滌 28462 濂 28463 灑 28464 瀾 28465 漱

```

```

32      33 !    34 "    35 #    36 $    37 %    38 &    39 '
40 (    41 )    42 *    43 +    44 ,    45 -    46 .    47 /
48 0    49 1    50 2    51 3    52 4    53 5    54 6    55 7
56 8    57 9    58 :    59 ;    60 <    61 =    62 >    63 ?
64 @    65 A    66 B    67 C    68 D    69 E    70 F    71 G
72 H    73 I    74 J    75 K    76 L    77 M    78 N    79 O
80 P    81 Q    82 R    83 S    84 T    85 U    86 V    87 W
88 X    89 Y    90 Z    91 [    92 ¥    93 ]    94 ^    95 _
96 `    97 a    98 b    99 c    100 d   101 e   102 f   103 g
104 h   105 i   106 j   107 k   108 l   109 m   110 n   111 o
112 p   113 q   114 r   115 s   116 t   117 u   118 v   119 w
120 x   121 y   122 z   123 {   124 |   125 }   126 ~   127

```

OK

考え方1：Mid\$関数を使い、1文字ごとに分解し、ユニコードで分類する。

●プログラム (K1013a. bas)

```

1  ' << K1013a. bas >>
2  ' 文字の識別
3  '
4  Do
5  ' 文字列の読み込み。
6  Read S$
7  If S$ = "" Then Exit Do
8  ' 確認のため実行画面に表示。
9  Print"データ：" ; S$
10 '
11 ' 文字列から1文字ごと取り出す。
12 For I=1 To LEN(S$)
13   ' T$：取り出した1文字。
14   T$=Mid$(S$, I, 1)
15   Select Case AscU(T$)
16     ' その他の記号。
17     Case 0 To 47
18       Print T$;" その他の記号"
19     ' 数字。
20     Case 48 To 57
21       Print T$;" 数字"
22     ' その他の記号。
23     Case 58 To 64
24       Print T$;" その他の記号"
25     ' 英大文字。
26     Case 65 To 90
27       Print T$;" 英大文字"
28     ' その他の記号。
29     Case 91 To 96
30       Print T$;" その他の記号"
31     ' 英小文字。
32     Case 97 To 122
33       Print T$;" 英小文字"

```

```

34     ' その他の記号。
35     Case 123 To 127
36         Print T$;" その他の記号"
37
38         Case Is >= 128
39             Print T$;" 漢字"
40     End Select
41 Next I
42 Loop
43 End
44 '
45 ' データ。
46 Data "a1B,2=c 45日本語", ""

```

実行結果

```

データ : a1B,2=c 45日本語
a 英小文字
1 数字
B 英大文字
, その他の記号
2 数字
= その他の記号
c 英小文字
  その他の記号
4 数字
5 数字
日 漢字
本 漢字
語 漢字
OK

```

考え方2 : Left\$関数とRight\$関数を使い、先頭の1文字と残りの文字列に分解する。

●プログラム (K1013b. bas)

```

1  ' << K1013b. bas >>
2  ' 文字の識別
3  '
4  Do
5  ' 文字列の読み込み。
6  Read S$
7  If S$ = "" Then Exit Do
8  ' 確認のため実行画面に表示。
9  Print"データ : ";S$
10 '
11 ' 文字列から1文字ごと取り出す。
12 While S$ <> ""
13     ' T$: 取り出した1文字。
14     T$=Left$(S$, 1)

```

```

15 Select Case AscU(T$)
16   ' その他の記号。
17   Case 0 To 47
18     Print T$;" その他の記号"
19     ' 数字。
20     Case 48 To 57
21       Print T$;" 数字"
22       ' その他の記号。
23       Case 58 To 64
24         Print T$;" その他の記号"
25         ' 英大文字。
26         Case 65 To 90
27           Print T$;" 英大文字"
28           ' その他の記号。
29           Case 91 To 96
30             Print T$;" その他の記号"
31             ' 英小文字。
32             Case 97 To 122
33               Print T$;" 英小文字"
34               ' その他の記号。
35               Case 123 To 127
36                 Print T$;" その他の記号"
37                 Case Is >= 128
38                   Print T$;" 漢字"
39   End Select
40   S$=Right$(S$, Len(S$)-1)
41 Wend
42 Loop
43 End
44 '
45 ' データ。
46 Data "a1B,2=c 45日本語", ""

```

実行結果

```

データ : a1B,2=c 45日本語
a 英小文字
1 数字
B 英大文字
, その他の記号
2 数字
= その他の記号
c 英小文字
  その他の記号
4 数字
5 数字
日 漢字
本 漢字
語 漢字
OK

```

2. 文字列の順序

アルファベットからなる2つの文字列A\$とB\$について、辞書の中でA\$がB\$より先に現れる時、 A < B$$ または B > A$$ と書く。

●辞書式順序

- ・文字列pの長さmと文字列qの長さnが同じ場合 ($m=n$)

文字列 $p=p(1)p(2)\cdots p(m)$ と $q=q(1)q(2)\cdots q(m)$ において、
 $p(1) = q(1), \dots, p(i-1) = q(i-1), p(i) < q(i)$ ($1 \leq i < m$)のとき、
 文字列pは文字列qより辞書式順序で小さいといい、 $p < q$ と書く。
 $p(i) = q(i)$ ($1 \leq i \leq m$)のとき、文字列pと文字列qは等しいといい、
 $p=q$ と書く。

- ・文字列pの長さmと文字列qの長さnが異なる場合 ($m < n$)

文字列 $p=p(1)p(2)\cdots p(m)$ と $q=q(1)q(2)\cdots q(n)$ において、
 $p(1) = q(1), \dots, p(i-1) = q(i-1), p(i) < q(i)$ ($1 \leq i < m < n$)のとき、
 文字列pは文字列qより辞書式順序で小さいといい、 $p < q$ と書く。
 $p(i) = q(i)$ ($1 \leq i \leq m < n$)のとき、文字列pは文字列qより辞書式順序で
 小さいといい、 $p < q$ と書く。

考え方：辞書式順序の定義通り、文字列A\$, B\$の先頭の文字から比較していく。

●プログラム (K1021. bas)

```

1  ' << K1021. bas >>
2  ' 文字列の辞書式順序判定
3  '
4  Do
5  ' 文字列A$, B$の読み込み。
6  Read A$, B$
7  If (A$ = "") and (B$ = "") Then Exit Do
8  ' 確認のため実行画面に表示。
9  Print "文字列A$"; "|"; A$; "| "
10 Print "文字列B$"; "|"; B$; "| "
11 '
12 ' 短い方の長さを求める。
13 L=Min(Len(A$), Len(B$))
14 ' 文字列A$, B$から1文字ずつ取り出す。
15 For I=1 To L
16 ' A: A$から取り出した文字のアスキーコード。
17 ' B: B$から取り出した文字のアスキーコード。
18 A=Asc(Mid$(A$, I, 1))
19 B=Asc(Mid$(B$, I, 1))
20 If A < B Then
21   Print A$; "<"; B$: Exit For

```

```

22     End If
23     If A > B Then
24         Print A$;">";B$: Exit For
25     End If
26 Next I
27 '
28 ' 長さLまでの比較で判定できなかった場合。
29 If I > L Then
30     If Len(A$) > L Then Print A$;">";B$
31     If Len(B$) > L Then Print A$;"<";B$
32     If Len(A$) = Len(B$) Then Print A$;"=";B$
33 End If
34 Print
35 Loop
36 End
37 '
38 ' データ。
39 Data "abc","abc", "abc","abcd", "abc","aaa", "", ""

```

実行結果

```

文字列A$|abc|
文字列B$|abc|
abc=abc

文字列A$|abc|
文字列B$|abcd|
abc<abcd

文字列A$|abc|
文字列B$|aaa|
abc>aaa

OK

```