

1 文字検査/変換マクロ 220 頁

1.1 学習のポイント

文字の種類を調べたり、文字の変換をする関数について学びます。

is で始まる文字検査マクロと、to で始まる文字変換マクロが ctype.h で定義されています。これらは関数ではなくマクロとして定義されています。

is ~ (文字検査マクロ)

書式

is ~ (c)

機能

文字 c が ~ で記述される検査項目を満たせば真 (1)、満たさなければ偽 (0) を返します。これらのマクロを使用する際は必ず ctype.h を取り込まなければなりません。

to ~ (文字変換マクロ)

書式

to ~ (c)

機能

文字 c に ~ で示された変換をします。

これらのマクロを使用する際は必ず ctype.h を取り込まなければなりません。

1.2 例題 49 reidai49.c

入力した文字列の小文字は大文字に、大文字は小文字に変換しなさい。

```
/*
  入力した文字列の小文字は大文字に、大文字は小文字に変換しなさい。
  reidai49.c
*/

#include <stdio.h>
```

```

#include <string.h>
#include <ctype.h>

int main()
{
    int i;
    char s[20];
    char t[20];

    printf("moji ? "); scanf("%s", s);

    for(i = 0; i <= strlen(s); i++){
        if(isupper(s[i]))
            t[i] = tolower(s[i]);
        else if(islower(s[i]))
            t[i] = toupper(s[i]);
        else
            t[i] = s[i];
    }

    printf("%s ---> %s\n", s, t);

    return 0;
}

```

1.3 練習問題 49 rensyu49.c

入力した文字列中の文字の種類を調べて、各個数をカウントする。調べる文字の種類はアルファベット、数字、その他の3種類とする。

```

/*
練習問題 49
入力した文字列中の文字の種類を調べて、各個数をカウントする。調べる文字の
種類は アルファベット、数字、その他の3種類とする。
rensyu49.c
*/

#include <stdio.h>
#include <string.h>

```

```

#include <ctype.h>

int main()
{
    int i;
    int alpha = 0;
    int digit = 0;
    int other = 0;

    char s[20];

    printf("moji ? "); scanf("%s", s);

    for(i = 0; i < strlen(s); i++){
        if(isalpha(s[i]))
            alpha++;
        else if(isdigit(s[i]))
            digit++;
        else
            other++;
    }

    printf("alpha=%d\n", alpha);
    printf("digit=%d\n", digit);
    printf("other=%d\n", other);

    return 0;
}

```

2 AI による大規模データ処理入門

2.1 言語処理アルゴリズム

自然言語で記述された大規模データから、その特徴を抽出する方法を考えます。人工知能の歴史においては、自然言語処理の研究は、機械翻訳や自動要約などへの応用を念頭に置いて、さまざまな側面から取り組まれてきました。

2.2 単語レベルでの特徴抽出

n-gram による処理では、文書の自然言語としての特徴は全く利用していませんが、それでも文書のある特徴を表現することが可能です。そこで次に、文書が単語の連なりで記述されているという自然言語としての特徴を使った解析方法を考えます。

日本語の文書では、単語と単語の区切りは明確ではありません。このため、文から単語を切り出す形態素解析には一般に複雑な処理を行いません。これに対して英語では、単語と単語は空白で区切られていますから、文から単語を取り出すのは容易です。そこで、空白を手掛かりに英文から単語を取り出す処理を考えます。(AI による大規模データ処理入門 215 頁)

2.3 単語を抽出するプログラム getword.c

```
/*
*****
/* データ前処理プログラム      */
/*      getword.c              */
/* 単語の取り出し              */
*****

#include <stdio.h>

/*
*****
/*      main() 関数          */
*****
int main()
{
    char chr ; /*入力文字*/

    while((chr=getchar())!=EOF){
        /*アルファベット以外なら改行*/
        if(('A'<=chr) && (chr<='z'))
            putchar(chr); /*一文字出力*/
        else
            putchar('\n') ; /*改行*/
    }
}
```

```
    return 0 ;  
}
```

2.4 getword の出力例

```
Alice  
was  
beginning  
to  
get  
very  
tired  
of  
sitting  
by  
her  
sister  
on  
the  
bank  
  
and  
of  
having  
nothing
```

2.5 getword.exe の使い方

```
./getword < alice.txt
```

```
./getword < alice.txt | sort | uniq -c | sort -rg
```