

```
/******  
プログラム名: a2AA.c  
英字の大文字を小文字、または、その逆に変換するプログラム。  
ただし、英字以外の文字は変換しない。  
*****/  
/******  
2019/08/20 修正  
2011/11/07 2012年度の実験用プログラムの試作  
*****/  
/******  
コマンドライン引数(command-line argument)を利用して、  
英字の大文字を小文字またはその逆変換を行うプログラム  
  
コマンドライン引数(command-line argument)  
  コマンド名と同時に指定するコマンドのための実引数で、  
  main関数に文字列として渡される。  
  
具体的には、ポインタ配列の形式でプログラムに渡される。  
パラメータの文字列は、main関数のパラメータによって  
プログラムに取り込むことができる。  
*****/  
  
#include <stdio.h> /* printf 関数宣言 */  
#include <stdlib.h> /* exit,atoi 関数宣言 */  
  
void main(int argc, char *argv[]); /* プロトタイプ宣言 *//* メイン関数 */  
void readwrite(char *fpr, char *fpw); /* プロトタイプ宣言 *//* 入力ファイルから読み込み、  
出力ファイルに書き出す */  
int convert(int a); /* プロトタイプ宣言 *//* 英字の大文字を小文字に変換する、  
または、その逆を行う */  
  
/* main関数 */  
void main(int argc, char *argv[]){ /* int argc と char *argv[] はセットで使う */  
    /* int argc は入力の個数(入力パラメータ + 1 個),  
    char *argv[] は入力パラメータの文字列が  
    格納される配列のポインタ情報 */  
  
    if(argc == 3){ /* 入力の個数が 3 ならば true */  
  
        readwrite(argv[1], argv[2]); /* argv[1] 入力ファイル名が書かれた配列*/  
        /* argv[2] 出力ファイル名が書かれた配列*/  
  
    }else{ /* 入力の個数が 3 以外の場合 */  
  
        printf(" usage : %s InputFILE OutputFILE\n", argv[0]);  
  
    }  
}  
  
/* readwrite関数 *//* 入力ファイル fpr から読み込み、出力ファイル fpw に書き出す */  
void readwrite(char *fpr, char *fpw){  
    FILE *ifp, *ofp;  
    int M,C;  
  
    /* file open *//* fopen関数: 標準入出力を利用せずに、ファイルを読み書きする */  
    /* fopen関数で読み書きの準備をし、fclose関数で開いたファイルをクローズする */  
    /* fopen関数はファイルポインタを返す */  
    /* オープンしたファイルは、必ずクローズの処理をしないと書き込みは 完了しない */  
  
    if((ifp = fopen(fpr, "rb")) == NULL) exit(-1); /* 入力ファイル名 fpr のデータの読み込み、  
    バイナリーモードでオープンする */  
    /* ifp はファイル名 fpr のデータを指すポインタ */  
  
    if((ofp = fopen(fpw, "wb")) == NULL) exit(-1); /* 出力ファイル名 fpw のデータの書き込み、  
    バイナリーモードでオープンする */  
    /* ofp はファイル名 fpw のデータを指すポインタ */  
  
    while(( M = fgetc(ifp) ) != EOF){ /* fgetc(ifp) はファイルポインタ ifp で指定されたファイルの  
    現在位置から1バイト読み込む。  
    読み込んだ1バイトを変数 M に代入する。  
    このとき、fgetc(ifp) は整数型である。すなわち、M は整数型である。  
    つまり、入力ファイルから読み込んだ平文 ( 1バイト) を M とする。  
    ファイルの最後(EOF(End Of File)まで読み込みを実行する。*/  
  
        C=convert(M); /* 変数 M の値 (整数型)を変数 C に代入する (単純に代入しているだけ) */  
  
        fputc(C, ofp); /* fputc(C, ofp)関数: 整数 C で与えられた1バイトをファイルポインタ ofp で  
        指定されたファイルの現在位置に出力する (書き込む) */  
    }  
  
    /* file close */  
    fclose(ifp); /* オープンした入力ファイルをクローズする */
```

```
fclose(ofp); /* オープンした出力ファイルをクローズする */
}

/* convert関数 */ /* 英字の大文字を小文字に変換する、または、その逆を行う */
int convert(int a)
{
    int b;

    if ( 65 <= a && a <= 90){ /* 大文字ならば 32 を加えて 小文字に変換 */

        b=a+32;

    }
    else{

        if ( 97 <= a && a <= 122){ /* 小文字ならば 32 を引いて 大文字に変換 */
            b=a-32;
        }
        else{ /* 大文字でも小文字でもないならば、同じ値 a を返す */
            b=a;
        }
    }

    return b;
}

/*****
コンパイル方法:
$ gcc -o a2AA.out a2AA.c
```

```
実行方法:
$ cat uecoo <--- catコマンドでファイの中身を確認
The University of Electro Communications
$ ./a2AA.out uecoo goo <--- 入力ファイル uecoo、 出力ファイルgoo
$ cat goo
tHE uNIVERSITY OF eLECTRO cOMMUNICATIONS
$
*****/
```