

コマンドラインから受け取った文字列の大文字と小文字を変換するプログラムを作成せよ。入力は1バイトの表示文字とし、アルファベット文字以外は変換しない。

<ソースプログラム> argcv.c

```
1  #include <stdio.h>
2
3  int get_n(char *);
4  void print_data(char *,int);
5  void replace(char *dest, char *str);
6
7  int main(int argc, char *argv[]){
8      int i,j;
9      printf("文字列の数 = %d\n",argc);
10     printf("文字列のアドレス = %08x\n\n",&argc);
11
12     for(i=1,argv++;*argv !=NULL;i++,argv++){
13         printf("No.%2d\t%s\n",i,*argv);
14         printf("[%s] ==変換==>[",*argv);
15         replace(*argv,*argv);
16         puts("]");
17         j=get_n(*argv);
18         puts("-----");
19         print_data(*argv,j);
20         puts("-----");
21     }
22     return(0);
23 }
24
25 void replace(char *dest, char *str){
26     int i;
27     for(i=0; str[i]!= NULL; i++){
28         if(isalpha(str[i])){
29             {
30                 if(islower(str[i]))
31                     dest[i] = toupper(str[i]);
32                 else
33                     dest[i] = tolower(str[i]);
34             }
35         }else{
36             dest = (str);
37             printf("%c",dest[i]);
38         }
39         printf("%c",dest[i]);
40     }}
41
42 int get_n(char *pa){
```

```

43  int i=0;
44  for (i=1;*pa !=NULL;i++,pa++);
45  return(i);
46  }
47
48  void print_data(char *pa,int n){
49  printf("文字列の文字数=%2d %s\n",n,pa);
50  }

```

<実行結果> argcv01

```
nw0945:kadai6 e095745$ ./argv01 You Are King Of King
```

文字列の数 = 6

文字列のアドレス = bfffffa20

No. 1 You

[You] ==変換==>[yOU]

文字列の文字数= 4 yOU

No. 2 Are

[Are] ==変換==>[aRE]

文字列の文字数= 4 aRE

No. 3 King

[King] ==変換==>[kING]

文字列の文字数= 5 kING

No. 4 Of

[Of] ==変換==>[oF]

文字列の文字数= 3 oF

No. 5 King

[King] ==変換==>[kING]

文字列の文字数= 5 kING

<解説>

main 関数のコマンドラインから受け取った文字列を replace 関数で小文字を大文字に、大文字を小文字に変換している。その後、get_n 関数で引数を返し、関数 print_data でその文字列の文字数を出力している。

<考察>

何度も bus error が出て混乱させられたが、実際に絵を描きながら配列に気をつけたり、ポインタとアドレス、特に char のポインタに気をつけながらやって、なんとか題意に沿うことができた。文字列の最後には、表示されない NULL の存在が重要な問題だと思った。

文字列を反転して表示するプログラムも作成せよ。(例 "abcd" => "dcba")

<ソースプログラム>hanten02.c

```
nw0945:kadai6 e095745$ cat -n hanten02.c
 1  #include <stdio.h>
 2
 3  int  get_n(char *);
 4  void print_data(char *,int);
 5
 6  int main(int argc, char *argv[]){
 7      int i,j;
 8
 9      for(i=1,argv++;*argv !=NULL;i++,argv++){
10          printf("No.%2d\t%s\n",i,*argv);
11          replace(*argv,*argv);
12          printf("\n");
13          j=get_n(*argv);
14          puts("-----");
15      }
16      return(0);
17  }
18
19 void replace(char *dest, char *str){
20     int i;
21     printf("反転==> ");
22     for(i=1,str++; *str!= NULL; i++,str++){ }
23     for(; i>=0; str--,i--){
24         dest = str;
25         printf("%c",*dest);
26     }
27 }
28
29 int get_n(char *pa){
30     int i=0;
31     for (i=1;*pa !=NULL;i++,pa++);
32     return(i);
33 }
```

<実行結果>

```
N-juuuun-book:kadai6 e095745$ ./hanten02 I gotta feeling
No. 1 I
反転==> I
-----
No. 2 gotta
反転==> attog
-----
```

No. 3 feeling

反転==> gnileef

N-juuuun-book:kadai6 e095745\$./hanten02 mintia COLDSMASH

No. 1 mintia

反転==> aitnim

No. 2 COLDSMASH

反転==> HSAMSDLOC

<解説>

最初に作ったプログラムを少し改良して、作成した。replace 関数で反転した文字列を作っている。

<考察>

何度も失敗したが、変更を重ねて、なんとか題意に沿うことができた。

配列を実際に絵に描いてどこに何があるかを正確につかみながら、プログラミングすることが大事なところだと思う。

<感想>

2週間も遅れて、徹夜して、なんとかできたが、思うようにいつていない。

勉強が足りないという簡単なことが解決できていない。

<参考文献>

初心者のためのポイント C 言語 <http://www9.plala.or.jp/sgwr-t/index.html>

初級 C 言語講座 <http://www.sgnet.co.jp/c/>