

[1] trupperプログラム

ライブラリ関数 `islower()`, `toupper()`を使い、`trlowup`プログラムを書き換えて`trupper`プログラムを作成せよ。

1-1. プログラム

```
-----  
/*****  
Program      : trupper.c  
Student-ID   : 045713C  
Author       : OSHIRO,Kazuya  
Date         : 04/06/15  
Comment      : toupperとislowerを使っての書き換え  
*****/  
  
#include <stdio.h>  
#include <ctype.h>  
  
char trupper(char);  
/* char型(一文字)のtrupper()関数を宣言、char型のパラメータを持つ */  
  
int main()  
{  
    char c;                               /* char型の変数cの宣言 */  
  
    while((c = getchar()) != EOF) /* cの一文字を取得、EOFでなければ繰り返す */  
        putchar(trupper(c));      /* trupper関数を呼び出し、値を出力する */  
    return(0);  
}  
  
char trupper(char c)                     /* trupper関数の定義 */  
{  
    if(islower(c) != 0)                   /* cの文字が英小文字なら真を返す */  
        return(toupper(c));              /* 英小文字を英大文字に変換して値を返す */  
    else  
        return(c);                       /* 上の条件外ならそのままの値を返す */  
}  
-----
```

1-2. 実行結果

```
[nw0413:~/prog1/report/#4] j04013% ./trupper  
abcDEF  
ABCDEF  
xyzXYZYYZZ  
XYZXYZYYZZ  
123  
123  
!  
!
```

1-3. 考察

新しく出た関数について

islower(c)

アルファベットの小文字かどうかを判断する関数。

文字cが英小文字(a~z)なら真(0以外の数)を返し、文字cが英小文字以外なら偽(0)を返す。

この関数と似たような働きを持つisupper(c)という関数があり、これは文字cが英大文字かどうかを判断する。

toupper(c)

文字cが小文字なら、大文字に変換した値を返す。

英大文字以外なら、そのままの値を返す。

似たような関数にtolower(c)があり、こちらは小文字なら大文字に変換した値を返す。

EOFについて

EOFはEnd Of Fileの意味。

EOFは記号定数でありstdio.hで-1に定義されている。

このプログラムではCtrl+z, Ctrl+c, Ctrl+dで終了した。

プログラムについて

今回のプログラムは英小文字を入力すると英大文字となって出力するプログラムです。

islower(c)で判断させた後、英小文字ならtoupper(c)により大文字で返されるようになっていきます。先生のプログラムからの変更点は今述べた所くらいで、関数をいじっただけですね。

ネットでは文字処理関数は#include<ctype.h>が必要と書いていたがそれがなくてもプログラムは問題なく動作した。

[2] rot13プログラム

trupperプログラムを書き換えて、rot13暗号化プログラム、rot13 複合化プログラムを作成せよ。rot13とは、次のようにアルファベットを13文字ずらす暗号化の方法である。

A --> N	a --> n
B --> O	b --> o
C --> P	c --> p
.
Z --> M	z --> m

2-1. プログラム

```
-----  
/*****  
Program    : rot13.c  
Student-ID : 045713C
```

Author : OSHIRO, Kazuya
Date : 04/06/15
Comment : 暗号化13ずらす
*****/

```
#include <stdio.h>

char rot(char);          /* char型のrotという関数を宣言 */

int main()
{
    char c;
    while((c = getchar()) != EOF) /* cの一文字を取得、EOFでなければ繰り返す */
        putchar(rot(c));        /* rot関数を呼び出し、値の文字を出力する */
    return(0);
}

char rot(char c)        /* rot関数の定義 */
{
    if(('a' <= c && c <= 'm') || ('A' <= c && c <= 'M'))
        return(c+=13);
    /* cがa~mもしくはA~Mのとき真のとき文字cのASCIIの10進数に13足した値を返す */
    else if(('n' <= c && c <= 'z') || ('N' <= c && c <= 'Z'))
        return(c-=13);
    /* cがn~zもしくはN~Zのとき文字cのASCIIの10進数から13を引きその値を返す */
    else
        return(c);
    /* 上の条件のどれにも当てはまらなかった場合そのまま値を返す */
}
-----
```

2-2. 実行結果

```
[nw0413:~/prog1/report/#4] j04013% ./rot13
123
123
abcABC
nopNOP
xyzXYZ
klmKLM
?!
?!
```

2-3. 考察

プログラムについて

このプログラムは入力された文字を判断し、a~m,A~MならASCIIコードを10進数であらわしたものに13を足し、n~z、N~Zなら13を引く。それにより元の文字より13ずれた文字が出力される。このプログラムは英文字を真ん中から分けて片方はプラス13、もう一方はマイナス13しているので暗号化と複合化を同時に果たしている。

t rupperを書き換えてと書いていたが作ってるうちにどちらかという元のtrlowupにちか
いものになった。

結果からわかるように英文字を入れた場合は別のものになるが記号や数字はそのまま出力された。

[3] オリジナルプログラム

オリジナルの暗号化・複合化プログラムを作成せよ。

3-1. プログラム

```
-----  
/*****  
Program    : original.c  
Student-ID : 045713C  
Author     : OSHIRO, Kazuya  
Date      : 04/06/14  
Comment   : オリジナル暗号化  
*****/  
  
#include <stdio.h>  
#include <string.h>  
#define SIZE 80                /* SIZEを80に定義する */  
char rot(char);                /* char型のrot()関数を宣言する */  
  
main()  
{  
    char s[SIZE];                /* 要素数80の配列sを宣言 */  
    int k = 0;                    /* int型のkを0に初期化 */  
  
    printf("1行入力してください、¥n");  
    printf("before  = ");        /* printfでそれぞれ出力 */  
    fgets(s, sizeof(s), stdin); /* キーボードから入力した行を配列sに読み込む */  
    printf("after  = ");  
    while(s[k] != '\0'){        /* 要素s[k]がヌル文字になるまで繰り返す */  
        printf("%c", rot(s[k])); /* 関数rot()を呼び出しその値を出力する */  
        k++;  
    }  
    /* kに1を足すこれにより要素s[k]が変化していき、最後の要素にはヌル文字が入る */  
}  
  
char rot(char n)  
{  
    if(n%5!=0 && ('0' <= n && n <= '~'))  
        return(n-=47);  
    /* nが'0'から'~'でnのASCII10進コードを5で割ったとき余りがあったらnの10進コードから47  
    を引いてその値を返す */  
    else if(' ' <= n && n <= 'N')  
        return(n+=47);  
    /* nが' 'から'N'であるときASCII10進コードから47を引いてその値を返す */  
    else  
        return(n);  
    /* それ以外の場合そのまま出力する */  
}
```

3-2. 実行結果

```
[nw0413:~/prog1/report/#4] j04013% ./original
1行入力してください、
before   = asdIoL+P!h8(93'&[[1
after    = 2sdx@{ZPP9gWhbVU,,`
[nw0413:~/prog1/report/#4] j04013% ./original
1行入力してください、
before   = 2sdx@{ZPP9gWhbVU,,`
after    = asdxoLZPPh8(93'U[[1
```

3-3. 考察

ヌル文字について

ヌル文字とは文字列終端の'0'のこと。
文字列を入力した際には最後の要素にこの文字が入る。

プログラムについて

このプログラムは入力された文字をrot()関数内でずらして暗号化するものです。
先生の見本とさしてかわらなくなっていました・・・。
このプログラムもrot13プログラムと同じように暗号化兼、複合化のプログラムにしました。
rot13と違う所は変更範囲を表示できるASCIIコードすべてにしたところ('0'~' 'で5で割ったとき0になるものはそのまま出力)ぐらいですね。あとwhile文内を文字の要素を使って終わらせたことです。

[4] 感想

今回のレポートの反省はオリジナル暗号化プログラム。はっきりいって自分で見てもダメなような気がします。初めの二つはパッパと進んでこのときにはまだ時間に余裕すら残ってたのになー。けどしかたがない、暗号化で色々試行錯誤して(例えば文字の出力を逆からやったり、ビット反転してみたり)初めて使う関数とかもあったので勉強になったということでよし！今まで一番悩んだプログラムでした。そんなこんなで今回も時間ギリギリ、次こそは余裕を持ってレポート終わらします。

参考文献

- C実践プログラミング 第3版 (Steve Oualline 著)
- C言語によるプログラミング (スーパーリファレンス編) (北川雅己)
- 初心者のためのポイント学習C言語 (<http://www9.plala.or.jp/sgwr-t/index.html>)