

プログラミング I

Report#3

提出日:2006年某日(木)

所属 :工学部情報工学科

学籍番号: 065702G

氏名 : 新垣 智規

◎問題 sample#1.c を解析し、ASCII コード(0x00～0x7f)の各範囲 (Scope)を判断するプログラムを作成せよ。

```
/*                                     Program :
while,if,if-else,T or F             Student-ID :065702G
Author                               :           ARAKAKI,Tomonori
UpDate :2006/05/19(Fri)             Comments :while
if T or F                           */
```

```
#include <stdio.h>
```

```
#define FALSE 0
```

```
#define TRUE !FALSE
```

```
int main(){
```

```
    int value,c, count;
```

```
    char line[128];
```

```
    count = 0;
```

```
    while(TRUE){
```

```
        count++;
```

```
        if(count > 5) break;
```

```
        printf("Enter a HexValue ==> ");
```

```
        fgets(line, sizeof(line), stdin);
```

```
        sscanf(line, "%x", &c);
```

```
        printf("Colum=%2d:%d(%3d)-%x(%2x)" ,count,c,c);
```

```
        if(0x20 <= c && c <= 0x7e)
```

```
            printf("-%c(%c)\n",c);
```

```
        else
```

```

printf("-Not Printable character\n");

if (0x30 <= c && c <= 0x39){
    puts("====> figures\n");
}else if(0x41 <= c && c <= 0x5a){
    puts("====> capotal letter\n");
}else if(0x61 <= c && c <= 0x7a){
    puts("====> small letter\n");
}else if(0x00 <= c && c <= 0x1f && c = 7f){
    puts("====> Not Printable character\n");
}else{
    puts("====> misc.\n");
}
}

return(0);
}

```

◎実行結果

```

Enter a HexValue ==> 31
Colum= 1:%d( 49)-%x(31)-%c(1)
====> figures

Enter a HexValue ==> 42
Colum= 2:%d( 66)-%x(42)-%c(B)
====> capotal letter

Enter a HexValue ==> 70
Colum= 3:%d(112)-%x(70)-%c(p)
====> small letter

Enter a HexValue ==> 05

```

```
Colum= 4:%d( 5)-%x( 5)-Not Printable character
```

```
=====> Not Printable character
```

```
Enter a HexValue ===> 122
```

```
Colum= 5:%d(290)-%x(122)-Not Printable character
```

```
=====> misc.
```

◎ 考察

ASCII コードの分類を判断するプログラム。

数字、英語の大小文字、非表示文字、その他という分類をし、実行し入力された 16 進数を ASCII コード表の範囲内で判断をするプログラム。

分類は課題の例に従ってそれぞれ「数字: **figures**、英大文字: **capital letter**、英小文字: **small letter**、非表示文字: **Not Printable character**、その他: **misc**」という形で作成した。

実行結果は、分類した範囲の数字を各々入力し、正確に動作しているかを確認した。

◎問題 `sample#2.c` のプログラムの動作を考察せよ。

◎ソースコード

```
/*
```

```
Program : sample#2.c
```

```
Student-ID : 065702G
```

```
Author : ARAKAKI, Tomonori
```

```
UpDate : 2006/05/21(Sun)
```

```
Comments : while & for
```

```
*/
```

```
#include <stdio.h>
```

```
#define FALSE 0
```

```
#define TRUE !FALSE
```

```
int main(){
    int count;

    count = 0;
    while(TRUE){
        count++;
        if(count > 5) break;
        printf("while-Count=%2d\n",count);
    }

    for(count=1; count<=5; count++){
        printf("for -Count=%2d\n",count);
    }

    return(0);
}
```

◎実行結果

```
while-Count= 1
while-Count= 2
while-Count= 3
while-Count= 4
while-Count= 5
for -Count= 1
for -Count= 2
for -Count= 3
for -Count= 4
for -Count= 5
```

◎考察

while&for を用いたプログラム。

while 文はループ処理を繰り返し、「count++」とあるように繰り返すごとに数字を1ずつ加えていき、count が5を超えたので「break」、つまり while の処理を終え、for の処理に移った。

for 処理もまた while 処理と同じように、「count++」で5を超えたら処理を終えた。

しかし、同じ処理をしているように見えるが、while 処理は「5以上でブレイク(処理終了)」であるのに対し、for 処理は「5以下の処理を行う(5を超えたら処理終了)」という条件で動いている。

おそらく、「sample#2.c」の実行結果は、while と for の処理結果は同じであるが、その過程の内容は真逆の事柄であると推測できる。

◎問題 sample#3.c を解析し、表示可能な文字による ASCII コード表を作成せよ。

◎ソースコード

```
/*  
Program : sample#3a.c  
Student-ID : 065702G  
Author : ARAKAKI,Tomonori  
UpDate : 2006/05/21(Sun)  
Comments :ASCII code program  
*/  
  
#include <stdio.h>  
  
int main(){  
    int c;  
  
    for(c = 0x00; c<=0x7f; c++){
```

```

    if((c % 4) == 0) printf("\n");
    printf("%x(%x)-%c(%c) | ",c,c);
}
printf("\n");

return(0);
}

```

◎実行結果

```

%x(0)-%c() | %x(1)-%c() | %x(2)-%c() | %x(3)-%c() |
%x(4)-%c() | %x(5)-%c() | %x(6)-%c() | %x(7)-%c() |
%x(8)-%c() | %x(9)-%c( ) | %x(a)-%c(
) | %x(b)-%c(
) |
%x(c)-%c(
) | %x(e)-%c() | %x(f)-%c() |
%x(10)-%c() | %x(11)-%c() | %x(12)-%c() | %x(13)-%c() |
%x(14)-%c() | %x(15)-%c() | %x(16)-%c() | %x(17)-%c() |
%x(18)-%c() | %x(19)-%c() | %x(1a)-%c() | %x(1b)-%c(
|x(1c)-%c() | %x(1d)-%c() | %x(1e)-%c() | %x(1f)-%c() |
%x(20)-%c( ) | %x(21)-%c(!) | %x(22)-%c(") | %x(23)-%c(#) |
%x(24)-%c($ ) | %x(25)-%c(%) | %x(26)-%c(&) | %x(27)-%c(') |
%x(28)-%c(( ) | %x(29)-%c()) | %x(2a)-%c(*) | %x(2b)-%c(+) |
%x(2c)-%c(, ) | %x(2d)-%c(-) | %x(2e)-%c(.) | %x(2f)-%c(/) |
%x(30)-%c(0) | %x(31)-%c(1) | %x(32)-%c(2) | %x(33)-%c(3) |
%x(34)-%c(4) | %x(35)-%c(5) | %x(36)-%c(6) | %x(37)-%c(7) |
%x(38)-%c(8) | %x(39)-%c(9) | %x(3a)-%c(:) | %x(3b)-%c(;) |
%x(3c)-%c(<) | %x(3d)-%c(=) | %x(3e)-%c(>) | %x(3f)-%c(?) |
%x(40)-%c(@) | %x(41)-%c(A) | %x(42)-%c(B) | %x(43)-%c(C) |
%x(44)-%c(D) | %x(45)-%c(E) | %x(46)-%c(F) | %x(47)-%c(G) |
%x(48)-%c(H) | %x(49)-%c(I) | %x(4a)-%c(J) | %x(4b)-%c(K) |
%x(4c)-%c(L) | %x(4d)-%c(M) | %x(4e)-%c(N) | %x(4f)-%c(O) |
%x(50)-%c(P) | %x(51)-%c(Q) | %x(52)-%c(R) | %x(53)-%c(S) |

```

```
%x(54)-%c(T) | %x(55)-%c(U) | %x(56)-%c(V) | %x(57)-%c(W) |  
%x(58)-%c(X) | %x(59)-%c(Y) | %x(5a)-%c(Z) | %x(5b)-%c([) |  
%x(5c)-%c(\) | %x(5d)-%c(]) | %x(5e)-%c(^) | %x(5f)-%c(_) |  
%x(60)-%c(`) | %x(61)-%c(a) | %x(62)-%c(b) | %x(63)-%c(c) |  
%x(64)-%c(d) | %x(65)-%c(e) | %x(66)-%c(f) | %x(67)-%c(g) |  
%x(68)-%c(h) | %x(69)-%c(i) | %x(6a)-%c(j) | %x(6b)-%c(k) |  
%x(6c)-%c(l) | %x(6d)-%c(m) | %x(6e)-%c(n) | %x(6f)-%c(o) |  
%x(70)-%c(p) | %x(71)-%c(q) | %x(72)-%c(r) | %x(73)-%c(s) |  
%x(74)-%c(t) | %x(75)-%c(u) | %x(76)-%c(v) | %x(77)-%c(w) |  
%x(78)-%c(x) | %x(79)-%c(y) | %x(7a)-%c(z) | %x(7b)-%c({} |  
%x(7c)-%c(|) | %x(7d)-%c(}) | %x(7e)-%c(~) | %x(7f)-%c( ) |
```

◎ ソースコード

```
/*
```

```
Program : sample#3b.c
```

```
Student-ID : 065702G
```

```
Author : ARAKAKI, Tomonori
```

```
UpDate : 2006/05/21(Sun)
```

```
Comments :ASCII code program
```

```
*/
```

```
#include <stdio.h>
```

```
int main(){
```

```
    int c;
```

```
    for(c = 0x21; c<=0x7e; c++){
```

```
        if((c % 4) == 0) printf("\n");
```

```
        printf("%x(%x)-%c(%c) | ", c, c);
```

```
    }
```

```
    printf("\n");
```

```
return(0);  
}
```

◎実行結果

```
%x(21)-%c(!) | %x(22)-%c(") | %x(23)-%c(#) |  
%x(24)-%c($ ) | %x(25)-%c(%) | %x(26)-%c(&) | %x(27)-%c(') |  
%x(28)-%c(C) | %x(29)-%c( ) | %x(2a)-%c(*) | %x(2b)-%c(+ ) |  
%x(2c)-%c(, ) | %x(2d)-%c(-) | %x(2e)-%c(. ) | %x(2f)-%c(/) |  
%x(30)-%c(0) | %x(31)-%c(1) | %x(32)-%c(2) | %x(33)-%c(3) |  
%x(34)-%c(4) | %x(35)-%c(5) | %x(36)-%c(6) | %x(37)-%c(7) |  
%x(38)-%c(8) | %x(39)-%c(9) | %x(3a)-%c(:) | %x(3b)-%c(; ) |  
%x(3c)-%c(<) | %x(3d)-%c(=) | %x(3e)-%c(>) | %x(3f)-%c(?) |  
%x(40)-%c(@) | %x(41)-%c(A) | %x(42)-%c(B) | %x(43)-%c(C) |  
%x(44)-%c(D) | %x(45)-%c(E) | %x(46)-%c(F) | %x(47)-%c(G) |  
%x(48)-%c(H) | %x(49)-%c(I) | %x(4a)-%c(J) | %x(4b)-%c(K) |  
%x(4c)-%c(L) | %x(4d)-%c(M) | %x(4e)-%c(N) | %x(4f)-%c(O) |  
%x(50)-%c(P) | %x(51)-%c(Q) | %x(52)-%c(R) | %x(53)-%c(S) |  
%x(54)-%c(T) | %x(55)-%c(U) | %x(56)-%c(V) | %x(57)-%c(W) |  
%x(58)-%c(X) | %x(59)-%c(Y) | %x(5a)-%c(Z) | %x(5b)-%c([) |  
%x(5c)-%c(\) | %x(5d)-%c(]) | %x(5e)-%c(^) | %x(5f)-%c(_ ) |  
%x(60)-%c(`) | %x(61)-%c(a) | %x(62)-%c(b) | %x(63)-%c(c) |  
%x(64)-%c(d) | %x(65)-%c(e) | %x(66)-%c(f) | %x(67)-%c(g) |  
%x(68)-%c(h) | %x(69)-%c(i) | %x(6a)-%c(j) | %x(6b)-%c(k) |  
%x(6c)-%c(l) | %x(6d)-%c(m) | %x(6e)-%c(n) | %x(6f)-%c(o) |  
%x(70)-%c(p) | %x(71)-%c(q) | %x(72)-%c(r) | %x(73)-%c(s) |  
%x(74)-%c(t) | %x(75)-%c(u) | %x(76)-%c(v) | %x(77)-%c(w) |  
%x(78)-%c(x) | %x(79)-%c(y) | %x(7a)-%c(z) | %x(7b)-%c({} |  
%x(7c)-%c(|) | %x(7d)-%c( ) | %x(7e)-%c(~) |
```

◎考察

ASCIIコード表を参考に、表示可能な文字表を作成。

まずはどのような文字が表示できるかを調べるために00~7fまでを表示させるプログラム (sample#3a.c) を作成。

その「sample#3a.c」を参考に、表示されている範囲 (21~7e) までをまとめてコード表化した。

◎問題 文字 (文字列では無い) の演算について考察せよ。

◎考察

文字での演算 (例では'a'-'A'など) は、ASCIIコード表を元にして演算できる。コード表での「a」は10進数で表すと97、「A」は41である。

つまり'a'-'A'=97-41=56となり、コード表の10進数の「56」を参照すると数字の8になる。

このことにより'a'-'A'=8という演算ができる。

これを利用すると'f'-'a'は102-97=5より「ENQ」という結果になる。

故に、文字の演算は、まず対象の文字を進数変換し、数字に戻して計算、その後結果を文字変換という演算方法をとると考察できる。

◎感想&泣き言

感想はただただ難しいの一言。

自分のようなプログラミング初心者にとっては、先生の説明を聞いて内容を把握し、課題を試行錯誤しながらなんとか終わらせているという状況です。

課題を終わらせるための時間が恐ろしくかかり、課題をやっているだけで空の色が変わってたりして、驚いています。

正直な本音をいうと、受験よりも勉強しています。本当です、高校時代より忙しくなってます…。高校のときの大学生活=大学は暇で遊べそうだ、というイメージはプログラミングの授業であっさりと覆った気がします。

しかしこれも自分選んだ道のためだと思うと、高校とはまた違ったやる気が出てきて不思議です。

課題が出されて、その内容を完璧に理解しながらやっている、ということは胸

を張っては言えませんが、自分は自分のできる範囲でプログラミングの授業を
がんばっていきたいと思います。