



2007年10月26日

第10回 C言語によるプログラミング (7)

情報処理論 (応用)

松山大学 経営学部

檀 裕也

<http://www.cc.matsuyama-u.ac.jp/~dan/education/application/>



前回の実習課題

- `chessboard2.c` を参考にして、模様を表示するプログラムを作成し、ソースコード `chessboard3.c` をメールで送信せよ。
 - 宛先: `dan@cc.matsuyama-u.ac.jp`
 - 件名: プログラミング課題#08



実行例

- なるべく複雑な模様になるように工夫せよ。

```
C:¥>chessboard3
```

```
C:¥>chessboard3
```

```
C:¥>chessboard3
```

```
C:¥>
```

```
C:¥>
```

```
C:¥>
```



解答例 (1)

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

```
chessboard3.c
```

```
int DisplayLine( int row )
```

```
{
```

```
    int x;
```

```
    for( x = 0; x < 8; x++ ){
```

```
        if( x % 2 == 0 && row % 2 == 0
```

```
            || x % 4 == 2 || row % 4 == 2 ){
```

```
            printf( "  " );
```

```
        }
```

```
    else{
```

```
        printf( "  " );
```

```
    }
```

```
}
```

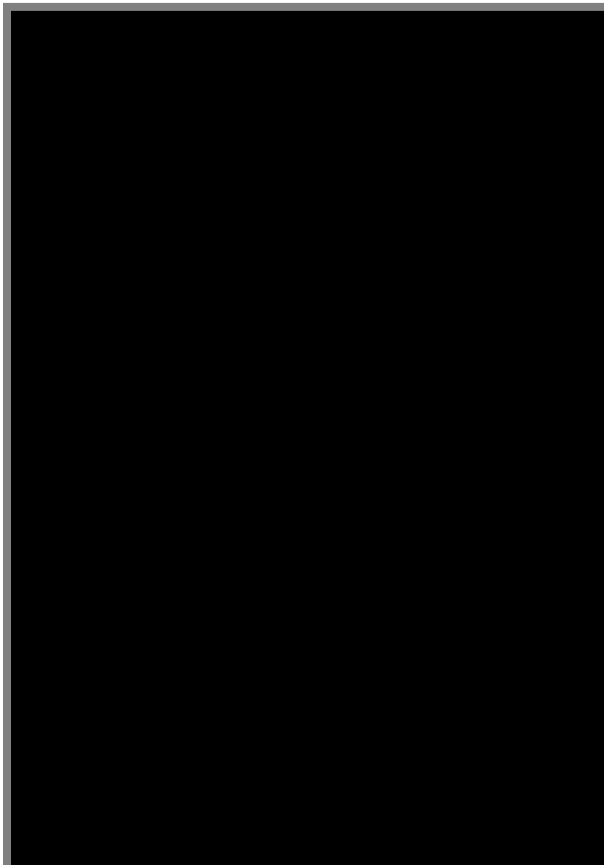


解答例 (2)

```
printf( "%n" );  
  
return 0;  
}  
  
int main()  
{  
    int i;  
  
    for( i = 0; i < 8; i++ ){  
        DisplayLine( i );  
    }  
  
    return 0;  
}
```



その他の解答



$$(2 * x + row + 5) \% 5 == 2$$

$$(x * row) \% 3 == 0$$

$$x + row == 7$$



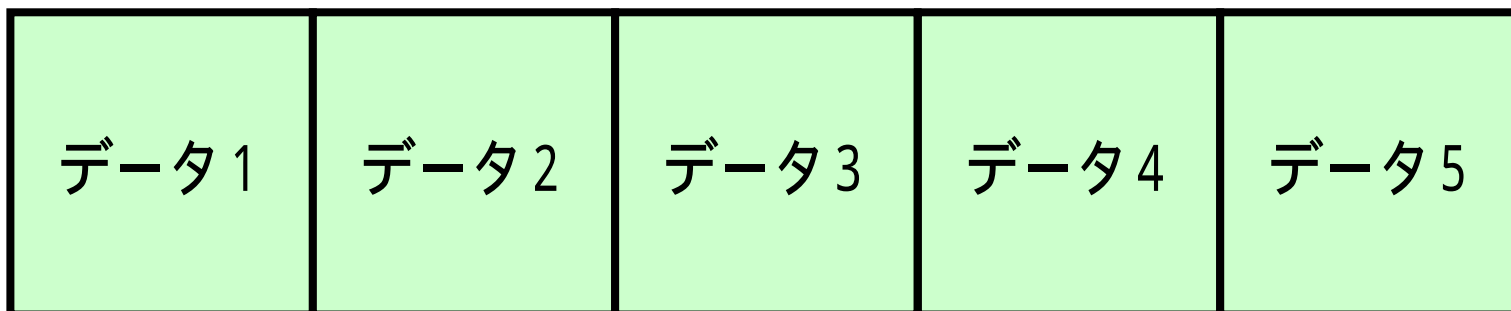
今回の授業内容

- C言語によるプログラミング (7)
 - 配列



配列

- 同じ型の複数のデータをまとめて記憶する
 - 同種のデータを扱う場合、
複数の変数を使うよりも、配列のほうが便利！

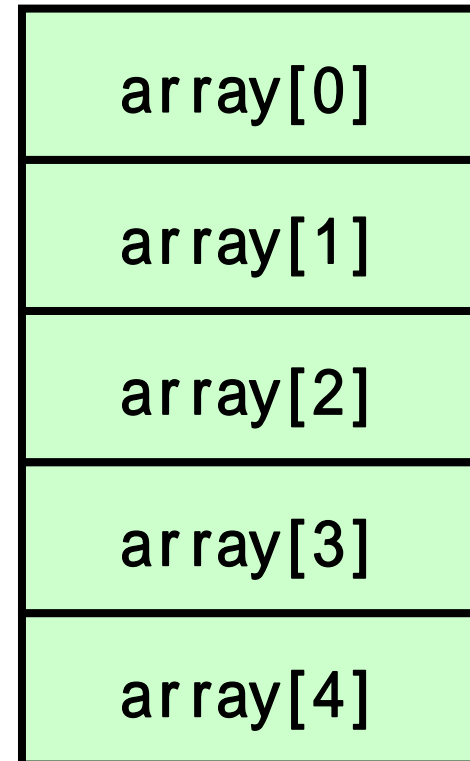




配列の宣言

- 要素の数を指定し、変数と同様に宣言する
 - 要素の番号は0から始まる
 - 変数と同様に使用できる

```
int array[5];  
  
array[0] = 100;  
scanf( "%d", &array[1] );  
  
array[2] = array[0] + array[1];  
  
printf( "%d", array[2] );
```





例題 1

reverse.c

```
#include <stdio.h>

int main()
{
    int i;
    int a[5];

    for( i = 0; i < 5; i++ ){
        printf( "%d番目の整数:", i + 1 );
        scanf( "%d", &a[i] );
    }

    for( i = 4; i >= 0; i-- ){
        printf( "%d番目のデータは%dです。¥n", i + 1, a[i] );
    }

    return 0;
}
```



実行例

- 入力した5つの整数を逆順に表示する。
 - 配列を使わずにコードを記述する場合と比べてみよ。

```
C:¥>reverse.exe
1番目の整数:5
2番目の整数:10
3番目の整数:20
4番目の整数:50
5番目の整数:100
5番目のデータは100です。
4番目のデータは50です。
3番目のデータは20です。
2番目のデータは10です。
1番目のデータは5です。

C:¥>
```



文字コード

- 半角文字には8ビットの数値（コード）が対応
 - 例えば、文字 'A' は 65 (0x41)
 - 例えば、文字 'a' は 97 (0x61)
- 16進数を表記するとき、頭に 0x をつける
 - 0x01
 - 0x0A
 - 0x0F
 - 0xFF



文字コード表 (一部)

| | | | | | | | |
|---|------|---|------|---|------|---|------|
| @ | 0x40 | P | 0x50 | ` | 0x60 | p | 0x70 |
| A | 0x41 | Q | 0x51 | a | 0x61 | q | 0x71 |
| B | 0x42 | R | 0x52 | b | 0x62 | r | 0x72 |
| C | 0x43 | S | 0x53 | c | 0x63 | s | 0x73 |
| D | 0x44 | T | 0x54 | d | 0x64 | t | 0x74 |
| E | 0x45 | U | 0x55 | e | 0x65 | u | 0x75 |
| F | 0x46 | V | 0x56 | f | 0x66 | v | 0x76 |
| G | 0x47 | W | 0x57 | g | 0x67 | w | 0x77 |
| H | 0x48 | X | 0x58 | h | 0x68 | x | 0x78 |
| I | 0x49 | Y | 0x59 | i | 0x69 | y | 0x79 |
| J | 0x4A | Z | 0x5A | j | 0x6A | z | 0x7A |
| K | 0x4B | [| 0x5B | k | 0x6B | { | 0x7B |
| L | 0x4C | ¥ | 0x5C | l | 0x6C | | 0x7C |
| M | 0x4D |] | 0x5D | m | 0x6D | } | 0x7D |
| N | 0x4E | ¥ | 0x5E | n | 0x6E | ~ | 0x7E |
| O | 0x4F | _ | 0x5F | o | 0x6F | | 0x7F |



例題 2

caps.c

```
#include <stdio.h>

int main()
{
    char x;

    do{
        printf( "アルファベット大文字:" );
        scanf( "%c", &x );
    }while( x < 0x41 || 0x5A < x );

    printf( "%c¥n", x + 0x20 );

    return 0;
}
```



実行例

- 入力した大文字を小文字に変換する。

```
C:¥>caps.exe  
アルファベット大文字:A  
a
```

```
C:¥>
```



解説

- 文字コードは8ビット整数型 char に収まる
 - int型 16 ~ 32ビット整数 (処理系に依存)
 - char型 8ビット整数
- 文字の変換仕様は %c を使う

```
char ch;  
  
ch = 0x41;  
ch++;  
  
printf( "%c", ch );
```




文字と文字列

- 文字を 'A' と表記できる (char型)
- 文字の配列で文字列を構成する
- 文字列は '¥0' で終了する

配列 `str[10]`

| | | | | | | | | | |
|---|---|---|---|---|---|----|--|--|--|
| H | e | l | l | o | ! | ¥0 | | | |
|---|---|---|---|---|---|----|--|--|--|



例題 3

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

```
string.c
```

```
int main()
```

```
{
```

```
    int i;
```

```
    char str[20];
```

```
    printf( "文字列:" );
```

```
    scanf( "%s", str );
```

```
    for( i = 0; i < 20 && str[i] != '¥0'; i++ ){
```

```
        printf( "%c¥n", str[i] );
```

```
    }
```

```
    return 0;
```

```
}
```



実行例

- 入力した文字列を文字単位に分解して縦方向に表示する。

```
C:¥>reverse.exe
```

```
文字列:Hello!
```

```
H  
e  
l  
l  
o  
!
```

```
C:¥>
```



解説

- 変換仕様
 - 整数 %d
 - 文字 %c
 - 文字列 %s



実習課題

- 入力した文字列のうち、アルファベット小文字を大文字に変換して表示するプログラム `strconvert.c` を作成せよ。
 - 宛先: dan@cc.matsuyama-u.ac.jp
 - 件名: プログラミング課題#09

```
C:¥>strconvert.exe  
文字列:Hello_world!  
HELLO_WORLD!  
  
C:¥>
```



次回の予定

- 日時： 10月31日（水）
5時限
- C言語によるプログラミング（8）
– ポインタ