

# プログラミング概論

<http://bit.ly/kosen01>

Week14@後期  
(2017/01/12)

# 本日の内容

- 講義(配列)
  - 配列の復習
  - 配列と繰り返し構文の組み合わせ
  - 定数の利用
- 演習
  - 配列とfor文を使ったプログラム

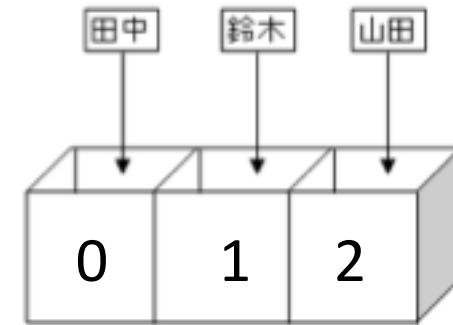
# (復習)配列の使い方

- 配列の使い方は3ステップ
  1. 配列の宣言
  2. 配列への値の代入
  3. 配列の値を取り出す

# (復習)1. 配列の宣言

- 配列の宣言

```
int person[3];
```



型名指定

(int , float, charなど)

配列名指定

要素数指定

配列を宣言するとperson[0], person[1], person[2]の3つの番号(=添字)の付いた変数が作成される  
C言語では添字は0から始まることに注意！

## (復習)2. 配列への値の代入(1/2)

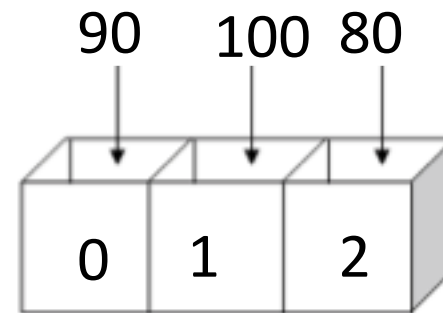
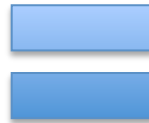
配列名[番号(=添字)] = 値;

例)

person[0] = 90;

person[1] = 100;

person[2] = 80;

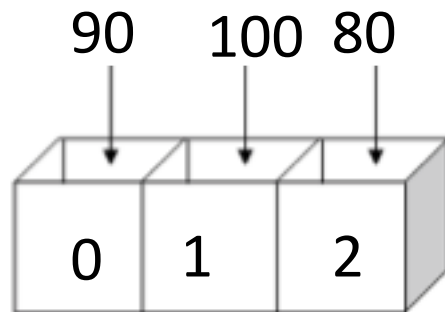


## (復習)1.2. 宣言と初期化を同時に行う

配列の宣言と初期化(初めて変数に値を代入する処理)は以下のように簡潔に記述可能

例)

```
int person[3] = {90, 100, 80};
```



## (復習)3. 配列の値を取り出す

- 例1) printf()関数を用いて値を表示する  
printf("0番目の値: %d", person[0]);  
printf("1番目の値: %d", person[1]);  
printf("2番目の値: %d", person[2]);
- 例2) 計算する  
int result;  
result = (person[0]+person[1]+person[2])/3;

# プログラム例1(array\_sample.c)

要素数3の配列にキーボードから値を読み込み、配列の要素を表示

```
1 #include <stdio.h>
2
3 int main(){
4     int numbers[3];
5     // 配列に3つの値を読み込み
6     scanf("%d", &numbers[0]);
7     scanf("%d", &numbers[1]);
8     scanf("%d", &numbers[2]);
9     // 配列の要素を表示
10    printf("numbers[0]: %d\n", numbers[0]);
11    printf("numbers[1]: %d\n", numbers[1]);
12    printf("numbers[2]: %d\n", numbers[2]);
13    return 0;
14 }
```



# 本日の内容

- 講義(配列)
  - 配列の復習
  - 配列と繰り返し構文の組み合わせ
  - 定数の利用
- 演習
  - 配列とfor文を使ったプログラム

# 質問

先程のプログラム(array\_sample.c)を書き換え、要素数10の配列に  
キーボードから値を読み込み、配列の要素を表示する  
プログラムはどう作る？

# 再掲: プログラム例1(array\_sample.c)

```
1 #include <stdio.h>
2
3 int main(){
4     int numbers[3];
5     // 配列に3つの値を読み込み
6     scanf("%d", &numbers[0]);
7     scanf("%d", &numbers[1]);
8     scanf("%d", &numbers[2]);
9     // 配列の要素を表示
10    printf("numbers[0]: %d\n", numbers[0]);
11    printf("numbers[1]: %d\n", numbers[1]);
12    printf("numbers[2]: %d\n", numbers[2]);
13    return 0;
14 }
```

要素数を10に変更

scanf()を10回書く  
→大変!

printf()を10回書く  
→大変!

# プログラム例2(array10\_sample.c)

- for文を組み合わせることにより簡潔に記述可能

```
1 #include <stdio.h>
2
3 int main(){
4     int numbers[10];
5     int i; // カウンタ変数を宣言
6     // 配列に10個の値を読み込み
7     for(i=0; i<10; i++) {
8         scanf("%d", &numbers[i]);
9     }
10    // 配列の要素を表示
11    for(i=0; i<10; i++) {
12        printf("numbers[%d]: %d", i, numbers[i]);
13    }
14    return 0;
15 }
```

# 本日の内容

- 講義(配列)
  - 配列の復習
  - 配列と繰り返し構文の組み合わせ
  - 定数の利用
- 演習
  - 配列とfor文を使ったプログラム

# 再掲: プログラム例2(array10\_sample.c)

プログラム中で10という数値が何度も登場。要素数を11に変更しようとするプログラムを何箇所も書き換えなければならない

```
1 #include <stdio.h>
2
3 int main(){
4     int numbers[10];
5     int i; // カウンタ変数を宣言
6     // 配列に10個の値を読み込み
7     for(i=0; i<10; i++) {
8         scanf("%d", &numbers[i]);
9     }
10    // 配列の要素を表示
11    for(i=0; i<10; i++) {
12        printf("numbers[%d]: %d", i, numbers[i]);
13    }
14    return 0;
15 }
```

# プログラム例3(array10\_const.c)

C言語の定数という機能を使うと10などの値に名前をつけることができる(変数と似ている！)

```
1 #include <stdio.h>
2
3 #define N 10
4
5 int main(){
6     int numbers[N];
7     int i; // カウンタ変数を宣言
8     // 配列に10個の値を読み込み
9     for(i=0; i<N; i++) {
10         scanf("%d", &numbers[i]);
11     }
12     // 配列の要素を表示
13     for(i=0; i<N; i++) {
14         printf("numbers[%d]: %d", i, numbers[i]);
15     }
16     return 0;
17 }
```

# プログラム例3(array10\_const.c)

C言語の定数という機能を使うと10などの値に名前をつけることができる

```
1 #include <stdio.h>
2
3 #define N 10
4
5 int main(){
6     int numbers[N];
7     int i; // カウンタ変数を宣言
8     // 配列に10個の値を読み込み
9     for(i=0; i<N; i++) {
10         scanf("%d", &numbers[i]);
11     }
12     // 配列の要素を表示
13     for(i=0; i<N; i++) {
14         printf("numbers[%d]: %d", i, numbers[i]);
15     }
16     return 0;
17 }
```

#define 定数名 置き換わる値

実行時にNが10に置き換わる



# 変数と定数

- 変数
  - 変数には何度でも代入可能  
→ プログラム中で値が何度でも変わる場合に使用
  - 宣言方法:  
例) `int average = 10;`
- 定数
  - 定数には一度しか代入できない  
→ プログラム中で値が変わらない場合に使用
  - 宣言方法(main()関数の外で):  
`#define N 100`

# 演習0

- 講義資料中の以下のプログラムを入力し動作を確認しなさい
  - array\_sample.c
  - array10\_sample.c
  - array10\_const.c

# 演習1

1. array10\_sample.cを、要素数を11にしたプログラムに書き換えarray11\_sample.cというファイル名で保存しなさい
  2. array10\_const.cを、要素数11にしたプログラムに書き換えarray11\_const.cというファイル名で保存しなさい
- 定数を使ったプログラムにはどのような利点があるか考察しなさい

# 演習2

- 要素数10の配列にscanf()関数を用いてキーボードから値を読み込み、合計値を表示するプログラムをarray\_sum.cというファイル名で作成しなさい。

実行例)

\$ ./a.out

number[0]: 100

number[1]: 40

...

number[9]: 0

合計値は70です。

# 演習3

- 要素数5の配列にキーボードから整数を読み込み、最大値を表示するプログラムを作成し、array\_max.cというファイル名で保存しなさい

ヒント1: 最大値を格納する変数maxを宣言)

ヒント2: if文を使う

出力例)

\$ ./a.out

4

3

5

6

7

最大値: 7

## (余力がある人向け)演習4

- 要素数5の配列にキーボードから整数を読み込み、大きい順(降順)に並べ替えるプログラムを作成しarray\_sort.cというファイル名で保存しなさい

ヒント2: for文の入れ子を使うと良い

次週解説します。どのように並べ替えたら良いか考えてみてください。

# 次回

- 次回は演習4の解説+残りの時間は演習にします。

# 試験

- 2/2 11:10-12:40 (予定)
- マシン内の資料、プログラムのみ参照可