

プログラミング概論

<http://bit.ly/kosen01>

Week13@後期
(2016/12/21)

本日の内容

- 講義(配列)
 - 配列とは？
 - 配列の宣言、代入、呼び出し
 - 配列のプログラム例
- 演習
 - 配列を使ったプログラム
- (次回; 2017/1/12)
 - 配列と繰り返し構文を組み合わせる

本日の内容

- 講義(配列)
 - 配列とは？
 - 配列の宣言、代入、呼び出し
 - 配列のプログラム例
- 演習
 - 配列を使ったプログラム
- (次回; 2017/1/12)
 - 配列と繰り返し構文を組み合わせて使う

質問1

- 3人の学生の平均点を計算し、
結果を整数で表示する
プログラムはどのように作りますか？

3人の点数の平均

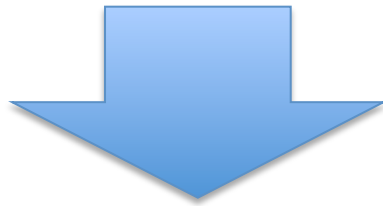
```
1 #include <stdio.h>
2
3 int main() {
4     // 1. int型の変数を3つ作る
5     int person1;
6     int person2;
7     int person3;
8     // 2. 値を代入
9     person1 = 90;
10    person2 = 100;
11    person3 = 80;
12    // 3. 平均を計算して表示
13    printf("平均値 : %d\n", (person1+person2+person3)/3);
14    return 0;
15 }
```

質問2

- 1学年(100人)の平均点はどのように計算しますか？

質問2

- 1学年(100人)の平均点はどのように計算しますか？



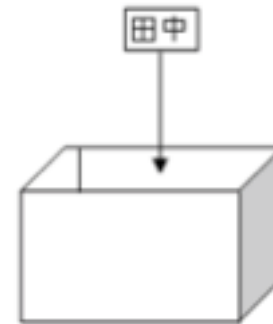
- int型の変数をperson1からperson100まで宣言しなければならない
→ こんなときは“配列”を使うと便利

配列とは？

- 同じ型の変数をまとめて作成する機能

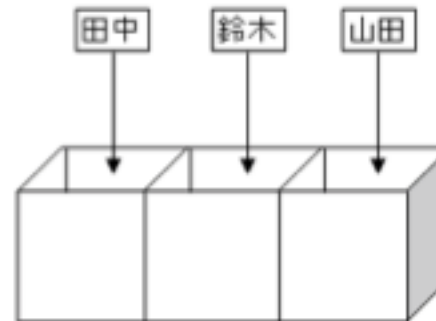
通常の変数の宣言

```
int person;
```



配列の宣言

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



本日の内容

- 講義(配列)
 - 配列とは？
 - 配列の宣言、代入、呼び出し
 - 配列のプログラム例
- 演習
 - 配列を使ったプログラム
- (次回; 2017/1/12)
 - 配列と繰り返し構文を組み合わせて使う

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

1. 配列の宣言

- 配列の宣言

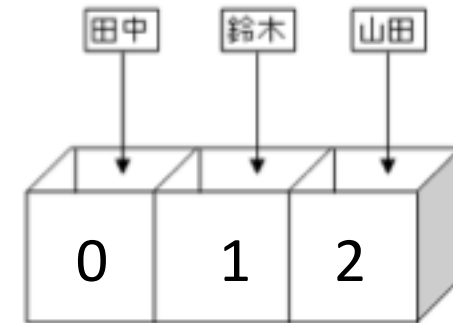
```
int person[3];
```

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！

2. 配列への値の代入(1/2)

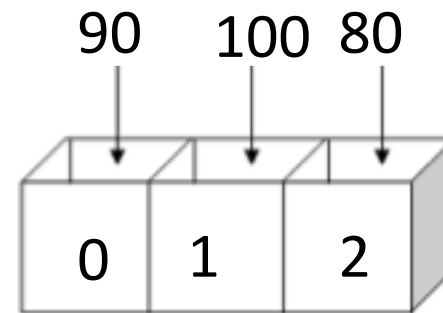
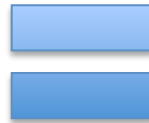
配列名[番号(=添字)] = 値;

例)

person[0] = 90;

person[1] = 100;

person[2] = 80;

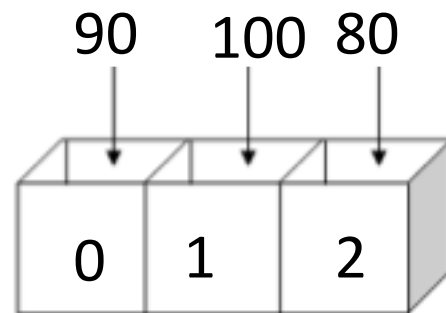


1.2. 宣言と初期化を同時に行う

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};
```



3. 配列の値を取り出す

- 例1) printf()関数内で呼び出す
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) 計算する際に呼び出す
int result;
result = (person[0]+person[1]+person[2])/3

配列はどのようなときに使う？

- 同じ種類のデータを複数列挙する場合には配列が使えないか検討する
 - クラス全員の成績一覧
 - フィボナッチ数を計算して配列に保存しておく
0 1 1 2 3 4 ...

同じ処理をするプログラムでも
配列を使うとより簡潔に読みやすいプログラムになることがある

本日の内容

- 講義(配列)
 - 配列とは？
 - 配列の宣言、代入、呼び出し
 - 配列のプログラム例
- 演習
 - 配列を使ったプログラム
- (次回; 2017/1/12)
 - 配列と繰り返し構文を組み合わせる

3人の点数の平均(配列なしの場合)

```
1 #include <stdio.h>
2
3 int main() {
4     // 1. int型の変数を3つ作る
5     int person1;
6     int person2;
7     int person3;
8     // 2. 値を代入
9     person1 = 90;
10    person2 = 100;
11    person3 = 80;
12    // 3. 平均を計算して表示
13    printf("平均値 : %d\n", (person1+person2+person3)/3);
14    return 0;
15 }
```

3人の点数の平均(配列を使う場合)

average3_array.c

```
1 #include <stdio.h>
2
3 int main() {
4     // 1. 配列を宣言
5     int person[3];
6     // 2. 配列への値の代入
7     person[0] = 90;
8     person[1] = 100;
9     person[2] = 80;
10    // 3. 平均を計算して表示
11    printf("平均値 : %d\n", (person[0]+person[1]+person[2])/3);
12    return 0;
13 }
```

3人の点数の平均(宣言と初期化を同時)

average3_array2.c

```
1 #include <stdio.h>
2
3 int main() {
4     // 1. 配列を宣言と初期化を同時に行う
5     int person[3] = {90, 100, 80};
6     // 3. 平均を計算して表示
7     printf("平均値 : %d\n", (person[0]+person[1]+person[2])/3);
8     return 0;
9 }
```

演習0

- 講義資料中のaverage3_array1.cを記述し動作を確認しなさい。
- 講義資料中のaverage3_array2.cを記述し動作を確認しなさい。

演習1

1. 5人の平均値を計算するように
average3_array1.cを書き換えなさい
2. 5人の平均値を計算するように
average3_array2.cを書き換えなさい

演習2

- 値を入れ替えるプログラム(`swap.c`)を作成しなさい
 1. 要素数2の配列`numbers`を宣言する
 2. `printf()`関数で「値:」と表示し、その後`scanf()`関数を用いて`numbers`の各要素に整数を代入する
 3. `printf()`関数を用いて`numbers[0]`, `numbers[1]`の値を表示する
 4. `numbers[0]`と`numbers[1]`の値を入れ替える
 5. `printf()`関数を用いて`numbers[0]`, `numbers[1]`の値を再度表示する

正しく動作した場合の例は以下のコマンドで確認できます。
\$ ~ishigaki/swap

演習3

- 配列の要素を小さい順に並べ替えるプログラム (sort.c)を作成しなさい
 1. 要素数3の配列numbersを宣言する
 2. scanf()関数を用いてnumbersの各要素に整数を代入する
 3. printf()関数を用いてnumbers[0], numbers[1], numbers[2]の値を表示する
 4. 配列numbersの要素を小さい順に並べる (ヒントは次のページ)
 5. printf()関数を用いてnumbers[0], numbers[1], numbers[2]の値を再度表示する

正しく動作した場合の例は以下のコマンドで確認できます。
\$ ~ishigaki/sort

ヒント: 3つの要素を小さい順に 並べ替える

- 並べ替え問題のことを“ソート”と呼びます。

Step1: numbers[0]とnumbers[1]を比較
numbers[1]の方が小さければ値を入れ替える

90	40	20
----	----	----

2つの値を比較し入れ替える

Step2: numbers[0]とnumbers[2]を比較
numbers[2]の方が小さければ値を入れ替える

40	90	20
----	----	----

2つの値を比較し入れ替える

Step3: numbers[1]とnumbers[2]を比較
numbers[2]の方が小さければ値を入れ替える

40	90	20
----	----	----

2つの値を比較し入れ替えるとソート完了

次回

- 「配列」と繰り返し文(for ,whileなど)を組み合わせ、値の代入や呼び出しをより短く簡潔に記述する方法を学習します。
- 次々回(1/19)は演習回、その次は試験になります。
- 放課も演習室を開放しているので活用してください。(開いていないときは近くの先生に開けてもらってください。)