





プログラミング概論

<http://bit.ly/kosen01>

Week4@後期
2016/10/19

全体像-プログラミング概論-

1.  C言語プログラムの実行方法
(コンパイルcc、実行a.out)
2.  変数
(int, float, char)
3.  条件分岐
(if文、switch文、条件式、論理演算)
4.  繰り返し
(for文、while文、do while文)
5. 関数
6. 配列

本日の内容

- 講義
 - 関数とは？
- 演習
 - 関数を使ったプログラム
 - 試験に向けた自習

関数とは？

- 関数(function) :
よく使う処理をまとめ、簡潔な記述で呼び出す機能
- 関数について学ぶべき内容
 1. 関数の宣言と呼び出し(今回)
 2. 関数の戻り値と引数 (試験明け)
 3. 関数の再帰呼び出し(試験明け)

関数の記述方法

- 以下の順で記述する
 1. 形式(シグネチャ)宣言
 2. 処理記述
 3. 呼び出し

1. 2. 3. の記述場所

```
1 #include <stdio.h>
2
3 // 1. 形式宣言
4
5
6 // 2. 処理記述
7
8
9 int main() {
10     // 3. main()関数からの呼び出し
11     return 0;
12 }
```

1. 形式宣言

```
1 #include <stdio.h>
2
3 // 1. 形式宣言
4 void average();
5
6 // 2. 処理記述
7
8
9 int main() {
10     // 3. main()関数からの呼び出し
11     return 0;
12 }
```

2. 処理記述

```
1 #include <stdio.h>
2
3 // 1. 形式宣言
4 void average();
5
6 // 2. 処理記述
7 void average() {
8     // 1と2と3の平均値を表示
9     printf("%d", (1+2+3) / 3);
10 }
11
12 int main() {
13     // 3. main()関数からの呼び出し
14
15     return 0;
16 }
```


3. main()関数からの呼び出し

```
1 #include <stdio.h>
2
3 // 1. 形式宣言
4 void average();
5
6 // 2. 処理記述
7 void average() {
8     // 1と2と3の平均値を表示
9     printf("", (1+2+3) / 3);
10 }
11
12 int main() {
13     // 3. main()関数からの呼び出し
14     average();
15     return 0;
16 }
```

\$./a.out後の処理順

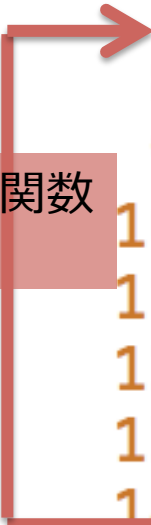
```
1 #include <stdio.h>
2
3 // 1. 形式宣言
4 void average();
5
6 // 2. 処理記述
7 void average() {
8     // 1と2と3の平均値を表示
9     printf("%d", (1+2+3) / 3);
10 }
11
12 int main() {
13     // 3. main()関数からの呼び出し
14     average();
15     return 0;
16 }
```

1. まずはmain()から実行開始

\$./a.out後の処理順

```
1 #include <stdio.h>
2
3 // 1. 形式宣言
4 void average();
5
6 // 2. 処理記述
7 void average() {
8     // 1と2と3の平均値を表示
9     printf("%d", (1+2+3) / 3);
10 }
11
12 int main() {
13     // 3. main()関数からの呼び出し
14     average();
15     return 0;
16 }
```

average()関数
を実行



1. まずはmain()から実行開始

\$./a.out後の処理順

```
1 #include <stdio.h>
2
3 // 1. 形式宣言
4 void average();
5
6 // 2. 処理記述
7 void average() {
8     // 1と2と3の平均値を表示
9     printf("%d", (1+2+3) / 3);
10 }
11
12 int main() {
13     // 3. main()関数からの呼び出し
14     average();
15     return 0;
16 }
```

2. average()関数
を実行

1. まずはmain()から実行開始

3. main()に戻る

補足

- 関数は何個でも宣言可能
- 関数は何回でも呼び出し可能

演習1

- 「こんにちは。」と表示する関数、「おやすみ」と表示する関数をそれぞれ記述し、main()関数から呼び出しなさい。

```
void greeting_daytime(); // こんにちは  
void greeting_night();  // おやすみ
```

出力例)

```
$ ./a.out
```

```
こんにちは。
```

```
おやすみ
```

演習2

- 1から1000までの奇数を表示する関数を記述し、main()関数から呼び出しなさい。

関数の形式

```
void odd_to_1000();
```

出力例)

```
$ ./a.out
```

```
1
```

```
2
```

```
...
```

```
..
```

```
1000
```

演習3

- 1から1000までの素数をすべて表示する関数を記述しmain()関数から呼び出さない。

関数の形式

```
void prime_to_1000();
```

実行例)

```
$ ./a.out
```

```
1
```

```
3
```

```
5
```

```
7
```

```
...
```

```
..
```


演習4

- 試験に向けた練習も兼ねて、過去に作ったプログラムを関数を使った処理に書き換えなさい

次回

- 試験は
- 次回は関数の引数と戻り値という概念を学習します
- 引数や戻り値を使うとmain()関数と独自に宣言した関数の間でデータをやりとりすることが出来ます。