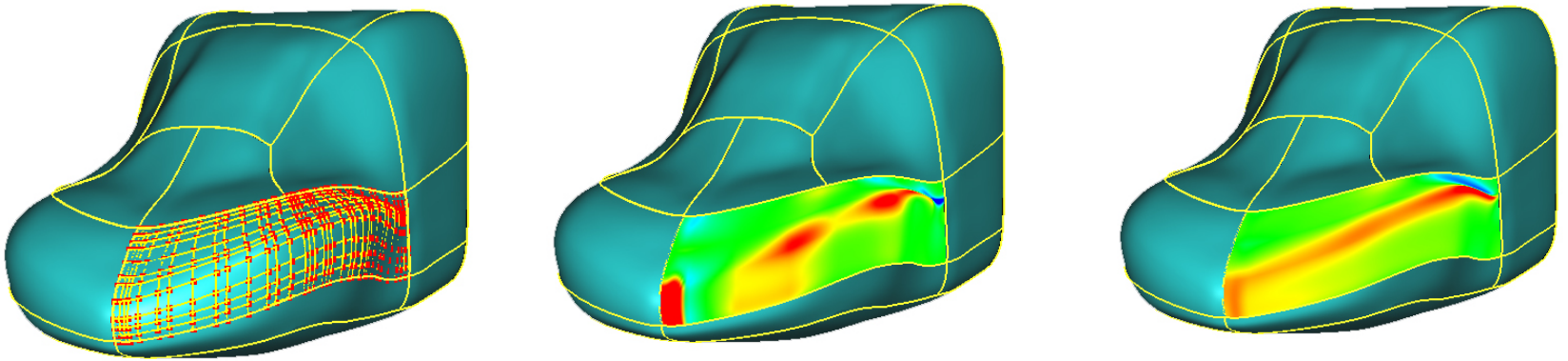
数値解析

2021年度前期　第2週 [4月15日]



静岡大学工学部機械工学科
ロボット・計測情報講座　三浦 憲二郎

講義アウトライン [4月15日]

- C言語の基礎
  - 関数
    - 引数
    - 基本構成
    - データ渡し
    - return文

- C言語の復習・確認テスト

関数　p.124

◇ 関数

【説明】

Cプログラムは関数の寄せ集め。

プログラムはmainだけでも十分に機能する。

プログラムが大きくなると、巨大な一つのmain関数だけでは、プログラムの見通しが悪くなる。そこでプログラムを手頃な大きさの関数に分割する。

```
void function_name (void)
{
}
```

関数には自由に名前をつけてかまわないが、“main”だけは特別な名前でプログラムの中には必ず“main”という名前の関数が必要。プログラムはこのmain関数から実行開始される。

関数

```
#include <stdio.h>
void line(void) {
    printf("━");
}
int main(void) {
    printf("静大¥n");
    line(); /* 関数line()の呼び出し  */
    return 0;
}
```

は以下のプログラムと同じ働き

```
#include <stdio.h>
int main(void) {
    printf("静大¥n");
    printf("━"); /* line() */;
    return 0;
}
```

関数の引数 p.126

◇ 関数の引数

【説明】

関数の呼び出し側と関数本体側でデータのやりとりをするには引数を用いる。引数は関数名に接続するかっこの中に、データ型と変数名を書く。たとえば、

```
void add(int a, int b)
{
  …
}
```

と書く。このときの変数 a や b を「仮引数」と呼ぶ。それはどんな値が渡されるか分からない仮の引数だから。この関数を呼び出す側では、

```
add(100, 200);  /* 値を指定。      */
add(y, z);      /* 変数名を指定   */
```

のように実際のデータを記述する。これを「実引数」と呼ぶ。

関数の引数:用例

【用例】

```
void print(int n)
{ /* 引数を持つ関数を定義する。*/
    int i;
    for(i=0; i<n; i++) { /* 引数 n の値の回数だけ */
                         /* printf() を実行      */
       printf("hello¥n");
    }
}
int main(void) {
    int i;
    i = 5;
    print(3); /* 定数を引数として関数を呼び出す。*/
    print(i); /* 変数を引数として関数を呼び出す。*/
    return 0;
}
```

関数の引数:用例 p.124

リスト　2数の平均値を求めるプログラム

```
#include <stdio.h>
double ave(double x, double y); /*プロトタイプ宣言*/
int main(void) {
     double a,b,avdt;
     a=11.11; b=33.33;
     avdt=ave(a,b);
     printf("a=%f b=%f 平均=%f¥n",a,b,avdt);
     return 0;
}
double ave(double x, double y){
    double wk;
    wk=(x+y)/2.0;
    return wk;
}
```

関数の基本構成 p.124

関数の基本形

戻り値の型　関数名(引数の宣言)
{
　　変数の宣言と文の記述
}

値を返す方法
return 式;

処理を打ち切る**return**

```
void myfunc(int a)
{
     if(a<=0)return;
     . . .
}
```

void型の関数 p.126

```
#include <stdio.h>
void putd(int a); /* プロトタイプ宣言*/

int main(void)
{
    putd(123);
    return 0;
}

void putd(int a)
{
    printf("%d¥n",a);
}
```

実行結果

```
123
```

値渡しの方法 p.132

数値を渡す

値による呼び出し Call by value

(名前による呼び出し Call by name ではない)

呼び出し側

```
a=11.11;
b=33.33;
avdt=ave(a,b)
```

関数定義側

```
double ave(double x, double y)
{
    double wk;
    z=(x+y)/2.0;
    return z;
}
```

課題1-3

課題1－3

```
#include <stdio.h>
double product(double x, double y);
int main(void)
{
    double a, b, mul;
    a = 11.11; b = 22.22; mul = product(a,b);
    printf("a=%f b=%f mul=%f￥n",a,b,mul);
}

double product(double x, double y)
{
    double z;
    z=x*y;
    return z;
}
```

課題1-3 別解

課題1－3 別解

```
#include <stdio.h>
double product(double x, double y);
int main(void)
{
    double a, b, mul;
    a = 11.11; b = 22.22; mul = product(a,b);
    printf("a=%f b=%f mul=%f￥n",a,b,mul);
}

double product(double x, double y)
{
     return x*y;
}
```

課題1-5

課題1-5

```
#include <stdio.h>
double minValue(double x, double y);
int main(void)
{
    double a, b, mul;
    a = 11.11; b = 22.22; mul = minValue(a,b);
    printf("a=%f b=%f mul=%f¥n",a,b,mul);
}

double minValue(double x, double y)
{
     if(x<=y)
        return x;
    else
        return y;
}
```

課題1-5 別解

課題1-5 別解

```
#include <stdio.h>
double minValue(double u, double v);
int main(void)
{
    double a, b, min0;
    a = 11.11; b = 22.22; min0 = minValue(a,b);
    printf("a=%f b=%f min0=%f¥n",a,b,min0);
}

double minValue(double u, double v)
{
     return u<=v ? u : v;    /* ? : は演算子 */
}
```

課題1-6 double型マシンイプシロン

課題1-6

```
#include <stdio.h>
int main(void)
{
    double deps=1.0;
    double dtmp;
    for(dtmp=deps+1.;dtmp>1;){
        deps/=2.0;
        dtmp=deps+1.0;
    }
    printf("double型のMachine epsilonは%-16e¥n",2.0*deps);
}
```

まとめ

- C言語の基礎
  - 関数
    - 引数
    - 基本構成
    - データ渡し
    - return文
    - マシンイプシロンの計算