

# 関数のまとめ

山本昌志\*

2006年11月24日

## 概要

本日で関数の学習を終了する。そこで、これまで学習した内容をまとめ、最後の練習を尾子なう。

## 1 関数のまとめ

### 1.1 関数

関数とは、処理の手続きをまとめたものである。関数を使うことにより、C言語のプログラムは

- 何度も同じことを書くのは面倒なので、処理を一つにまとめる。
- プログラムの内容を分かりやすくするために、処理を機能単位に分割する。

とすることができる。特に、2番目が重要で「プログラムのソースは分かりやすくなくてはならない」ということが実現できる。

関数をつくり、使うためには、ソースプログラムを図1のように記述する。必要な記述は、以下の3つである。

- プロトタイプ宣言。関数の入出力(引数や戻り値)の仕様を示す。
- 関数の定義。関数での処理の内容を示す。
- 関数のコール(Call:呼び出し)。関数を動作させる。

関数は特定の処理をする専用のプログラムで、必要な引数を伴って関数名を書けばその関数のプログラムを実行することができる。処理に必要なデータは関数を呼び出すときに引数として渡す。呼出元の引数を実引数、呼び出された関数側の引数を仮引数と言う。関数を呼び出したとき、実引数の値が仮引数の変数にコピーされる。それを使って、関数は処理を行い。呼出元へ値を返すときには、return文を使う。return文により返す値を戻り値と言う。

---

\*独立行政法人 秋田工業高等専門学校 電気情報工学科

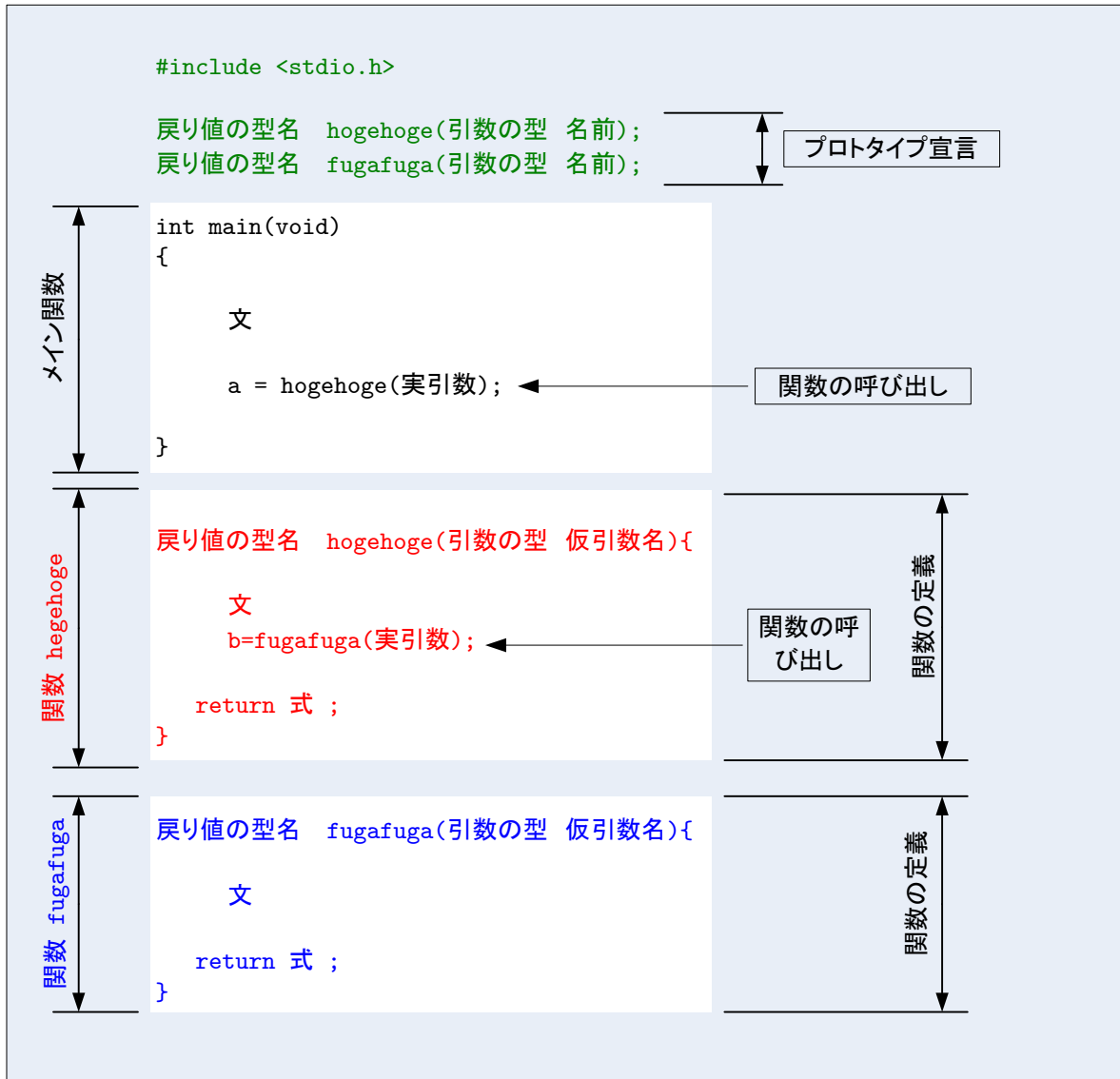


図 1: ユーザー定義関数を含んだソースプログラムの書き方

## 1.2 変数のスコープ

先週は、C言語の変数のスコープについて学習した。変数のスコープとは、その変数の有効範囲のこと  
で、それは宣言する場所で決まる。以下の箇条書きと図2に示すように3種類のスコープがある。

- 全ての関数の外側、プログラムの先頭付近で宣言した変数は全ての関数で有効である。これをグローバル変数と呼ぶ。
- 関数の先頭で宣言した変数は、その関数内のみで有効である。これを、ローカル変数と呼ぶ。また、関数の仮引数もローカル変数である。
- コードブロック—{ } で囲まれた部分<sup>1</sup>—の先頭で宣言した変数は、そのブロック内のみで有効になる。これもローカル変数のひとつであるが、ここではブロック内宣言の変数と呼ぶことにする。

スコープが異なれば、同じ名前の変数名を使うことができる。その場合、優先度の高い順に並べると、ブロック内宣言の変数 → ローカル変数 → グローバル変数となる。

## 1.3 記憶クラス

C記憶クラスは、メモリーにデータを格納する方法のことである。C言語には、4個の記憶クラス指定子(1)auto, (2)extern, (3)register, (4)staticがある。これらは、変数宣言の先頭に

```
auto int hogehoge;  
extern int fugafuga;  
register int foo;  
static int bar;
```

と書く。

記憶クラスは4種類あるが、この中でstaticはローカル変数<sup>2</sup>とグローバル変数では、動作が異なる。したがって、5種類の動作があると考えられる。

記憶クラスを指定しない場合、自動変数(auto)となる。デフォルトは自動変数。そのため、通常、自動変数を使う場合、記憶クラス指定子autoは書かない。

---

<sup>1</sup>if文やループの時使った。

<sup>2</sup>ブロック内宣言の変数もローカル変数とする。

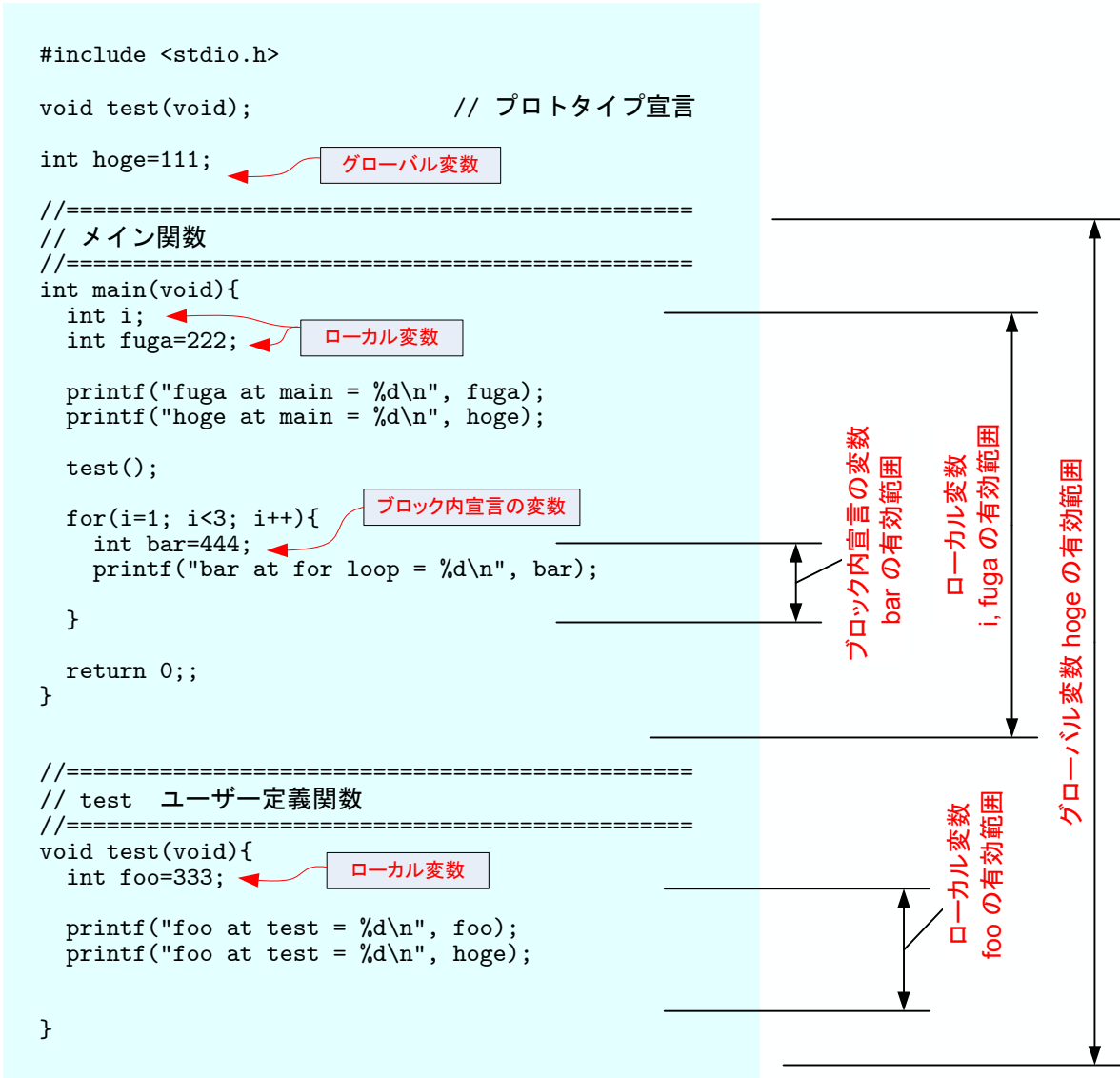


図 2: 変数のスコープ (有効範囲)

表 1: 記憶クラスの特徴

記憶クラス	指定子	特徴
自動変数	auto	ローカル変数の場合は、関数が呼び出される毎に記憶領域の確保と初期化 (初期値の指定がある場合) が行われ、関数の処理が終わったときにその変数は消滅する。グローバル変数の場合は、プログラム実行中、変数の記憶領域は保持される。
ローカル変数の静的変数	static	プログラムの実行に先立って、記憶領域の確保と初期が行われる。プログラムの実行中、記憶領域は保持される。
グローバル変数の静的変数	static	分割コンパイルの場合、そのファイルのみで使えるグローバル変数ということを示している。
外部変数	extern	この記憶クラスはグローバル変数のみに使う。これが指定されると、記憶領域の確保は行われず、どこか他のファイルにこの変数があることを示す。記憶領域は、どこか他のファイルにある変数を使う。
レジスター変数	register	レジスターを記憶領域として使う。レジスターは CPU の記憶領域で、メインメモリーよりも高速のアクセスが可能である。したがって、処理の高速になる。

## 2 プログラム作成の練習

[練習 1] 三角形の面積  $S$  をヘロンの公式

$$s = \frac{a + b + c}{2} \quad (1)$$

$$S = \sqrt{s(s-a)(s-b)(s-c)} \quad (2)$$

を使って計算するプログラムを作成せよ。ここで、 $a, b, c$  は辺の長さである。プログラムの条件は以下の通りとする。

- キーボードから、辺の長さを読み込む。
- 面積の計算は、関数を使うこと。ただし、 $(s-a)(s-b)(s-c) \leq 0$  の場合は、“三角形になりません”と表示して、プログラムを止めること。

[練習 2] 数学関数

$$f(x) = \frac{x+5}{x^2+5} - x^2 - 10x + 15 - 20 \sin(x) \quad -100 \leq x \leq 100 \quad (3)$$

の最大値を求めるプログラムを作成せよ。計算の精度は、 $10^{-4}$  とする。ただし、この数学関数の計算には C 言語のユーザー定義関数を使うこと。

[練習 3] リスト 1 と 2 の 2 つのソースファイルから実行ファイルを作成する。以下の問に答えよ。

- プログラムに使われている変数を (1) グローバル変数、(2) ローカル変数、(3) ブロック内宣言の変数に分けよ。

- プログラムに使われている変数を記憶クラス毎に分けよ .
- プログラムの実行結果を示せ .

リスト 1: 様々な変数に関する問いのプログラム (prog1.c) .

```

1 #include <stdio.h>
2
3 int hoge(int a, int b);
4 void fuga(int b);
5
6 int a=2, i, j=3;
7 extern int k;
8
9 //===== main 関数 =====
10 int main(void)
11 {
12
13     int a=6, m;
14
15     for (m=1; m<=3; m++){
16         int j=4;
17         a++;
18         printf("\n ----- m=%d -----\n", m);
19         i=hoge(a, j);
20         fuga(i);
21         printf("at main a=%d\ti=%d\n", a, i);
22     }
23
24     return 0;
25 }
26
27 //===== ユーザー定義関数 =====
28 int hoge(int a, int b)
29 {
30     static int k=10;
31     int l=20;
32
33     printf("at hoge a=%d\tb=%d\n", a, b);
34     printf("at hoge k=%d\tl=%d\n", k, l);
35
36     k++;
37     l++;
38
39     return a+b;
40 }

```

リスト 2: 様々な変数に関する問いのプログラム (prog2.c) .

```

1 #include <stdio.h>
2
3 void fuga(int b);
4
5 //===== ユーザー定義関数 =====
6 static int i=30;
7 int d=1;
8
9 void fuga(int b)
10 {
11     i++;

```

```
12     d+=i;
13     printf(" at fuga i=%d\td=%d\n", i,b);
14
15 }
```

### 3 課題

試験前なので，課題は無し．ただし，試験範囲，

- 教科書の p.153-216
- 第 17 回～第 24 回の講義

を十分勉強すること．