

# 1 局所的変数 (ローカル変数)

## 1.1 学習のポイント

関数の中で宣言されている変数の特徴についてまなびます。

## 1.2 局所的変数 (ローカル変数)

C の変数は、それが宣言されている関数の中だけ有効です。これを局所的変数 (ローカル変数) と呼びます。局所的変数は、関数をモジュール化し独立性を高める上で重要なものです。

「C 言語」(河西朝雄著 ナツメ社)104 頁

## 1.3 例題 22

```
/*
  例題 22 C 言語 105 頁
  main の変数 i と関数 func の変数 i の値の変化を見る。
  reidai22.c
*/

#include <stdio.h>

void func(void)
{
    int i;

    for(i = 0; i < 3; i++){
        printf("    func...i = %d\n", i);
    }
}

int main()
{
    int i;
```

```
    for(i = 0; i < 3; i++){
        printf("main...i = %d\n", i);
        func();
    }

    return 0;
}
```

「C 言語」(河西朝雄著 ナツメ社)105 頁

## 2 AI による大規模データ処理入門

### 2.1 前処理プログラム

時間とともに変化する時系列データの変動を、上昇あるいは変化なし (1)、および、下降 (0) で判定するためのプログラムです。時系列データとして、たとえば株価データを用いれば、変動の傾向を記号的に把握することが可能です。

AI による大規模データ処理入門 17 頁 小高知宏著 オーム社

### 2.2 データ前処理プログラム

```
/*
   データ前処理プログラム
   prep.c
   入力された数値の増減を調べる
*/

#include <stdio.h>

#define LIMITL 256
#define TRUE 1
#define FALSE 0

void eval(double data, double olddata);
```

```

int main()
{
    char    linebuf[LIMITL];
    double data;
    double olddata;
    int     firstdata = TRUE;

    while(fgets(linebuf, LIMITL, stdin) != NULL){

        if(sscanf(linebuf, "%lf", &data) <= 0){
            break;
        }

        if(firstdata == TRUE){
            firstdata = FALSE;
        }else{
            eval(data, olddata);
        }

        olddata = data;
    }

    return 0;
}

void eval(double data, double olddata)
{
    if(data >= olddata){
        printf("1\n");
    }else{
        printf("0\n");
    }
}

```

### 2.3 日経平均株価 nikkei.txt

```

8802.51
9723.24
10083.56

```

9520.89  
8542.73  
9006.78  
8695.06  
8839.91  
8870.16  
8928.29  
9446.01  
10395.18  
11138.66  
11559.36  
12397.91  
13860.86  
13774.54  
13677.32  
13668.32  
13388.86  
14455.80  
14327.94  
15661.87  
16291.31  
14914.53  
14841.07  
14827.83  
14304.11  
14632.38  
15162.10  
15620.77  
15424.59  
16173.52  
16413.76  
17459.85  
17635.14

## 2.4 使い方

```
./prep < nillei.txt
```