

プログラミング基礎演習

第6回

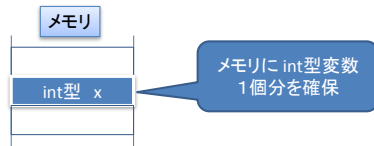
配列変数

- 変数
 - 値を格納しておくもの
- 配列変数
 - 変数の集合
 - 複数の値を効率よく扱うための変数
 - 複数の値を連続したメモリ空間に記憶

配列変数と宣言

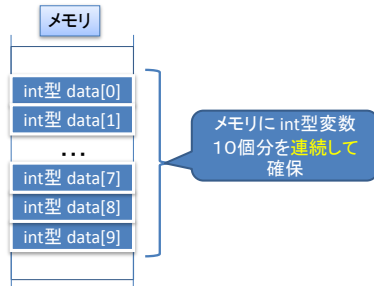
変数の宣言例

```
int x;
```



配列変数の宣言例

```
int data[10];
```



配列変数と宣言

配列変数の宣言

型 変数名 [確保する数];

(例)

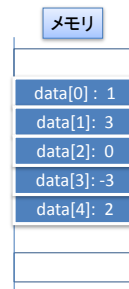
```
int data[5];
float x[1000];
```

注意:

int data[5]; と宣言した場合
data[0] ~ data[4] まで利用可

```
data[0] = 1;
data[1] = 3;
data[2] = 0;
data[3] = -3;
data[4] = 2;
```

5個



配列変数の初期化

変数の場合

```
int x=0;
```

配列変数の場合

```
int a[5] = { 1, 3, 2, 0, -1};
```



```
a[0] = 1;
a[1] = 3;
a[2] = 2;
a[3] = 0;
a[4] = -1
```

ただし、宣言部以外で

```
a = { 1, 3, 2, 0, -1};
```

のようには**できない**

配列変数 利用例

```
int a, sum=0, data[10];
double average;

for (i=0; i<10; i++)
{
    printf("%d 人目の点数:", i+1);
    scanf("%d", &a);
    data[i] = a;
}

for (i=0; i<10; i++)
    sum = sum + data[i];

average = (double)sum / 10.0;

printf("平均点は %f 点です。", average);
```

配列変数 利用例2

$$a_n = \frac{1}{3} a_{n-1} + 3, \quad a_0 = 2$$

```
#include <stdio.h>
#define MAX 1000

void main()
{
    int i;
    double a[MAX];

    a[0]=2;

    for ( i=1 ; i<MAX; i++)
        a[i] = a[i-1] / 3.0 + 3;
}
```

どうなる？

main()

```
{
    int x[5];
    int i;

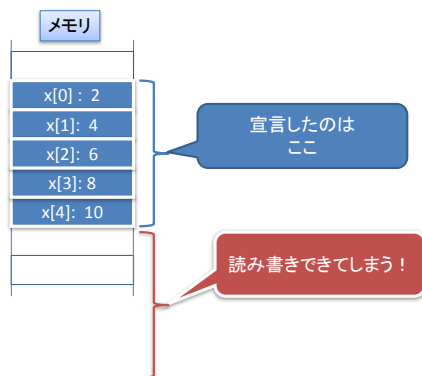
    for( i=0 ; i<10 ; i++)
        x[i] = i;
}
```

main()

```
{
    int x[5]={2,4,6,8,10};
    int i;

    for( i=0 ; i<10 ; i++)
        printf("%d ¥n", x[i]);
}
```

C言語の欠点



BOF(Buffer overflow)

