

アルゴリズム  
第7回授業  
“2次元配列と計算量”  
(教科書 Page 43-48)

山口雅樹 (CISSP)

<https://github.com/masakage/algorithm>

# 本日の進め方

- ・ 前回の復習（引数と返却値）
- ・ Training 1-4と1-5
- ・ 2次元配列
- ・ Training 1-6
- ・ ドリルの解答
- ・ 計算量
- ・ まとめ

# 1-9 2次元配列 (WIKIより)

添字が一つの配列を“1次元配列”と呼ぶ。  
2次元配列は、添字二つで要素を特定する配列となる。  
エクセルと考え方は同じ

例：配列名[行要素数][列要素数]

列

行

整数型配列 T[][]	[0]	[1]	[2]	[3]	[4]
[0]	T[0][0]	T[0][1]	T[0][2]	T[0][3]	T[0][4]
[1]	T[1][0]	T[1][1]	T[1][2]	T[1][3]	T[1][4]
[2]	T[2][0]	T[2][1]	T[2][2]	T[2][3]	T[2][4]

# 配列の初期化について (44page)

- プログラム名：2次元配列初期化 /\* 教科書 44ページサンプル \*/
- 整数型：T[3][5]
- 整数型：j

■ j < 5  
| ● T[0][j] ← 0  
| ● j ← j + 1  
□

0行目の要素を0で初期化していく。

列

行

整数型配列 T[][]	[0]	[1]	[2]	[3]	[4]
[0]	T[0][0]←0	T[0][1] ←0	T[0][2] ←0	T[0][3] ←0	T[0][4] ←0
[1]	T[1][0]	T[1][1]	T[1][2]	T[1][3]	T[1][4]
[2]	T[2][0]	T[2][1]	T[2][2]	T[2][3]	T[2][4]

# 配列の初期化について (45page)

○プログラム名：2次元配列初期化 /\* 教科書 45ページサンプル \*/

○整数型：T[3,5]

○整数型：i

○整数型：j

```
■ i < 3
| ● j ← 0
| ■ j < 5
| | ● T[i,j] ← 0
| | ● j ← j + 1
| □
| ● i ← i + 1
□
```

まず 0行目の要素を0で初期化していく。  
次に 1行目の要素を0で初期化していく。  
最後に 2行目の要素を0で初期化していく。

列

行

整数型配列 T[][]	[0]	[1]	[2]	[3]	[4]
[0]	T[0][0]←0	T[0][1] ←0	T[0][2] ←0	T[0][3] ←0	T[0][4] ←0
[1]	T[1][0] ←0	T[1][1] ←0	T[1][2] ←0	T[1][3] ←0	T[1][4] ←0
[2]	T[2][0] ←0	T[2][1] ←0	T[2][2] ←0	T[2][3] ←0	T[2][4] ←0

# 2次元配列行毎の合計について (46page)

○プログラム名：2次元配列行合計算出 /\* 教科書 46ページサンプル \*/

○整数型：T[3,5]

○整数型：Line[3] /\* 行の合計を代入する \*/

○整数型：Gtotal

○整数型：Lidx

○整数型：Ridx

■Lidx : 0, Lidx < 3, 1

| ●Line[Lidx] ← 0

| ■Ridx : 0, Ridx < 5, 1

| | ●Line[Lidx] ← Line[Lidx] + T[Lidx, Ridx]

| □

□

●Gtotal ← 0

■Lidx : 0, Lidx < 3, 1

| ●Gtotal ← Gtotal + Line[Lidx]

□

まず 0行目の要素の全てをLine[0]に足していく  
次に 1行目の要素の全てをLine[1]に足していく  
最後に 2行目の要素の全てをLine[2]に足していく

Gtotal に Line[0]と、Line[1]と、Line[2]の合計を足す。

# 2次元配列行毎の合計について（46page）

整数型配列 T[][]	[0]	[1]	[2]	[3]	[4]	
[0]	T[0][0]	T[0][1]	T[0][2]	T[0][3]	T[0][4]	Line[0]=0行目要素の合計
[1]	T[1][0]	T[1][1]	T[1][2]	T[1][3]	T[1][4]	Line[1]=1行目要素の合計
[2]	T[2][0]	T[2][1]	T[2][2]	T[2][3]	T[2][4]	Line[2]=2行目要素の合計

最後にGtotal に Line[0]と、Line[1]と、Line[2]の合計を足している。

# Training 1-6 解答例 (教科書 46Page)

○プログラム名：9列9行の九九の表 /\* 教科書 46ページサンプル \*/

○整数型：Kuku[9,9]

○整数型：Lidx

○整数型：Cidx

■ Lidx : 0, Lidx < 9, 1

| ■ Cidx : 0, Cidx < 9, 1

| | ● Kuku[Lidx][Cidx] ← (Lidx + 1) × (Cidx + 1)

| □

□

	0	1	2	3	4	5	6	7	8
0	1	2	3	4	5	6	7	8	9
1	2	4	6	8	10	12	14	16	18
2	3	6	9	12	15	18	21	24	27
3	4	8	12	16	20	24	28	32	36
4	5	10	15	20	25	30	35	40	45
5	6	12	18	24	30	36	42	48	54
6	7	14	21	28	35	42	49	56	63
7	8	16	24	32	40	48	56	64	72
8	9	18	27	36	45	54	63	72	81



# 1-10 計算量

アルゴリズムの性能を“実行にどれだけの時間がかかるか”で考える。  
これを計算量といい、同じ結果が得られるなら計算量の小さいアルゴリズムがいい。

$T[0]$ から、 $T[N-1]$ までの合計は、 $O(\text{Order})$ 記法では、 $O(N)$ と表記する。

2重ループの場合は、 $O(MN)$ となり、ループが2つある場合は  $O(M+N)$  となる。  
この場合、 $O(M+N)$ のほうが、時間計算量が小さいと評価できる。