

# プログラミング基礎 2次元配列の課題

## 行列のスキャンとその確認

```
#include <stdio.h>
#include <math.h> //様々な計算が出来るようになる
```

絶対値; abs(x)  
平方根; sqrt(x)  
三角関数; sin(x)  
累乗; pow(x,y)

```
int main(){
int i,j;
double A[100][100];
int n;//行列の長さ
```

```
int n;
scanf("%d", &n);
double A[n][n];
```

これでも出来るようだが  
止めたほうが良い

```
scanf("%d", &n);
for(i=0; i<n; i++)
    for(j=0; j<n; j++)
        scanf("%lf", &(A[i][j]));
```

行列のスキャン  
-行列の長さをスキャン  
-行列の要素をスキャン

```
for(i=0; i<n; i++){
    for(j=0; j<n; j++){
        printf("%lf ", A[i][j]);
    }
    printf("¥n");
}
return 0;
}
```

行列のプリント(確認)  
-行列の要素をプリント  
-1行終わったら改行

"%lf "  
のように空白を入れると  
見栄えが良い

# サブ関数の利用

行列の長さn  
を書き換える必要があるので  
アドレスを渡す

```
#include <stdio.h>
#include <math.h>
```

`scanMatrix(A, &n);`//サブ関数。メイン関数内に書く

```
int main(){
int i,j;
double A[100][100];
int n;//行列の長さ
```

```
scanf("%d", &n);
for(i=0; i<n; i++)
    for(j=0; j<n; j++)
        scanf("%lf", &(A[i][j]));
```

```
void scanMatrix(int mat[100][100], int *sizeP){
int i, j;

/*sizePはポインタなので & は不要。&(*sizeP) でも可*/
scanf("%d", sizeP);
for(j=0; j<*sizeP; j++)
    for(i=0; i<*sizeP; i++)
        scanf("%d", &(mat[j][i]));
}
```

```
for(i=0; i<n; i++){
    for(j=0; j<n; j++){
        printf("%lf ", A[i][j]);
    }
    printf("\n");
}
return 0;
}
```

`printMatrix(A, n);`

```
void printMatrix(int mat[100][100], int size){
int i, j;
for(j=0; j<size; j++){
    for(i=0; i<size; i++){
        printf("%d ", mat[j][i]);
    }
    printf("\n");
}
}
```

## 指定した二つの行(列)を交換する

```
int main(){
int r1, r2; /* 交換する行の番号 */
int n;
double A[100][100];
scanMatrix(A, &n);
printMatrix(A, n);
```

```
/* 交換される番号をスキャン */
printf("\nTwo rows?");
scanf("%d %d", &r1, &r2);
swapRow(A, n, r1, r2); //交換する
```

```
void swapRow(int mat[100][100], int size, int r1, int r2){
int i;
int temp; //入れ替えの為の変数
for(i=0; i<size; i++){
    temp = mat[r1][i];
    mat[r1][i] = mat[r2][i];
    mat[r2][i] = temp;
}
}
```

```
printMatrix(A, n); //確認
}
```

```
A=3;
B=2;

A=B;(A=2になる)
B=A;(Aは2なので、B=2)
```

```
A=3;
B=2;
C; //入れ替えの為の変数

C=A;(Aを保存 ;C=3)
A=B;(A=2)
B=C;(B=3)
```

引く必要はないが  
計算が多少速くなる

```
for(i=0; i<100; i++)
    より
for(i=0; i<size; i++)
    のほうが早い
```

# 行と列を入れ替える; 転置

```
int main(){
  int n; /* 行列のサイズ */
  int A[100][100]; /* 行列 */

  scanMatrix(A, &n); // スキャン
  printMatrix(A, n); // 確認

  transpose(A, n); // 行列の転置

  printMatrix(A, n); // 確認
  return 0;
}
```

```
void transpose(int mat[100][100], int size){
  int i, j;
  int temp;
  for(j=0; j<size; j++){
    for(i=j; i<size; i++){
      でも可
    }
  }
}
```

