How to Run

$ bash hw6.sh
$ ./hw6

<실행결과>

```
(mteb) qmin@rose:~/na/Homework #6/Code$ ./hw6
============fitdata1=============
0.981888
0.002540
-0.375178
0.001250
0.982163
1.157731
============fitdata2=============
0.979907
0.000452
-1.192226
-0.001069
0.980346
0.491567
============fitdata3=============
0.980806
0.000545
-0.944462
-0.000717
0.979108
0.428951
```

Logics

```
void read_data(int num, float (*input1)[3], float (*output1)[1], float (*output2)[1], char* name){
    FILE *fp;
    fp = fopen(name, "r");
    for (int i = 0; i < 77; i++)
            fscanf(fp, "%f %f %f %f ", &input1[i][0], &input1[i][1], &output1[i][0], &output2[i][0]);
    fclose(fp);
}
```

* read_data함수에서는 fitdata#을 읽고 x,y는 input에 담아 저장하고, output1,2에 x', y'를 나눠담는다.

```
void fitting(float (*X)[3], float (*y)[1])
{

    float **X_t_mul_X = calloc(3 + 1, sizeof(float *));
    for (int i = 0; i <= 3; i++) {
        X_t_mul_X[i] = calloc(3 + 1, sizeof(float));
    }
    float **X_t_mul_y = calloc(3 + 1, sizeof(float *));
    for (int i = 0; i <= 3; i++) {
        X_t_mul_y[i] = calloc(3 + 1, sizeof(float));
    }

    for (int i = 1; i <= 3; i++)
        for (int j = 1; j <= 3; j++){
            X_t_mul_X[i][j] = 0;
            for(int k = 0; k < 77; k++)
                X_t_mul_X[i][j] += X[k][i-1] * X[k][j-1];
        }

    for (int i = 1; i <= 3; i++)
        for (int j = 1; j <= 1; j++){
            X_t_mul_y[i][j] = 0;
            for(int k = 0; k < 77; k++)
                X_t_mul_y[i][j] += X[k][i-1] * y[k][j-1];
        }

    gaussj(X_t_mul_X, 3, X_t_mul_y, 1);
```

- fitting에서는 (Xt X)a = Xt y 의 linear equation을 풀어준다.
  linear equation을 풀때는 gaussj를 활용한다.