

```

#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <math.h>
#include <float.h>
#define N 220
#include "my_library_v3.h"
#include "minjijo_lusolve_extended.h"

int main(void)
{
    int n; /* n個データ点数, */
    double x[N], y[N]; /* データ */
    int k, f_id[N]; /* モデル：基本関数の数k、基本関数のid */
    double sol[N] = {0}; /* 求めたのモデル係数a1,...,ak*/
    double err; /* モデルの誤差*/
    int m; /* 調べるモデル数 */
    /* データファイル */
    char *data = "example2.txt";
    // char *data="nh_covid-italy.txt";
    // char *data="nh_bb_age_weight.txt";
    // char *data="nh_bb_age_length.txt";
    // char *data="nh_fish.txt";
    // char *data="nh_wine.txt";

    char *path=
        "/Users/hernan/Documents/Shindai/Lectures/Lectures-Mac-2023/Numerical
        Methods/Class10/";

    /* ファイル名を扱う変数 */
    char fname_data[200], fname_models[200], fname_hokan[200],
        fname_out[200];
    /* ファイルポインタを扱う変数 */
    FILE *fp_models, *fp_out;

    /* データファイル名を設定する */
    sprintf(fname_data, "%s/data/%s", path, data);
    /* モデルの仕様ファイルを設定する */
    sprintf(fname_models, "%s/data/models.txt", path);
    /* 補間結果ファイルを設定する */
    sprintf(fname_out, "%s/results/out_%s", path, data);

    /* モデルの仕様ファイルを開く*/
    fp_models = fopen(fname_models, "r");
    /* モデルの仕様ファイルを開く*/
    fp_out = fopen(fname_out, "w");

```

```

printf("このプログラムは最小2乗法によって \n");
printf("y = a1*f1(x) + a2*f2(x) +...+ ak*fk(x) \n");
printf("の形の曲線をあてはめるものです. \n\n");

/** データの入力 **/
n = data_input(fname_data, x, y);
printf("%s ファイルから %d 個データ点数を読み込んだ\n", data, n);
/* 調べるモデルの数を読み込む */
fscanf(fp_models, "%d", &m);
printf("models.txtファイルから %d 個モデルを調べる\n", m);

/* models.txt ファイルで指定された各モデルを調べる */
for (int i=1; i<=m; i++)
{
    /* モデルを読み込む */
    k = read_model(fp_models, f_id);
    printf("m%d    k %d\n", i, k);
    /* モデルの係数を求める */
    compute_model(n, k, x, y, f_id, sol);
    /* モデルの誤差を求める */
    err = model_error(n, k, x, y, sol, f_id);
    /* モデルの誤差, モデルの基本関数と求めた係数を出力 */
    model_error_output(fp_out, i, err, k, sol, f_id);
    /* 補間結果ファイルを設定する */
    sprintf(fname_hokan, "%s/results/m%d_est_%s", path, i, data);
    /* 補間結果ファイル : グラフを描くための準備 (数表を出力) */
    data_output(fname_hokan, n, k, x, y, sol, f_id);
}
fclose(fp_models);
fclose(fp_out);
printf("モデル誤差と求めた基本関数の係数はファイル out_%s に保存されます\n", data);
printf("補間結果はファイル m#_est_%s に保存されます\n", data);
printf("処理の終了\n");

}

```