```
#include <stdio.h>
#include <math.h>
#include <fftw3.h>

#define NUM_POINTS 10000
#define REAL 0
#define IMAG 1

void create_input(fftwl_complex* signal) {
  /* The input is a sum of several cosines and sines with different frequencies
   * and amplitudes
   */
  int i;

  printf("Creating a signal with precision LONG DOUBLE (sizeof=%lu Bytes)\n", sizeof(long double
));

  for (i = 0; i < NUM_POINTS; ++i) {
    long double theta = (long double)i / (long double)NUM_POINTS * 2 * M_PI;

    signal[i][REAL] = 1.0 * cos(10.0 * theta) +
      2.0 * cos(20.0 * theta) +
      3.0 * cos(30.0 * theta) +
      4.0 * cos(40.0 * theta) +
      5.0 * cos(50.0 * theta);

    signal[i][IMAG] = 1.0 * sin(10.0 * theta) +
      2.0 * sin(20.0 * theta) +
      3.0 * sin(30.0 * theta) +
      4.0 * sin(40.0 * theta) +
      5.0 * sin(50.0 * theta);
  }
}

void print_magnitude(fftwl_complex* result, FILE *fp) {
  int i;
  for (i = 0; i < NUM_POINTS; ++i) {
    long double mag = sqrt(result[i][REAL] * result[i][REAL] +
                           result[i][IMAG] * result[i][IMAG]);
    fprintf(fp,"%34.25Le %34.25Le %34.25Le\n", result[i][REAL], result[i][IMAG], mag)
;
  }
}

int main() {
  FILE *fp;
  fftwl_complex signal[NUM_POINTS];
  fftwl_complex result[NUM_POINTS];

  fftwl_plan plan = fftwl_plan_dft_1d(NUM_POINTS,
                                      signal,
                                      result,
                                      FFTW_FORWARD,
                                      FFTW_ESTIMATE);

  create_input(signal);
  printf("Saving input signal...\n");
  fp = fopen("Input_FFT.dat", "w");
  print_magnitude(signal, fp);
  fclose(fp);

  fftwl_execute(plan);
  printf("Saving FFT from signal...\n");
  fp = fopen("Output_FFT.dat", "w");
  print_magnitude(result, fp);
  fclose(fp);

  fftwl_destroy_plan(plan);
  return 0;
}
```