```cpp
# include <cstdlib>
# include <cmath>
# include <iostream>
# include <iomanip>
# include <fstream>

using namespace std;

# include "cordic.hpp"
# include "cordic_ang.hpp"




int main (int argc, char * argv[]) {

  int    nIter = 3;
  int    nAngle = 1 << nIter;
  int    i;
  double *A;


  if (argc > 1 ) {
    nIter = atoi(argv[1]);
    nAngle = 1 << nIter;
  }

  cout << "nIter = " << nIter << endl;


  A = (double *) malloc((1<<nIter) * sizeof (double));

  for (i=0; i<nAngle; ++i) {
    A[i] = compute_angle(i, nIter);
  }




  // ------------------------------------------------
  //   Plot angle vectors on the unit circle
  // ------------------------------------------------
  plot_unit_circle_angle (A, nIter, nAngle);

  // ------------------------------------------------
  //   Plot residue errors at the leaf node angles
  // ------------------------------------------------
  plot_residual_errors (A, nIter, nAngle);

  // ------------------------------------------------
  //   Plot residue errors at the leaf node angles
  // ------------------------------------------------
  // calc_statistics (A, nIter, nAngle);


  // ------------------------------------------------
  //   Plot residue errors at the leaf node angles
  // ------------------------------------------------
  plot_angle_tree (nIter, nAngle);



  return 0;

}
```