The autocorrelation function for discrete signals is defined as:
\[ R_{zz}(j) = \sum_{i} z_{i} z^{*}_{i-j} \]

It is a measure of how well a signal matches a time-shifted version of itself as a function of the amount of time shift, \( j \). Autocorrelation is useful for finding repeating patterns in a signal, such as determining the presence of a periodic signal which has been buried under noise or identifying the missing fundamental frequency in a signal implied by its harmonic frequencies. (From Wikipedia: [http://en.wikipedia.org/wiki/Autocorrelation](http://en.wikipedia.org/wiki/Autocorrelation))

Write a complete console* ANSI-C program that calculates the autocorrelation for some random values and then stores the results in an array.

Specifically:
1) Declare two arrays of type `double`: The first, `Data`, has 1000 elements and will contain the data to be analyzed. (Note: the data is “real”, i.e., the imaginary part for each data point is 0.) The second array, called `R`, with 100 elements, will store the results of the autocorrelation calculations.

2) Before calculating the autocorrelation function, assign to every element in the data array random numbers, \( r_G \), that approximate a Gaussian distribution, i.e., they have a mean of 0 and a standard deviation of 1. Such values can be obtained by the following method:
\[ r_G = \left( \frac{\sum_{i=12} r_i}{\sum_{i=0} r_i} \right) - 6 \]
where \( r_i \) is a random number between 0 and 1 and can be obtained by the C-function `rand()/RAND_MAX`.

3) Finally, calculate \( R_{zz}(j) \) for \( 0 \leq j < 100 \) and store the value of each \( R_{zz}(j) \) in the corresponding array element `R[j]` declared earlier. Note: to avoid exceeding the limits of the array use the following “boundary conditions” over which to calculate \( R_{zz}(j) \):
\[ R_{zz}(j) = \sum_{i=j}^{i<1000-100+j} z_{i} z^{*}_{i-j} \]

Final Note: in your program do **not** include any input statements such as `scanf` or any include files. You will be graded on program logic and syntax mistakes. You may use local or global variables.

Console* means you must not include a LabWindows Graphical User Interface (GUI) or any (callback) function for it.
Solution

#include <ansi_c.h>
#define MAX 1000
#define J 100

double Data[MAX], R[J];

main()
{
  int i, j;

  //Assign Random Values
  for( i = 0; i < MAX; i++)
  {
    for( j = 0; j < 12; j++)
      //Since Data[i] is global, no need to initialize
      Data[i] = Data[i] + (double) rand()/RAND_MAX;
    Data[i] = Data[i] - 6.0;
  }

  //Calculate Autocorrelation
  for( j = 0; j < J; j++)
  {
    for( i = j; i < MAX - J + j; i++)
      R[j] = R[j] + Data[i]*Data[i-j];
  }
}