Using Jacobian or Gauss-Seidel Methods to solve Ax = b.

1. First google Gauss-Seidel method for solving Ax = b. Read and understand this iterative method. And compare with Jacobi's method.
2. Write a function like following:

```
void GaussSeidel(int n, double * A, double * b, double * x) {
        ……..
}
```

   Please use Gauss-Seidel method to solve x, supposing A and b are given.
3. Change the iteration steps number as following

```
#define ITER 20
```

   Later you can change it to 40, 80, 160.
4. Write the main function to compare Jacobi's method and Gauss-Seidel method.
   a. Declare matrix A, vector x and b as following:

```
double A[20][20];
double x[20];
double b[20];
```

      Diagonal elements A[i][i]= 2.0;
      The off diagonal elements A[i][i+1] = -1.0; A[i+1][i] = -1.0;
      All other elements of A are zeros.
      All elements of x are zeros;
      All elements of b are zeros, except b[0] = b[19] = 1.0;
   b. We know the exact solution **sol** = {1, 1, …, 1}. So please run your code and fill out the following table:

ITER	Error from Jacobi's Method	Convergence Rate	Error from Gauss-Seidel Method	Convergence Rate
20				
40				
80				
160				

Note that here Error is calculated by the vector Euclidian distance from your solution to the exact solution **sol**.

Please fill out this table in your report and submit together with your source code.