```
%Inputs:
%A: Matrix
%b: Vector
%tot_it: Number of iterations
%Output:
%:x The solution are tot_it iterations/sweeps

function x = my_jacobi(A, b, tot_it)

    % Creating x(0)
    width_of_A = size(A,2);
    x_prev = zeros(width_of_A, 1);

    D_inv = inv(diag(diag(A)));
    C_l = (tril(A) - diag(diag(A))) * -1;
    C_u = (triu(A) - diag(diag(A))) * -1;


    % Pulling two non-changing terms out of for loop for speed.
    term_1 = D_inv * (C_l + C_u);
    term_2 = D_inv * b;

    for i = 1:tot_it
        x_iter = (term_1 * x_prev) + term_2;
        x_prev = x_iter;
    end


    x = x_iter;
```

```
Not enough input arguments.
Error in my_jacobi (line 11)
    width_of_A = size(A,2);
```

Published with MATLAB® R2018a