```
function [Xf operations] = gauss_siedel(A, n, B)
    % Declaring XI, Xf, and operations
    operations = 0;
    Xg = zeros(n,1);
    Xf = zeros(n,1);

    tol = 10^(-10);

    while 1>0
        for i = 1:n
            sum = 0;
            % Sum of updated elements
            for j1 = 1:i-1
                sum = sum + A(i, j1)*Xf(j1);
                operations = operations + 2;
            end
            % Sum of non-updated previous elements
            for j2 = i+1:n
                sum = sum + A(i, j2)*Xg(j2);
                operations = operations + 2;
            end

            % Setting Xf values
            Xf(i) = (B(i) - sum)/A(i,i);
            operations = operations + 2;
        end

        isConverging = 1; % like boolean value for checking convergence

        % checking convergence for every element
        for k = 1:n
            if abs((Xf(k) - Xg(k))/Xf(k)) > tol
                isConverging = 0;
            end
        end

        if(isConverging == 1) % break if every element is converging
            break;
        end

        Xg = Xf;
    end
    return
end
```