```
%% TRIAD Algorithm

function [C_ba] = TRIAD(s1_a, s2_a, s1_b, s2_b)

    % normalizing input measurement vectors
    s1_a = s1_a / norm(s1_a);
    s2_a = s2_a / norm(s2_a);
    s1_b = s1_b / norm(s1_b);
    s2_b = s2_b / norm(s2_b);

    % defining w1 == s1 vector (normalized)
    w1_a = s1_a;
    w1_b = s1_b;

    % defining w2 == s1 x s2 (normalized cross product)
    w2_a = crossm(s1_a) * s2_a / norm( crossm(s1_a) * s2_a );
    w2_b = crossm(s1_b) * s2_b / norm( crossm(s1_b) * s2_b );

    % defining w3 == w1 x w2 (vectors are already normalized)
    w3_a = crossm(w1_a) * w2_a;
    w3_b = crossm(w1_b) * w2_b;

    % computing DCM estimate for attitude determination
    C_ba = [w1_b  w2_b  w3_b] * [w1_a  w2_a  w3_a]';

end
```