```
function [success, q_connect, Ta, Tb] = RRTConnect(q_start, q_goal,
 ws, max_nodes, step_len, step_size)
    % Return configuration path from c0 to cf in ws using RRT Connect
    % c0 - The initial configuration
    % cf - The final configuration
    % ws - Workspace of with obstacles
    % max_nodes - Max # nodes to generate before auto-termination

    % Initialize trees
    Ta = KdTree(KdNode(q_start, NullKdNode(), NullKdNode()));
    Tb = KdTree(KdNode(q_goal, NullKdNode(), NullKdNode()));
    success = false;
    a_first = true;

    for i=1:max_nodes
        % Generate random config
        q_rand = Config([rand_range(-pi, pi, 1),...
                         rand_range(-pi, pi, 1),...
                         rand_range(-pi, pi, 1)]);

        % Attempt to connect random config to tree
        if a_first
            [result, q_target] = RRTExtendSingle(Ta, q_rand, ws,
 step_len, step_size);
        else
            [result, q_target] = RRTExtendSingle(Tb, q_rand, ws,
 step_len, step_size);
        end
        if result
            % If extend successful, attempt to connect other tree to
 new
            % node of current tree
            if a_first
                [result2, q_connect] = RRTExtendMultiple(Tb, q_target,
 ws, step_len, step_size);
            else
                [result2, q_connect] = RRTExtendMultiple(Ta, q_target,
 ws, step_len, step_size);
            end
            success = result2;
        end
        if success
            % If connected trees, done
            break;
        end
        a_first = ~a_first;
    end

    if ~success
        warning('Terminated before connecting trees');
    end
```

*Published with MATLAB® R2020a*