```
function x = ScaledPartialPivoting(n, A)
%  Use Scaled Partial Pivoting to find the solution to a system of
%  equations
%  Input: A - augmented matrix representing the system of equations
%         n - number of unknowns
%  Output: x - unique solution

% Initialize scaled factor
s = zeros(1, n);

% Initialize multipliers
m = zeros(n, n);

% Initialize row pointers
NROW = zeros(1, n);

% Set scaled factor and row pointers
for i=1:n

    % Get scaled factor
    s(i) = getScaledFactor(A, n, i);

    % Check for unique solution
    if s(i) == 0
        disp('no unique solution exists');
        return
    else
        % Set row pointer
        NROW(i) = i;
    end
end

for i=1:n-1

    % Set pivot
    p = getPivotScaled(A, s, NROW, n, i);

    % Check for unique solution
    if A(NROW(p), i) == 0
        disp('no unique solution exists');
        return
    end

    % Swap ith and pth row

    if NROW(i) ~= NROW(p)
        NCOPY = NROW(i);
        NROW(i) = NROW(p);
        NROW(p) = NCOPY;
    end

    for j=(i+1):n
```

```
        % Calculate row modifier
        m(NROW(j), i) = A(NROW(j), i) ./ A(NROW(i), i);

        % Modify row
        A(NROW(j),:) = A(NROW(j),:) - m(NROW(j), i) .* A(NROW(i),:);
    end

end

% Check for unique solution
if A(NROW(n), n) == 0
    disp('no unique solution exists');
    return
end

% Use backward substitution to find the solution
x = zeros(1, n);
x(n) = A(NROW(n), n+1) ./ A(NROW(n), n);

i = n-1;
while i >= 1

    sum = 0;
    for j=i+1:n
        sum = sum + A(NROW(i), j) .* x(j);
    end

    x(i) = (A(NROW(i), n+1) - sum) ./ A(NROW(i), i);

    i = i - 1;
end


end
```

```
Not enough input arguments.

Error in ScaledPartialPivoting (line 9)
s = zeros(1, n);
```

Published with MATLAB® R2018b