```
% Simplex Algorithm Function
function [x_min, f_min, n_iter, T] = simplex(f,A,b)
    %   Syntax  : [x_min, f_min, n_iter, T] = simplex(f,A,b)
    %
    %
    %   Purpose : Solves the problem
    %
    %               min      f'*x
    %               st.      A*x    <= b
    %                        x >= 0
    %   Assumes no equality constraints and x_i >= 0 forall i
    arguments
        f (:,1) double {mustBeNumeric,mustBeReal}
        A (:,:) double {mustBeNumeric,mustBeReal} = []
        b (:,1) double {mustBeNumeric,mustBeReal} = []
    end
    % Assuming everything inputed is good....
    max_iter = 20;
    % Setup
    n = size(f,1);
    num_s = size(A,1);% 0;

    T = [[A;f'],eye(size(A,1)+1),[b;0]];
    disp('Initial T = ');
    disp(T);

    n_iter = 0;
    ratios = zeros(size(T,1)-1,1);
    while any(T(end,1:end-1)<0) %Keeps going until optimal (final row
 >= 0)
        n_iter = n_iter + 1;
        % complicated way to find smallest value index
        min_val = min(T(end,1:end-1));
        [~, min_col] = find(T(end,1:end-1)==min_val,1,'first');
        % find pivot row
        for row = 1:(size(T,1)-1)
            if T(end,row) >= 0
                if T(row, min_col) > 0
                    ratios(row) = T(row,end) / T(row,min_col);
                else
                    ratios(row) = inf;
                end
            else
                ratios(row) = inf;
            end
        end
        min_val = min(ratios);
        [min_row,~] = find(ratios==min_val,1,'first');

        % Pivoting
        new_T = zeros(size(T));
        new_T(min_row,:) = T(min_row,:)/T(min_row,min_col);
```

```
            for row = 1:size(T,1)
                if row ~= min_row
                    new_T(row,:) = T(row,:) ...
                        - T(row,min_col) * new_T(min_row,:);
                end
            end
            T = new_T;

            if n_iter >= max_iter
                error('too many iterations')
            end

        end
        disp('Final T = ');
        disp(T);


        % Calculating the basic variables
        j = 1;
        row = zeros(size(T,1),1);
        col = zeros(size(T,1),1);
        for i = 1:size(T,2)
            if nnz(T(:,i)) == 1
                col(j) = i;
                row(j) = find(T(:,i),1);
                j = j+1;
            end
        end

        % Solving for x values
        X = zeros([n,n+1]);
        for i = 1:n
            if col(i) <= n
                X(i,col(i)) = 1;
                X(i,end) = T(i,end);
            else
                X(i,:) = [A(i,:),b(i) - T(i,end)] ;
            end
        end
        X = rref(X);
        x_min = X(:,end);
        f_min = f'*x_min;
    end

    Initial T =
         2     1     1     1     0     0     0    15
        -1     2    -1     0     1     0     0     7
         1     5     5     0     0     1     0    25
         1     2    -7     0     0     0     1     0

    Final T =
         2     1     1     1     0     0     0    15
         1     3     0     1     1     0     0    22
        -9     0     0    -5     0     1     0   -50
```

```
    15     9     0     7     0     0     1   105

x_min =

     0
     0
    15

f_min =

  -105

n_iter =

     1

T =

     2     1     1     1     0     0     0    15
     1     3     0     1     1     0     0    22
    -9     0     0    -5     0     1     0   -50
    15     9     0     7     0     0     1   105
```

Published with MATLAB® R2020b