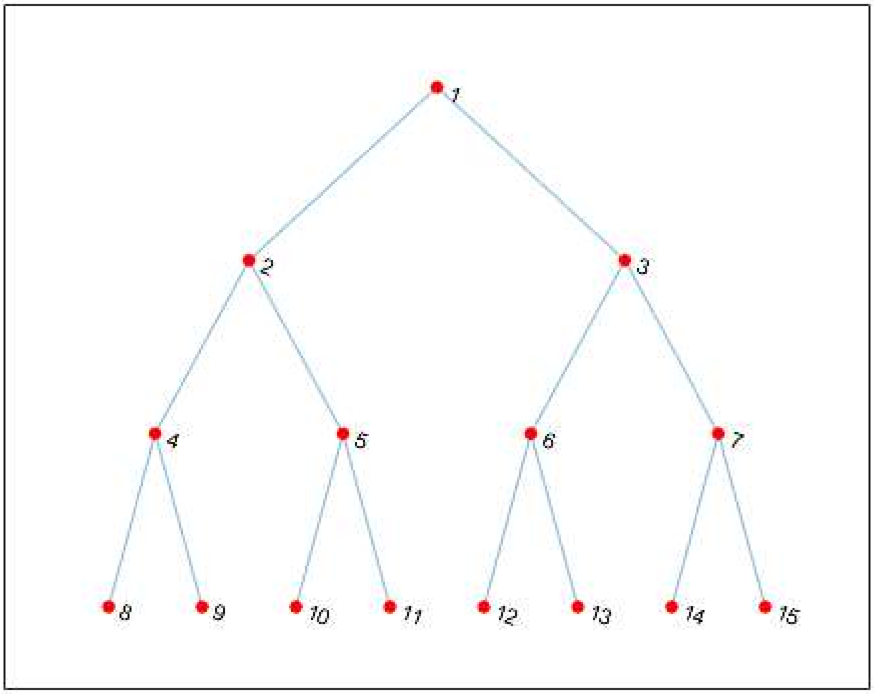
```
% Number of levels in the binary tree
L = 4; % Change this value as per your requirement

% Number of nodes
N = 2^L - 1;

% Weights for each edge
weights = rand(N, 1) * 10;

% Adjacency matrix
adjMatrix = zeros(N);
for i = 1:(N-1)/2
    adjMatrix(i, 2*i) = weights(2*i);
    adjMatrix(2*i, i) = weights(2*i);
    adjMatrix(i, 2*i + 1) = weights(2*i + 1);
    adjMatrix(2*i + 1, i) = weights(2*i + 1);
end

G = graph(adjMatrix);

% Visualization
nodeColor = 'red';
figure;
plot(G, 'NodeColor', nodeColor);

converged = false;
iterations = 0;
routingOverhead = 0;

% Initialize routing tables
dvrTables = cell(N, 1);
for i=1:N
    dvrTables{i} = inf(N, 2);
    dvrTables{i}(:, 1) = 1:N;
    dvrTables{i}(i, 2) = 0; % Distance to itself is 0
    if i ~= 1  % Not the root node
        parent = floor(i/2);
        dvrTables{i}(parent, 2) = adjMatrix(i, parent);  % Direct cost to parent
    end
    if 2*i <= N  % Left child exists
        dvrTables{i}(2*i, 2) = adjMatrix(i, 2*i);  % Direct cost to left child
    end
    if 2*i + 1 <= N  % Right child exists
        dvrTables{i}(2*i + 1, 2) = adjMatrix(i, 2*i + 1);  %Direct cost to right child
    end
end

tic;

% Run until convergence
while ~converged
    converged = true;  %Assume convergence
    iterations = iterations + 1;

    oldTables = dvrTables;


    for i=1:N
        neighbors = [];
        if i ~= 1  % Not the root node
            parent = floor(i/2);
            neighbors = [neighbors, parent];  %Add parent to neighbors
        end
        if 2*i <= N  %Left child exists
            neighbors = [neighbors, 2*i];  %Add left child to neighbors
        end
        if 2*i + 1 <= N  %Right child exists
            neighbors = [neighbors, 2*i + 1];  %Add right child to neighbors
        end
        for neighbor = neighbors
            routingOverhead = routingOverhead + 1;

            %Update neighbor's routing table
            for j=1:N
                costThroughMe = dvrTables{i}(j, 2) + adjMatrix(i, neighbor);
                if costThroughMe < dvrTables{neighbor}(j, 2)
                    dvrTables{neighbor}(j, 2) = costThroughMe;
                    converged = false;
                end
            end
        end
    end


    for i=1:N
        if ~isequal(oldTables{i}, dvrTables{i})
            converged = false;
            break;
        end
    end
end

convergenceTime = toc;

%Display routing tables
for i=1:N
    disp(['Routing table for node ' num2str(i)]);
    disp(dvrTables{i});
end

%Convergence Time
disp(['Converged in ' num2str(iterations) ' iterations.']);
disp(['Convergence time: ' num2str(convergenceTime) ' seconds.']);

%Routing Overhead
disp(['Total routing overhead: ' num2str(routingOverhead) ' messages.']);
```

```
Routing table for node 1
    1.0000         0
    2.0000    3.0018
    3.0000    4.0139
    4.0000   11.3355
    5.0000    7.0381
    6.0000    7.9156
    7.0000    7.6184
    8.0000   12.7380
    9.0000   13.9368
   10.0000    7.9063
   11.0000   11.3321
   12.0000   10.4885
   13.0000   10.8912
   14.0000   11.8669
   15.0000    8.8104

Routing table for node 2
    1.0000    3.0018
    2.0000         0
    3.0000    7.0157
    4.0000    8.3336
    5.0000    4.0363
    6.0000   10.9175
    7.0000   10.6202
    8.0000    9.7362
    9.0000   10.9349
   10.0000    4.9044
   11.0000    8.3303
   12.0000   13.4903
   13.0000   13.8930
   14.0000   14.8688
   15.0000   11.8123

Routing table for node 3
    1.0000    4.0139
    2.0000    7.0157
    3.0000         0
    4.0000   15.3493
    5.0000   11.0520
    6.0000    3.9018
    7.0000    3.6045
    8.0000   16.7519
    9.0000   17.9507
   10.0000   11.9202
   11.0000   15.3460
   12.0000    6.4746
   13.0000    6.8773
   14.0000    7.8531
   15.0000    4.7966

Routing table for node 4
    1.0000   11.3355
    2.0000    8.3336
    3.0000   15.3493
    4.0000         0
    5.0000   12.3699
    6.0000   19.2511
    7.0000   18.9538
    8.0000    1.4026
    9.0000    2.6013
   10.0000   13.2381
   11.0000   16.6639
   12.0000   21.8239
   13.0000   22.2267
   14.0000   23.2024
   15.0000   20.1459

Routing table for node 5
    1.0000    7.0381
    2.0000    4.0363
    3.0000   11.0520
    4.0000   12.3699
    5.0000         0
    6.0000   14.9538
    7.0000   14.6565
    8.0000   13.7725
    9.0000   14.9712
   10.0000    0.8682
   11.0000    4.2940
   12.0000   17.5266
   13.0000   17.9293
   14.0000   18.9051
   15.0000   15.8486

Routing table for node 6
    1.0000    7.9156
    2.0000   10.9175
    3.0000    3.9018
    4.0000   19.2511
    5.0000   14.9538
    6.0000         0
    7.0000    7.5062
    8.0000   20.6537
    9.0000   21.8524
   10.0000   15.8219
   11.0000   19.2477
   12.0000    2.5728
   13.0000    2.9756
   14.0000   11.7548
   15.0000    8.6983

Routing table for node 7
    1.0000    7.6184
    2.0000   10.6202
    3.0000    3.6045
    4.0000   18.9538
    5.0000   14.6565
    6.0000    7.5062
    7.0000         0
    8.0000   20.3564
    9.0000   21.5551
   10.0000   15.5246
   11.0000   18.9505
   12.0000   10.0791
   13.0000   10.4818
   14.0000    4.2486
   15.0000    1.1921

Routing table for node 8
    1.0000   12.7380
    2.0000    9.7362
    3.0000   16.7519
    4.0000    1.4026
    5.0000   13.7725
    6.0000   20.6537
    7.0000   20.3564
    8.0000         0
    9.0000    4.0039
   10.0000   14.6406
   11.0000   18.0664
   12.0000   23.2265
   13.0000   23.6292
   14.0000   24.6050
   15.0000   21.5485

Routing table for node 9
    1.0000   13.9368
    2.0000   10.9349
    3.0000   17.9507
    4.0000    2.6013
    5.0000   14.9712
    6.0000   21.8524
    7.0000   21.5551
    8.0000    4.0039
    9.0000         0
   10.0000   15.8394
   11.0000   19.2652
   12.0000   24.4252
   13.0000   24.8280
   14.0000   25.8037
   15.0000   22.7472

Routing table for node 10
    1.0000    7.9063
    2.0000    4.9044
    3.0000   11.9202
    4.0000   13.2381
    5.0000    0.8682
    6.0000   15.8219
    7.0000   15.5246
    8.0000   14.6406
    9.0000   15.8394
   10.0000         0
   11.0000    5.1621
   12.0000   18.3947
   13.0000   18.7975
   14.0000   19.7732
   15.0000   16.7167

Routing table for node 11
    1.0000   11.3321
    2.0000    8.3303
    3.0000   15.3460
    4.0000   16.6639
    5.0000    4.2940
    6.0000   19.2477
    7.0000   18.9505
    8.0000   18.0664
    9.0000   19.2652
   10.0000    5.1621
   11.0000         0
   12.0000   21.8206
   13.0000   22.2233
   14.0000   23.1990
   15.0000   20.1425

Routing table for node 12
    1.0000   10.4885
    2.0000   13.4903
    3.0000    6.4746
    4.0000   21.8239
    5.0000   17.5266
    6.0000    2.5728
    7.0000   10.0791
    8.0000   23.2265
    9.0000   24.4252
   10.0000   18.3947
   11.0000   21.8206
   12.0000         0
   13.0000    5.5484
   14.0000   14.3277
   15.0000   11.2711

Routing table for node 13
    1.0000   10.8912
    2.0000   13.8930
    3.0000    6.8773
    4.0000   22.2267
    5.0000   17.9293
    6.0000    2.9756
    7.0000   10.4818
    8.0000   23.6292
    9.0000   24.8280
   10.0000   18.7975
   11.0000   22.2233
   12.0000    5.5484
   13.0000         0
   14.0000   14.7304
   15.0000   11.6739

Routing table for node 14
    1.0000   11.8669
    2.0000   14.8688
    3.0000    7.8531
    4.0000   23.2024
    5.0000   18.9051
    6.0000   11.7548
    7.0000    4.2486
    8.0000   24.6050
    9.0000   25.8037
   10.0000   19.7732
   11.0000   23.1990
   12.0000   14.3277
   13.0000   14.7304
   14.0000         0
   15.0000    5.4407

Routing table for node 15
    1.0000    8.8104
    2.0000   11.8123
    3.0000    4.7966
    4.0000   20.1459
    5.0000   15.8486
    6.0000    8.6983
    7.0000    1.1921
    8.0000   21.5485
    9.0000   22.7472
   10.0000   16.7167
   11.0000   20.1425
   12.0000   11.2711
   13.0000   11.6739
   14.0000    5.4407
   15.0000         0
```

```
Converged in 4 iterations.
Convergence time: 0.0047371 seconds.
Total routing overhead: 112 messages.
```



Published with MATLAB® R2022a