CSE 514 Lab 2- Implementing a Routing Algorithm

Overview:

The routing algorithm for 4 nodes was implemented, and the corresponding minimum costs and distance tables were calculated for each node.

Technical specifications:

In the initialization functions – rinit0, rinit1, rinit2 and rinit3, the distance tables and the minimum costs were initialized according to the values in the graph. After the initialization, function tolayer2 was called and the minimum costs were sent to the other nodes.

Following are the functions for rinit0 and rupdate0 :

```
//function to initialise the distance table and the min costs for node 0
void rtinit0()
{
 printf("\n\n Function rinit0 is called at t=%f.\n",clocktime);
 int i,j;
 for(i =0; i<4;i++)
{
 for(j =0; j<4;j++)
{
 dt0.costs[i][j] = 999;
}}
 dt0.costs[0][0] =0;
 dt0.costs[1][1] =1;
 dt0.costs[2][2] =3;
 dt0.costs[3][3] =7;

 node0.sourceid = 0;
 node0.destid = 1;
 node0.mincost[0] = 0;
 node0.mincost[1] = 1;
 node0.mincost[2] = 3;
 node0.mincost[3] = 7;

 for (i =1; i<4;i++)
        {
        if (i != node0.sourceid)
        {
          node0.destid = i;
          tolayer2(node0);
```

```
        } } }
```

Rupdate0: This would check if the sum of the link cost to the neighboring node of node 0 and the mincost of the received packet is lesser than the corresponding value in the distance table. If the value is lesser, then the distance table is updated. The same check is performed for the node 0's mincost value to the source in the received packet, if lesser, the min cost is updated.

Finally, if the distance tables or the minimum costs are updated, the tolayer2 function is called to update the remaining nodes.

```
//function to update the distance table and the min costs for node 0
void rtupdate0(rcvdpkt)
  struct rtpkt *rcvdpkt;
{
        printf("\n\n Function rupdate0 is called at t=%f.\n",clocktime);
        printf("\n\n The source node to Node %d is Node %d.\n\n",rcvdpkt->destid,rcvdpkt->sourceid);

        int mincost_recalc,dtbl_recalc,i,source =rcvdpkt->sourceid;

        for (i = 0;i<4;i++){
                if (cvo[source] + rcvdpkt->mincost[i] < dt0.costs[i][source])
                {
                        dt0.costs[i][source] = cvo[source] + rcvdpkt->mincost[i];
                        dtbl_recalc =1;
                }
                if ( cvo[source] + rcvdpkt->mincost[i]<node0.mincost[i])
                {
                        mincost_recalc =1;
                        node0.mincost[i] =cvo[source] + rcvdpkt->mincost[i];
                }
        }
//checking if there is any change to the distance tables
        if (dtbl_recalc == 1)
        {
                printf("\n\n The distance table of node %d has been recalculated.\n\n", rcvdpkt->destid);
        }

        printdt0(&dt0);
//sending the updated min costs to the other tables
         if(dtbl_recalc ==1 || mincost_recalc == 1){
                for (i =0; i<4;i++)
          {
                  if(node0.sourceid !=i){
                    printf("\n\n Sending messages to Node %d from Node %d.\n\n", i, node0.sourceid);
                     node0.destid = i;
                     tolayer2(node0);}
                  }
          }
}
```

Similar functions were written for nodes 1, 2 and 3, with the exception that for nodes 1 and 3 which weren't connected, the updated distance tables of 1 and 3 weren't sent to each other.

Sample Output:

The final values for the distance tables and the minimum costs are as follows:

```
The source node to node 3 is node 2.

            via
  D3 |    0     2
 ----|-----------
    0|    7     4
dest 1|    8     3
    2|    9     2
```

```
Function rupdate0 is called at t=20.164196.
The source node to Node 0 is Node 2.

               via
  D0 |    1     2    3
 ----|----------------
    1|    1     4   10
dest 2|    2     3    9
    3|    4     5    7
MAIN: rcv event, t=20.621, at 1 src: 0, dest: 1, contents:   0   1   2   4

Function rupdate1 is called at t=20.621054.n The source node to Node 1 is Node 0.

            via
  D1 |    0     2
 ----|-----------
    0|    1     3
dest 2|    3     1
    3|    5     3
MAIN: rcv event, t=21.312, at 2 src: 1, dest: 2, contents:   1   0   1   3


Function rupdate2 is called at t=21.311695.


The source node to node 2 is node 1.

               via
  D2 |    0     1    3
 ----|----------------
    0|    3     2    6
dest 1|    4     1    5
    3|    7     4    2
MAIN: rcv event, t=21.429, at 0 src: 3, dest: 0, contents:   4   3   2   0
```

```
 Function rupdate1 is called at t=21.728632.n

 The source node to Node 1 is Node 2.

            via
   D1 |    0     2
  ----|-----------
     0|    1     3
dest 2|    3     1
     3|    5     3
MAIN: rcv event, t=22.431, at 0 src: 2, dest: 0, contents:   2   1   0   2


 Function rupdate0 is called at t=22.431103.
```