117A1091
BE CE D

Experiment No. 3

Aim : To implement A-Star Algorithm

Program :

```
#include <bits/stdc++.h>

using namespace std;
#define inf 1e18


int main()
{
   int n,m,u,v,w;
   cin>>n>>m;
   vector<int>h;

   vector<pair< long int, long int>>adj[n+1];
    vector< long int>ans;
   vector< long int>dist(n+1,inf),total_dist(n+1,inf);

   dist[1]=0;//Assuming start vertex is 1

   for(int i=1;i<=m;i++)
   {
      cin>>u>>v>>w;
      adj[u].push_back({v,w});
      adj[v].push_back({u,w});
   }
   h.assign(n+1,0);
   for(int i=0;i<n;i++)
   {
      cin>>h[i+1];
   }

   total_dist[1]=h[1];

   set<pair< long int, long int>>s;
   vector<int>parent(n+1,-1);

   s.insert(make_pair(total_dist[1],1));
    long int curr_dist,d,curr;
```

```
    while(!s.empty())
    {
        curr= (*s.begin()).second;

        cout<<"current node:"<<curr<<" total_distance:"<<total_dist[curr]<<"\n";

        curr_dist=dist[curr];
        s.erase(make_pair( (*s.begin()).first ,(*s.begin()).second));

        for( pair< long int, long int> edge: adj[curr])
        {
            //Update distance
            if(curr_dist + edge.second < dist[edge.first])
            {
                dist[edge.first]= curr_dist + edge.second;
            }



            if(curr_dist + edge.second + h[edge.first] < total_dist[edge.first])
            {
                total_dist[edge.first]= curr_dist + edge.second + h[edge.first];
                parent[edge.first]=curr;
                s.insert(make_pair(total_dist[edge.first],edge.first));


                cout<<"node:"<<edge.first<<" td: "<<curr_dist<<"+"<<edge.second<<"+"<<
h[edge.first]<<"="<<total_dist[edge.first]<<"\n";

            }

        }



    }


    for (int i=1;i<=n;i++)
        cout<<i<<" : "<<total_dist[i]<<"\n";

        return 0;
}
```

Input :

8 10
1 2 1
1 3 2
2 4 7
2 5 4
3 6 7
3 7 1
4 8 3
5 8 2
6 8 5
7 8 12
0 5 6 4 15 5 8 0

Output :

```
current node:1 total_distance:0
node:2 td: 0+1+5=6
node:3 td: 0+2+6=8
current node:2 total_distance:6
node:4 td: 1+7+4=12
node:5 td: 1+4+15=20
current node:3 total_distance:8
node:6 td: 2+7+5=14
node:7 td: 2+1+8=11
current node:7 total_distance:11
node:8 td: 3+12+0=15
current node:4 total_distance:12
node:8 td: 8+3+0=11
current node:8 total_distance:11
current node:6 total_distance:14
current node:8 total_distance:11
current node:5 total_distance:20
node:8 td: 5+2+0=7
current node:8 total_distance:7
1 : 0
2 : 6
3 : 8
4 : 12
5 : 20
6 : 14
7 : 11
8 : 7
```

Conclusion : Hence, A-Star Algorithm was implemented successfully