**Name**: Sakshi Shewale

**Class**: TY -CS   **Div**: D

**Roll no**:24

**Assignment 3:** Implementation of Informed strategies-A\* algorithm

**Code**:

#include <bits/stdc++.h>

using namespace std;

typedef pair<int, int> iPair;

struct node{

node(int a, int b)

{

dest = a;

weight = b;

}

int dest;

int weight;

};

void astar(int s, int d, vector<node> v[], int visited[], int parent[], int heu[], int dist[], int n)

{

priority\_queue<iPair, vector<iPair>, greater<iPair> > pq;

pq.push(make\_pair(heu[s], s));

dist[s] = 0;

while (!pq.empty())

{

int p\_s = pq.top().second;

if (p\_s == d)

{

return;

}

pq.pop();

if (!visited[p\_s])

{

for (int i = 0; i < v[p\_s].size(); i++)

{

if (!visited[v[p\_s][i].dest])

{

int f = v[p\_s][i].weight + heu[v[p\_s][i].dest];

pq.push(make\_pair(f, v[p\_s][i].dest));

dist[v[p\_s][i].dest] = dist[p\_s] + v[p\_s][i].weight;

parent[v[p\_s][i].dest] = p\_s;

}

}

visited[p\_s] = 1;

}

}

}

int main()

{

int total\_nodes = 7;

vector<node> v[total\_nodes];

int no\_of\_edges = 9;

v[0].push\_back(node(1, 4));

v[0].push\_back(node(2, 3));

v[1].push\_back(node(3, 5));

v[1].push\_back(node(4, 12));

v[2].push\_back(node(4, 10));

v[2].push\_back(node(5, 7));

v[3].push\_back(node(6, 16));

v[4].push\_back(node(6, 5));

v[5].push\_back(node(4, 2));

int visited[total\_nodes];

int parent[total\_nodes];

int heu[total\_nodes];

int dist[total\_nodes];

int heruristic\_values[] = {14, 12, 11, 11, 4, 6, 0};

for (int i = 0; i < total\_nodes; i++)

{

heu[i] = heruristic\_values[i];

visited[i] = 0;

parent[i] = i;

dist[i] = INT\_MAX;

}

int start = 0;

int goal = 6;

astar(start, goal, v, visited, parent, heu, dist, total\_nodes);

int cur = goal;

cout << endl;

cout << "Path from " << start << " to " << goal << " is " << endl;

stack<int> path;

do

{

path.push(cur);

cur = parent[cur];

} while (cur != 0);

path.push(start);

while (!path.empty())

{

cout << path.top();

path.pop();

if (!path.empty())

cout << " -> ";

}

return 0;

}

**Output**:

