STATE UNIVERSITY OF BANGLADESH

Course Code: CSE-0408
Course Name: Artificial Intelligence lab
Semester: Summer 2021

Submitted to:
Khan Md. Hasib
Lecturer,
Department of CSE
State University of Bangladesh

Submitted By:
Name: Shawon Mia
ID: UG02-44-17-025
Batch: 44
State University of Bangladesh

Question :

Write a program in any language [C, C++, JAVA, PYTHON] to solve the 8 -Puzzle problem using heuristic functions.

Solution :

```
#include<bits/stdc++.h>
using namespace std;

struct state{
	int grid[3][3]; int cost; int px,py,x,y,l = 0;
};

	int row[4] = { 1, 0, -1, 0 };
	int col[4] = { 0, -1, 0, 1 };

	bool operator < (state a, state b){
	return a.cost+a.l > b.cost+b.l;
}
int costCalculate(int current_state[3][3], int final_state[3][3]){
	int c = 0;
	 for(int i=0;i<3;i++)
	  for(int j=0;j<3;j++)if(current_state[i][j] && current_state[i][j] != final_state[i][j])
c++;
    return c;
}
bool isSafe(int x, int y){
  return (x>=0 && x<3 && y>=0 && y<3);
}

 void solution(int initial[3][3], int final_state[3][3], int x, int y){
	priority_queue<state>pq;

	state node;
	node.x = x; node.y = y; node.l = 0;
	node.px = -1; node.py = -1;
	node.cost = costCalculate(initial, final_state);

	for(int i=0;i<3;i++)
	 for(int j=0;j<3;j++) node.grid[i][j] = initial[i][j];
	pq.push(node);
	int f = 0;
```

```
        while(!pq.empty()){
        f = 0; node = pq.top(); pq.pop();
        x = node.x; y = node.y;
        for(int k=0; k<4; k++){
        int positionX = x+row[k];
        int positionY = y+col[k];
        state child;
        if(isSafe(positionX, positionY)){
          if(positionX == node.px && positionY == node.py) continue;

        for(int i=0;i<3;i++)
          for(int j=0;j<3;j++) child.grid[i][j] = node.grid[i][j];

        swap(child.grid[x][y], child.grid[positionX][positionY]);

        child.px = node.x; child.py = node.y; child.x = positionX; child.y = positionY;
child.l =node.l + 1;

                child.cost = costCalculate(child.grid, final_state);

                if(child.cost == 0){
                    printf("Final State: \n");
                for(int i=0;i<3;i++){
                     for(int j=0;j<3;j++) cout<<child.grid[i][j]<<" ";
                cout<<endl;
         }
         f = 1;
         break;
     }
        pq.push(child);
    }
  }
   if(f) break;
 }
}
int main(){

        int init[3][3] = {{7, 2, 3},{4 ,6 , 5},{1 ,8 ,0}};
        int goal[3][3] = {{1, 2, 3},{4, 5, 6},{7, 8, 0}};
        int x = 2, y = 2;
```

```
cout<<"Initial State:"<<endl;
for(int i=0;i<3;i++){
     for(int j=0;j<3;j++) cout<<init[i][j]<<" ";
    cout<<endl;
}cout<<endl;

 solution(init, goal, x, y);
```

Output: