Main code “code.m”

p = 1;

q = 0;

prompt = 'Enter the initial position of the puzzle - ';

Nodes(:,:,1) = input(prompt);

%% Nodes(:,:,1) will save the initial position of the matrix.

NodeInfo(:,:,p)= [p,q,0];

%% NodeInfo will provide information about any of the generated node in the following format [Node#, Parent node#,0]

q = 1;

while p <= 100000

%% The while loop will terminate when the number of nodes reaches to 100000.

[z] = findzero(Nodes(:,:,q));

%% findzero is a pre-defined function which will return the index of the zeroth element in the matrix.

templ = zeros(3);

[templ] = moveleft(Nodes(:,:,q),z);

%% moveleft is a predefined function which will move the zeroth element one step in the left direction.

%% Following portion compares the newly generated node by the moveleft function with all of the stored nodes, if the newly generated node is different then it will be saved as a new node.

t = 0;

for i = 1:p

c = (templ == Nodes(:,:,i));

x = 0;

for j = 1:9

if c(j) == 1

x = x+1;

end

end

if x == 9

t = t+1;

end

x = 0;

end

if t == 0

if p < 100000

p = p+1;

Nodes(:,:,p) = templ;

NodeInfo(:,:,p) = [p,q,0];

end

end

t = 0;

tempu = zeros(3);

[tempu] = moveup(Nodes(:,:,q),z);

%% moveup is a predefined function which will move the zeroth element one step in the up direction.

%% Following portion compares the newly generated node by the moveup function with all of the stored nodes, if the newly generated node is different than it will be saved as a new node.

t = 0;

for i = 1:p

c = (tempu == Nodes(:,:,i));

x = 0;

for j = 1:9

if c(j) == 1

x = x+1;

end

end

if x == 9

t = t+1;

end

x = 0;

end

if t == 0

if p < 100000

p = p+1;

Nodes(:,:,p) = tempu;

NodeInfo(:,:,p) = [p,q,0];

end

end

t = 0;

tempr = zeros(3);

[tempr] = moveright(Nodes(:,:,q),z);

%% moveright is a predefined function which will move the zeroth element one step in the right direction.

%% Following portion compares the newly generated node by the moveright function with all of the stored nodes, if the newly generated node is different than it will be saved as a new node.

t = 0;

for i = 1:p

c = (tempr == Nodes(:,:,i));

x = 0;

for j = 1:9

if c(j) == 1

x = x+1;

end

end

if x == 9

t = t+1;

end

x = 0;

end

if t == 0

if p < 100000

p = p+1;

Nodes(:,:,p) = tempr;

NodeInfo(:,:,p) = [p,q,0];

end

end

t = 0;

tempd = zeros(3);

[tempd] = movedown(Nodes(:,:,q),z);

%% movedown is a predefined function which will move the zeroth element one step in the down direction.

%% Following portion compares the newly generated node by the movedown function with all of the stored nodes, if the newly generated node is different than it will be saved as a new node.

t = 0;

for i = 1:p

c = (tempd == Nodes(:,:,i));

x = 0;

for j = 1:9

if c(j) == 1

x = x+1;

end

end

if x == 9

t = t+1;

end

x = 0;

end

if t == 0

if p < 100000

p = p+1;

Nodes(:,:,p) = tempd;

NodeInfo(:,:,p) = [p,q,0];

end

end

t = 0;

q = q +1;

% this line will increase the parent node number for the next iteration.

end

Pre-Defined functions

1. findzero()

% the function takes matrix as input and returns the index of the zeroth element.

function [z] = findzero(m)

for i = 1:9

if(m(i) == 0)

z = i;

end

end

1. moveleft()

function [templ] = moveleft(N,z)

% The function takes the matrix and the index of the zeroth element as

its inputs and returns the matrix in which the zeroth elements is moved

by one location in the left direction.

if(z~=1)&& (z~=2)&& (z~=3)

m = N;

T = m(z);

m(z) = m(z-3);

m(z-3) = T;

templ = m;

else

templ = N;

end

1. moveup()

function [tempu] = moveup(N,z)

% The function takes the matrix and the index of the zeroth element as

its inputs and returns the matrix in which the zeroth elements is moved by one location in the up direction.

if(z~=1)&& (z~=4)&& (z~=7)

m = N;

T = m(z);

m(z) = m(z-1);

m(z-1) = T;

tempu = m;

else

tempu = N;

end

1. movedown()

function [tempd] = movedown(N,z)

% The function takes the matrix and the index of the zeroth element as

its inputs and returns the matrix in which the zeroth elements is moved by one location in the down direction.

if(z~=3)&& (z~=6)&& (z~=9)

m = N;

T = m(z);

m(z) = m(z+1);

m(z+1) = T;

tempd = m;

else

tempd = N;

end

1. moveright()

function [tempr] = moveright(N,z)

% The function takes the matrix and the index of the zeroth element as its inputs and returns the matrix in which the zeroth elements is moved by one location in the right direction.

if(z~=7)&& (z~=8)&& (z~=9)

m = N;

T = m(z);

m(z) = m(z+3);

m(z+3) = T;

tempr = m;

else

tempr = N;

end