Mid Report
On
Assignment1,2

CSE-0408 Summer 2021

Rebecca Sultana
Dept. of CSE
State University of Bangladesh (SUB)
Dhaka, Bangladesh
rebecca.sultana.riya@gmail.com

***Abstract*—Main theme of your assignment or academic projects.**

Assignment 1. Heuristic functions for informed search

I. Introduction

Heuristic is a function which is used in Informed Search, and it finds the most promising path. It takes the current state of the agent as its input and produces the estimation of how close agent is from the goal. The heuristic method, however, might not always give the best solution, but it guaranteed to find a good solution in reasonable time. Heuristic function estimates how close a state is to the goal. It is represented by h(n), and it calculates the cost of an optimal path between the pair of states. The value of the heuristic function is always positive.

II. Admissibility

Heuristic function is given as: h(n) ¡= h*(n)
Here h(n) is heuristic cost, and h*(n) is the estimated cost. Hence heuristic cost should be less than or equal to the estimated cost.
In the informed search there is two main algorithms which are given below:
1. Best First Search Algorithm(Greedy search)
2. A* Search Algorithm

Assignment 2. Best-First search in Graph representation problem solving

III. Introduction

Best first search uses the concept of a priority queue and heuristic search. It is a search algorithm that works on a specific rule. The aim is to reach the goal from the initial state via the shortest path.

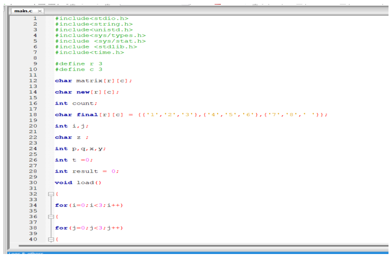
IV. Algorithm for BFS

Step 1: Choose the starting node and insert it into queue.
Step 2: Find the vertices that have direct edges with the vertex(node).
Step 3: Insert all the vertices found in step 3 into queue.
Step 4: Remove the first vertex(node) in queue.
Step 5: Continue this process until all the vertices are visited.

V. Conclusion

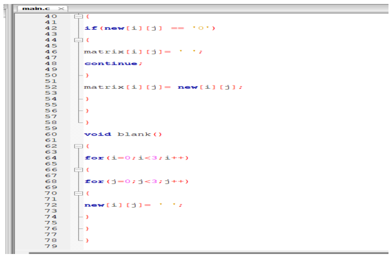
The BFS algorithm is useful for analyzing the nodes in a graph and constructing the shortest path of traversing through these.

VI. 8-puzzle problem Code



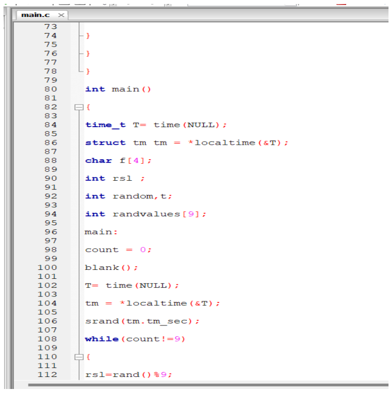
```
#include<stdio.h>
#include<string.h>
#include<unistd.h>
#include<sys/types.h>
#include <sys/stat.h>
#include <stdlib.h>
#include<time.h>

#define r 3
#define c 3

char matrix[r][c];

char new[r][c];

int count;

char final[r][c] = {{'1','2','3'},{'4','5','6'},{'7','8',' '}};

int i,j;

char z ;

int p,q,x,y;

int t =0;

int result = 0;

void load()

{

for(i=0;i<3;i++)

{

for(j=0;j<3;j++)

{
```

Fig. 1.



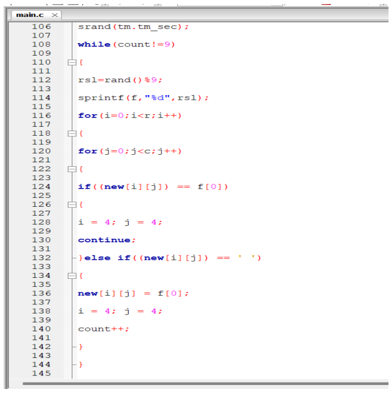
```
{

if(new[i][j] == '0')

{

matrix[i][j]= ' ';

continue;

}

matrix[i][j]= new[i][j];

}

}

}

void blank()

{

for(i=0;i<3;i++)

{

for(j=0;j<3;j++)

{

new[i][j]= ' ';

}

}

}
```

Fig. 2.



```c
}

}

}

int main()

{

time_t T= time(NULL);

struct tm tm = *localtime(&T);

char f[4];

int rsl ;

int random,t;

int randvalues[9];

main:

count = 0;

blank();

T= time(NULL);

tm = *localtime(&T);

srand(tm.tm_sec);

while(count!=9)

{

rsl=rand()%9;
```

Fig. 3.



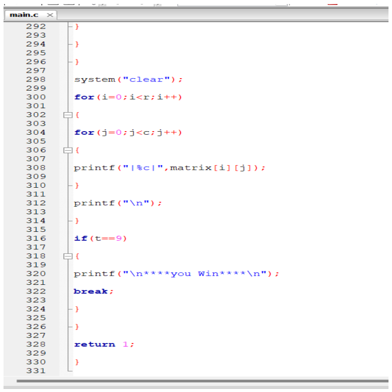
```c
srand(tm.tm_sec);

while(count!=9)

{

rsl=rand()%9;

sprintf(f,"%d",rsl);

for(i=0;i<r;i++)

{

for(j=0;j<c;j++)

{

if((new[i][j]) == f[0])

{

i = 4; j = 4;

continue;

}else if((new[i][j]) == ' ')

{

new[i][j] = f[0];

i = 4; j = 4;

count++;

}

}
```

Fig. 4.

VII. Output

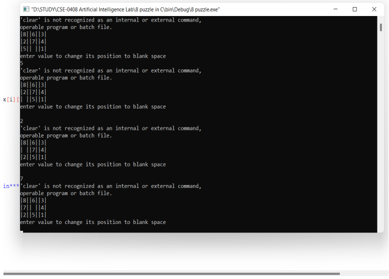
VIII. BFS Algorithm Code

IX. Output



```c
}

}

}

system("clear");

for(i=0;i<r;i++)

{

for(j=0;j<c;j++)

{

printf("|%c|",matrix[i][j]);

}

printf("\n");

}

if(t==9)

{

printf("\n****you Win****\n");

break;

}

}

return 1;

}
```

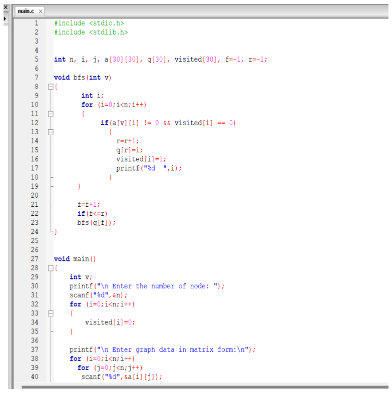
Fig. 5.



"D:\STUDY\CSE-0408 Artificial Intelligence Lab\8 puzzle in C\bin\Debug\8 puzzle.exe"

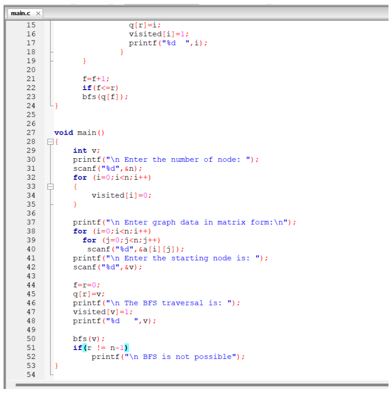
```
'clear' is not recognized as an internal or external command,
operable program or batch file.
|8||6||3|
|2||7||4|
|5|| ||1|
enter value to change its position to blank space
5
'clear' is not recognized as an internal or external command,
operable program or batch file.
|8||6||3|
|2||7||4|
| ||5||1|
enter value to change its position to blank space

2
'clear' is not recognized as an internal or external command,
operable program or batch file.
|8||6||3|
| ||7||4|
|2||5||1|
enter value to change its position to blank space

7
'clear' is not recognized as an internal or external command,
operable program or batch file.
|8||6||3|
|7|| ||4|
|2||5||1|
enter value to change its position to blank space
```

Fig. 6.



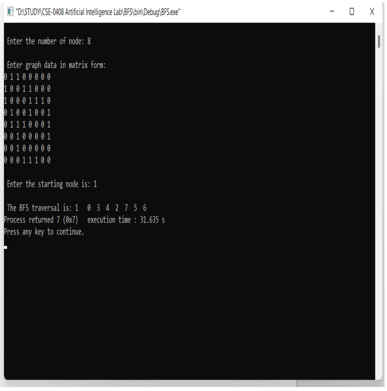
```c
#include <stdio.h>
#include <stdlib.h>


int n, i, j, a[30][30], q[30], visited[30], f=-1, r=-1;

void bfs(int v)
{
       int i;
       for (i=0;i<n;i++)
       {
            if(a[v][i] != 0 && visited[i] == 0)
              {
                r=r+1;
                q[r]=i;
                visited[i]=1;
                printf("%d  ",i);
              }
      }

      f=f+1;
      if(f<=r)
      bfs(q[f]);
}


void main()
{
    int v;
    printf("\n Enter the number of node: ");
    scanf("%d",&n);
    for (i=0;i<n;i++)
    {
        visited[i]=0;
    }

    printf("\n Enter graph data in matrix form:\n");
    for (i=0;i<n;i++)
      for (j=0;j<n;j++)
       scanf("%d",&a[i][j]);
```

Fig. 7. BFS code



```c
                q[r]=i;
                visited[i]=1;
                printf("%d  ",i);
              }
      }

      f=f+1;
      if(f<=r)
      bfs(q[f]);
}


void main()
{
    int v;
    printf("\n Enter the number of node: ");
    scanf("%d",&n);
    for (i=0;i<n;i++)
    {
        visited[i]=0;
    }

    printf("\n Enter graph data in matrix form:\n");
    for (i=0;i<n;i++)
      for (j=0;j<n;j++)
       scanf("%d",&a[i][j]);
    printf("\n Enter the starting node is: ");
    scanf("%d",&v);

    f=r=0;
    q[r]=v;
    printf("\n The BFS traversal is: ");
    visited[v]=1;
    printf("%d   ",v);

    bfs(v);
    if(r != n-1)
        printf("\n BFS is not possible");
}
```

Fig. 8. BFS code



"D:\STUDY\CSE-0408 Artificial Intelligence Lab\BFS\bin\Debug\BFS.exe"

```

 Enter the number of node: 8

 Enter graph data in matrix form:
0 1 1 0 0 0 0 0
1 0 0 1 1 0 0 0
1 0 0 0 1 1 1 0
0 1 0 0 1 0 0 1
0 1 1 1 0 0 0 1
0 0 1 0 0 0 0 1
0 0 1 0 0 0 0 0
0 0 0 1 1 1 0 0

 Enter the starting node is: 1

 The BFS traversal is: 1   0  3  4  2  7  5  6
Process returned 7 (0x7)   execution time : 31.635 s
Press any key to continue.
```

Fig. 9. BFS output