

```
1  /*PathFinder:
2
3      ~Nitin Rohit
4
5  This is a simple code which illustrates how the algorithm to find the path to exit
6  works. I have added a simulation using ASCII
7  characters which shows how the route is found. The path used may not be the most
8  efficient one but it finds a way if its there.*/
9
10 //Include the necessary header files
11 #include<iostream>
12 #include<unistd.h>
13
14 using namespace std;
15
16 //Size of the grid which the maze will be created
17 //The user can change the values according to the required map size
18 const int m=6,n=6;
19
20 int found=1;
21
22 //function to create a copy of the given array
23 void clone(int (*map)[n],int (*mapc)[n])
24 {
25     for(int i=0;i<m;i++)
26     for(int j=0;j<n;j++)
27     mapc[i][j]=map[i][j];
28 }
29
30 //function to display the map each time to create a simulation
31 void display(int (*mapc)[n],int x,int y)
32 {
33     system("cls");
34     for(int i=0;i<m;i++)
35     {
36         cout<<"|";
37         for(int j=0;j<n;j++)
38         {
39             if(i==x && j==y)
40                 cout<<":|";
41
42             else if(mapc[i][j])
43                 cout<<" |";
44             else
45                 cout<<"X|";
46         }
47         cout<<endl;
48     }
49     usleep(1000000);
50 }
51
52 //Recerssive function to find the path using a simple rule to go in all direction
53 until the exit is found
54 int route(int i,int j,int (*map)[n],int (*mapc)[n])
55 {
56     if(found)
57         display(mapc,i,j);
58
59     if(i==(m-1) && j==(n-1))
60     {
```

```
56         cout<<"YOU HAVE REACHED THE END!!";
57         found=0;
58         return 0;
59     }
60
61     map[i][j]=0;
62
63     if(map[i-1][j] && i>0)
64         route(i-1,j,map,mapc);
65
66     if(map[i][j+1] && j<n)
67         route(i,j+1,map,mapc);
68
69     if(map[i+1][j] && i<m)
70         route(i+1,j,map,mapc);
71
72     if(map[i][j-1] && j>0)
73         route(i,j-1,map,mapc);
74
75     return 0;
76
77 }
78
79 //function to get the map values from the user
80 void getValue(int (*map)[n])
81 {
82     cout<<"Enter the values:"<<endl;
83     for(int i=0;i<m;i++)
84         for(int j=0;j<n;j++)
85             cin>>map[i][j];
86 }
87
88 //Then just execute the made functions in the main function
89 int main()
90 {
91     int map[m][n],mapc[m][n];
92     getValue(map);
93     clone(map,mapc);
94     route(0,0,map,mapc);
95
96     return 0;
97 }
```