Homework 1

Part1:

1. In figure8, agent in cell A only knows the cells next to it, which means there is no blocked cells around the agent. We calculate the herustic function (f = g + h) for each cell around A. Because we use Manhattan distances as h-value, which means the cell in the east of A has f = 1 + 2 = 3, and the cell in the north of A has f = 1 + 4 = 5. So the cell in the east has smaller f than the cell in the north. The agent will choose to move to east instead of north.

Part4:

1. Assume there is a node m which is expanded by n, and in this problem, we can only move in four main compass directions, which means



And also, the agent will actually take action(n,m) which is definitely larger than the Manhattan distance between m and n, because the agent may encounter some obstacles.



 and 

So, 

So, the Manhattan distance are consistent in this case.

1.  and 

According to the fomular provided in the problem,

 and 

So, 

We assume that there exits  is larger than 



So, 

We can see that even though we increase the action cost between m and n,  is still the upper bound of the h value of n. So we can use  to find the path as well.

1. If a fucntion is consistent, which means 

Replace  with  ,we can see that



 , which means the function is admissible.

Part 6:

We set the maze size as 1001\*1001. We can try several ways to reduce the memory consumption. We can use one bit to store whether the cell is blocked or not, that is 1001\*1001 = 1002001bits = 125250byte. And only 30% cells will be marked as blocked, we don’t need to store g value and h value of each cell, that is 1001\*1001\*0.7\*1\*2 = 1402800byte. And visited and unvisited table need 1001\*1001 = 1002001bits = 125250byte. Open and close table are at most 1001\*1001\*0.7\*1\*2 = 1402800byte. Tree pointer 1001\*1001\*2 = 2004002bits = 250500byte.

So, the total memory is 3.15MB.