Armin Bazarjani
Csci360 Project 2

Task 4:

In the following questions the MAPF instances are shown in boxes with agents A_0 and A_1 labeled in blue and green respectively. The goal nodes are labeled as G_A and G_B. The gray boxes are locations that the agents cannot go into while the white boxes are valid locations.

Question 1:

A_0			
		G_1	**G_0**
A_1			

Finds an optimal solution for the priority ordering [0,1]
Fails to find a collision free solution for the ordering [1,0]

This fails the find an optimal solution for the second priority ordering because agent 1's goal node is in front of agent 0's goal node. Prioritized planning tells agent 1 to go to its goal node first and the lower priority agent, agent 0, has to plan its path around agent 1's path. The problem is that agent 1 blocks agent 0 from ever reaching its goal node, so agent 0 fails to find a path.

Question 2:

G_1	**A_0**	**A_1**	**G_0**

This fails to find a collision free path for both agents, regardless of ordering. This is because both of the agents starting locations block the other agent from its goal node and there are no spaces for the lower priority agent to move into except into the goal node, into the higher priority agent, or it can wait where it currently is. All three of which will result in a collision with the higher priority agent.

Question 3:

The answer to this is the same as in Question 1. The path where agent 0 has the highest priority is optimal because it allows agent 0 to move first and reach its goal node. If agent 1 had the highest priority it would block agent 0 from ever reaching its goal node. The path is optimal because one of the agents will have to wait for the other to move first in order for there to be no collision.

Question 4:

				G_1				
A_0								**A_1**
						G_0		

If the priority ordering is [0,1] the sum of costs is: 17
Whereas, if the priority ordering is [1,0] the sum of costs is: 13

This is because if agent 1 has the highest priority agent 0 only has to wait one time step before it can move forward and enter its goal state. However, if agent 0 has the highest priority agent 1 moves into the node just before the node below its goal state, however it can’t move into the node beneath its goal state because agent 0 will be going into that node on the next timestep, so agent 1 has to move backwards then go up into the node that is two nodes above agent 0’s goal state, spend an action waiting there, and then it can finally move onto the path to go into its goal state.