Project #2 - Bizarre Cave Problem

Mauricio Macias

Gabriel Rivera

Mitchell Norseth

Complexity Order of our Algorithm

For this project we as explorers want to find a path from the start of the 'cave' which is at coordinate (16,0,0) to reach the point (8,1,7). There are limits to the moves we make to reach the other coordinates, or rooms, as the limits of each variable are x = 16, y = 9, and z = 7. Another rule is that you are only able to change two variables per move and you either have to deplete one of the variables, or completely fill one to the limit. The final rule is that the sum of the coordinates never change (x+y+z = 16).

We have created an algorithm that randomly inputs coordinates until it *legally* reaches the point (8,1,7). If a point is selected that is not legal, the program will simply enter more points until a valid choice is selected. Each run of the program is different as the algorithm can sometimes be trapped in a loop for a while as it explores the rooms of the cave, which will of course slow down the process to reaching the end goal. This algorithm can be considered a Las Vegas type algorithm as it will always produce a correct result, but the time complexity is based on random values. We can say that the expected worst case time complexity of our algorithm is O(N^2) because we don't know the amount of inputs that the program will attempt before finding the target coordinate. The more inputs the algorithm attempts, the longer the program will be running. This is why we analyze the worst case scenario.