Name: Keaton Whitehead
ID: 104668391

CSCI 3104, Algorithms **Profs. Grochow & Layer**
Explain-It-Back 11 **Spring 2019, CU-Boulder**

A startup has hired you as the chief technology officer (i.e., the only one who knows how to program). After the founders (all MBAs) finish explaining their vision for changing the world, you realize that what they describe can be reduced to the traveling salesman problem. No worries, you develop a solution that is a 1.5 approximation. The founders are devastated that they cannot use the word "optimal" in their next VC pitch, and wonder out loud if they need to get a new CTO who can do better. Convince them that an efficient optimal solution is unlikely (i.e., P probably does not equal NP) and that your solution is quite good.

Ladies and Gentlemen,

I do appreciate your concern to finding the "optimal" solution to you great idea, however, I do not appreciate the disregard of my solution. May I remind you that this is my field and I know my solution has a 1.5 approximation it is the best that can be done.

To understand this you need to comprehend the problem in which you guys are looking at. This problem falls into a complexity class problems that are equivalent to other unsolved computer science problems. These group of problems ask whether every problem whose solution can be quickly verified in polynomial time or not, and then can also be solved in polynomial time or not. We have 3 groupings of problems: P - verified and solved in polynomial time, NP - fast to verify, but not solved in polynomial, NP Complete - an NP problem that can be reduced in polynomial time and whose solution may be verified in polynomial time.

Because I have been able to determine that your problem can reduce to the traveling salesman problem, that means that initially the problem can not be verified in polynomial time but because we can reduce it it can be reduced in polynomial time and the solution can be verified in polynomial time. To put this in simpler terms I basically have taken your solution and reduced it so that a result will be able to be checked on correctness in a reasonable amount of time. If the verification was not polynomial we would have a solution, but no way of knowing if it is accurate or not because our verification algorithm would take so long that it has potential to take multiple weeks, or months, and in some cases years. By reducing it to polynomial time for verification we can get results with in a day to a week.

Now I know this is not "optimal" but any other Computer Scientist will tell you the same thing and not be able to get it to be any more "optimal" than what I have.

Name: Keaton Whitehead
ID: 104668391

CSCI 3104, Algorithms **Profs. Grochow & Layer**
Explain-It-Back 11 **Spring 2019, CU-Boulder**

I know you don't understand the massivity to the problem you tasked me to solve, so I do not take you insult with any weight because you simply just misunderstand the problem at hand.

I hope this has shed some light on the situation and that you do not fire me :)

Best,

Keaton Whitehead

Name: Keaton Whitehead
ID: 104668391

CSCI 3104, Algorithms **Profs. Grochow & Layer**
Explain-It-Back 11 **Spring 2019, CU-Boulder**