Andy Arellano
Greedy Graph Write up

1A) The goal of this problem is to schedule students to complete all the steps with the minimum amount of switching possible. This can be achieved by using students who are able to do the most consecutive steps

1B) Start by finding a student that can do the first step and then the max consecutive steps that the student can do. Then keep repeating the process over until all the steps have been scheduled to a student. If there are multiple students that can do the same max consecutive steps take the first student that was found.

1D) Runtime for the algorithm is O(m*n). The worst-case scenario would be if all the students could do every step, then we would have to check every step(n) for each student(m).

1E) Let us say there is ALG which represents my algorithm and OPT which represents an optimal algorithm. Now lets assume that ALG gives a solution of a1,a2,…ak and x switches and OPT gives us a solution of o1,o2….ok and y switches and y<x. Lets say that at the until the ith index both algorithms are the same. If we use cut and paste we get a1,a2…ai, oi+1,oi+2…ok. Both algorithms are designed to take the students that do the most consecutive steps so replacing ai with oi will result in the same number of switches which is a contradiction of y<x which means that ALG is optimal.

2A) The algorithm used is a modified version of Dijkstra's algorithm that was altered to take into account of the wait time. I've read about the Dijkstra's algorithm from https://www.geeksforgeeks.org/dijkstras-shortest-path-algorithm-greedy-algo-7/

2B) In a worst case scenario the time complexity would be O(V^2logV+V^2)

2C) The shortest Time method uses Dijkstra's Algorithm

2D) The shortest Time algorithm would have to be altered to take into account additional wait time if the train is not present

2E) The Complexity of shortest Time is O(V^2*logV). The complexity can be reduced to O((E)+VlogV) by using a binary heap.