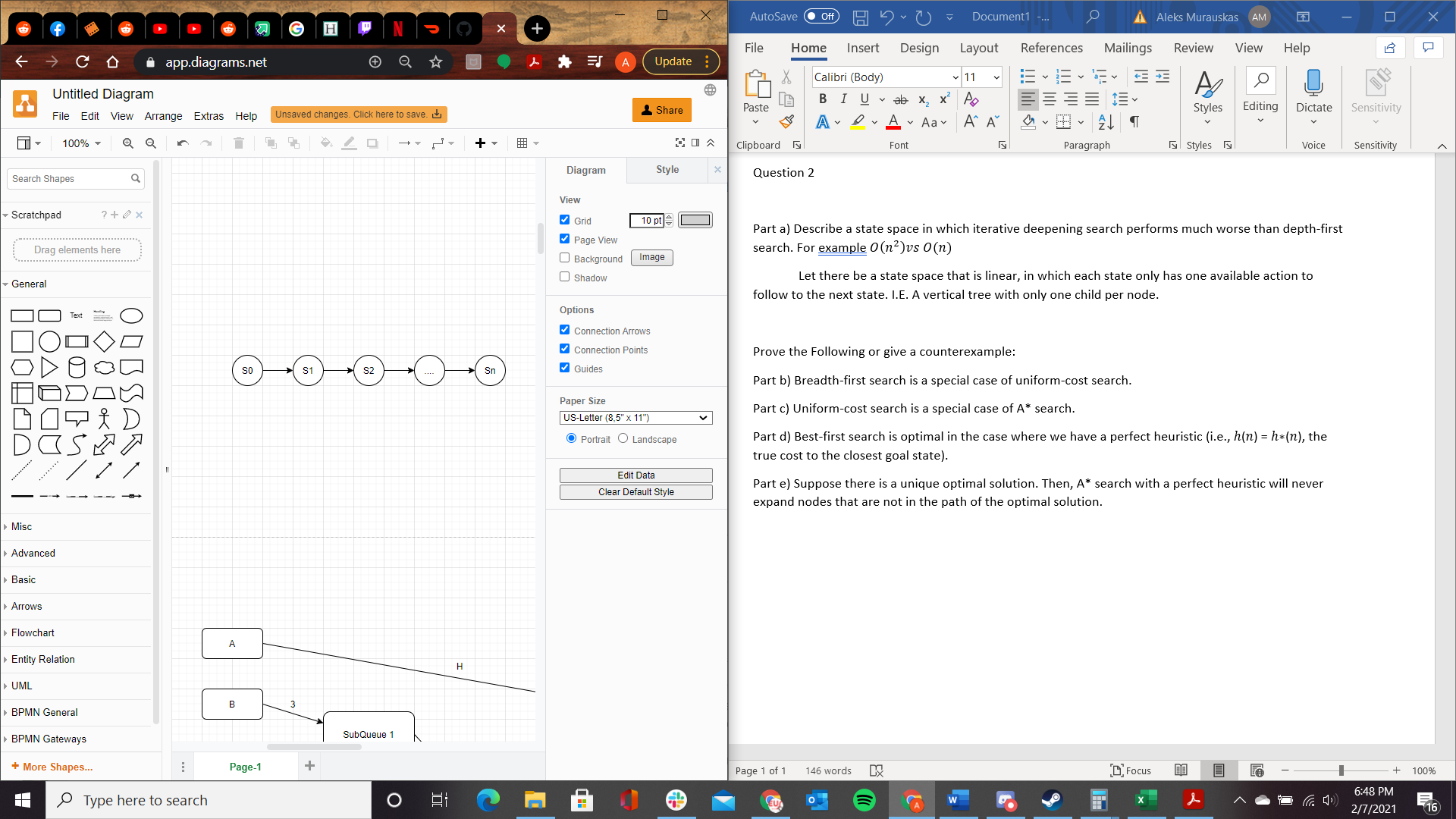
Aleksas Murauskas

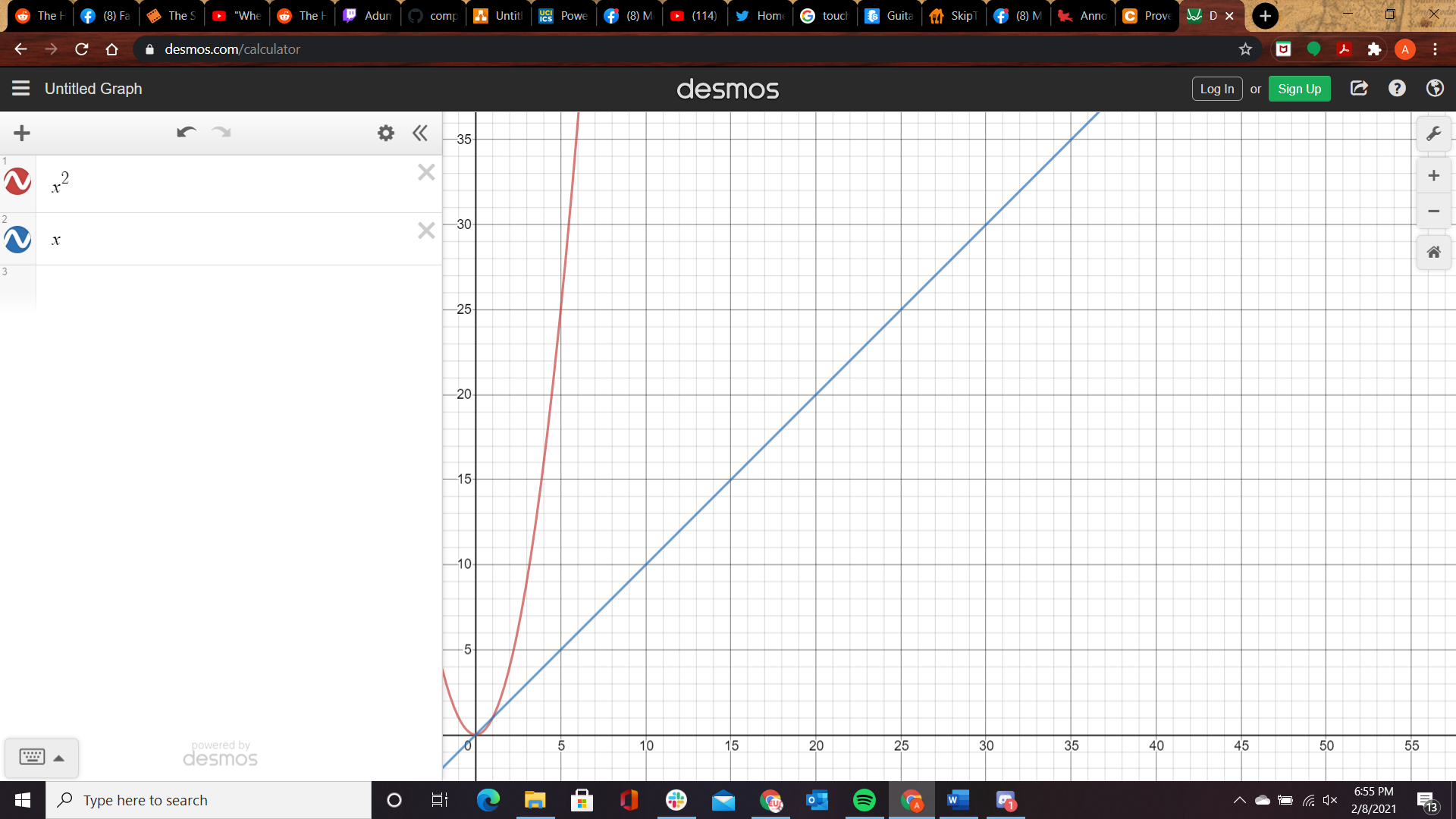
260718389

Question 2

Part a) Describe a state space in which iterative deepening search performs much worse than depth-first search. For example

Let there be a state space that is linear, in which each state only has one available action to follow to the next state. I.E. A vertical tree with only one child per node. When Depth first search acts upon the state tree, it has a complexity of O(n) as it acts upon each node only once. Iterative deepening will have a Time complexity of , as a new iteration will begin at every depth and these iterations will grow in size exponentially.





Prove the Following or give a counterexample:

Part b) Breadth-first search is a special case of uniform-cost search.

Let UCS: f(n) =g(n) where g(n) = 1\*(depth of node n)

Run BFS on a tree where all weights of state transition are 1. Since this is a tree with no loops, the cost of a path found by BFS will equal the Depth of the node. Therefore BFS is equivalent to UCS in the special case where there are unit cost transitions.

Part c) Uniform-cost search is a special case of A\* search.

Let There be a A\* search where the heuristic function will always return a unit value of 0. Let h(n)=0 for all n. Since A\*: f(n) =g(n)+h(n) and UCS: f(n)=g(n) and h(n) = 0 then A\*: f(n) =g(n)+0 which is equivalent to UCS f(n) = g(n). Therefore, UCS will operate equivalently to A\* and therefore is a special case of A\*.

Part d) Best-first search is optimal in the case where we have a perfect heuristic (i.e., ℎ(𝑛) = ℎ∗(𝑛), the true cost to the closest goal state).

True: Let Best first: f(n) = g(n) +h\*(n). We know that A\* Search is optimal given a perfect heuristic. A\* search operates by running best first search with a heuristic. Therefore, Best first search is equivalent to A\* search as long as H(n) is admissible. As we know that A\* Search is Optimal given a perfect heuristic, O(bd) as only nodes along the optimal path are expanded. Therefore, a Best First search with a perfect heuristic is also optimal.

Part e) Suppose there is a unique optimal solution. Then, A\* search with a perfect heuristic will never expand nodes that are not in the path of the optimal solution.

True: A\* operates by implementing a greedy Best First algorithm upon an admissible heuristic. At each node, it will calculate cost with f(n) = g(n) +h\*(n). h(n) is confirmed perfect by the question. G(n) is the cost so far, and since the cost so far was greedily chosen with the use of a perfect heuristic, therefore if h(n) is optimal then g(n) must be optimal as well. We know have f(n) =g\*(n) + h\*(n). The sum of two optimal functions will also be optimal. By dominance H2(n) >= h1(n) then h2 dominates h1, since we have a perfect heuristic it will always dominate and deliver the optimal path. In order for a node off the optimal path to be expanded it would have to have a higher heuristic to be chosen by the best first algorithm. For this node to exist it would have to be multiple optimal paths, which contradicts the given statement that the path is unique, or the heuristic is not perfect which contradicts the given heuristic. As both these cases contradict the given statement, A\* search will never expand nodes that are not in the path of the optimal solution.