A#_____ Name_____ Date 5/1/2023

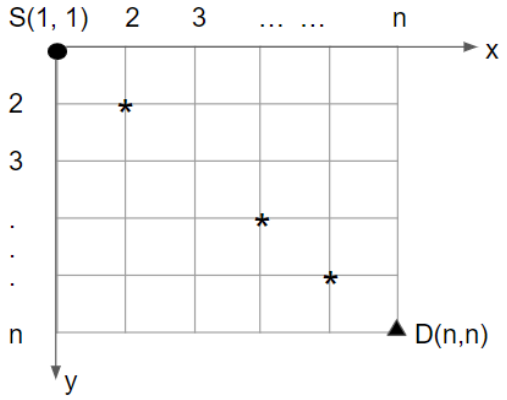
(100pts)

1. (20 pts) The knapsack problem is described as a combinatorial optimization. Given a set of n items with a weight and a value, we are going to find a collection to maximize the total value of the times we choose under or equal to the weight limit. Argue that the knapsack problem is an NP problem.

2. (20pts) An elevator runs between floor 1 and floor n. At floor *i*, there is an integer k_i. If you press button UP at floor i, the elevator will go up by k_i levels; if you press button DOWN, the elevator will go down by k_i levels. But the elevator can not go higher than n or lower than 1. If a person is at floor *a* and wants to go to floor *b*, how many times does he have to press the button at least? Design an algorithm to return the time or -1 when he can not reach *b*. (pseudo code is required)

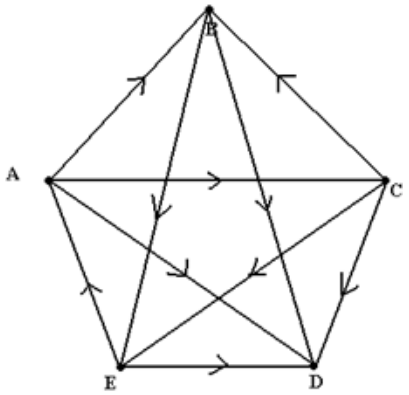
3. (20pts) A car sets off from S and drives to D as shown in the figure. Gas stations have been set at some intersections (asterisks). The car can only run along sides and a full tank runs K sides. The driver has to pay the following fees:
   - To refill a full tank of gas, pay $A.
   - If the car goes back (either x coordinate or y coordinate decreases when driving one more side), pay $B.
   - The driver can set up some extra gas stations, but has to pay $C for each setup. Neither S or D allows a gas station setup.

   We assume that the car sets off with a full tank of gas for free. Design an algorithm to return a path from S to D that spends minimal money. (pseudo code required)



4. (20pts) A digraph G is called a dominance-directed graph if for any pair of distinct vertices u and v of G, either u→v or v→u, but not both (here the notation u→v means there is an edge from u to v). Below is an example of a dominance-directed graph. A, B, C, D and E are five sports teams. Teams play each other exactly once, with no ties allowed. A -> B means team A beats team B.

In a dominance-directed graph, we define the ***power*** of a vertex as being the total number of 1-step and 2-step connections to other vertices. For example, the power of A is 4. Using the adjacency matrix and its square, calculate the power of each vertex and rank each team according to their vertex power.(show your work)



Graph H

5. (20pts) There are two types of professional wrestlers: "good guys" and "bad guys." Between any pair of professional wrestlers, there may or may not be a rivalry. Suppose we have n professional wrestlers and we have a list of r pairs of wrestlers for which there are rivalries. Give an O(n + r)-time algorithm that determines whether it is possible to designate some of the wrestlers as good guys and the remainder as bad guys such that each rivalry is between a good guy and a bad guy. If it is possible to perform such a designation, your algorithm should produce it. (pseudo code required)