For which of the following will you always find the same solution, even if you re-run the algorithm multiple times? * 1/1

Assume a problem where the goal is to minimize a cost function, and every state in the state space has a different cost.

- ( ) Steepest-ascent hill-climbing, each time starting from a different starting state
- (•) Steepest-ascent hill-climbing, each time starting from the same starting state ✓
- ( ) Stochastic hill-climbing, each time starting from a different starting state
- ( ) Stochastic hill-climbing, each time starting from the same starting state
- ( ) Both steepest-ascent and stochastic hill climbing, so long as you always start from the same starting state
- ( ) Both steepest-ascent and stochastic hill climbing, each time starting from a different starting state
- ( ) No version of hill-climbing will guarantee the same solution every time

The following two questions will both ask you about the optimization problem described below.

A farmer is trying to plant two crops, Crop 1 and Crop 2, and wants to maximize his profits. The farmer will make $500 in profit from each acre of Crop 1 planted, and will make $400 in profit from each acre of Crop 2 planted.

However, the farmer needs to do all of his planting today, during the 12 hours between 7am and 7pm. Planting an acre of Crop 1 takes 3 hours, and planting an acre of Crop 2 takes 2 hours.

The farmer is also limited in terms of supplies: he has enough supplies to plant 10 acres of Crop 1 and enough supplies to plant 4 acres of Crop 2.

Assume the variable C1 represents the number of acres of Crop 1 to plant, and the variable C2 represents the number of acres of Crop 2 to plant.

✕ What would be a valid objective function for this problem? * 0/1

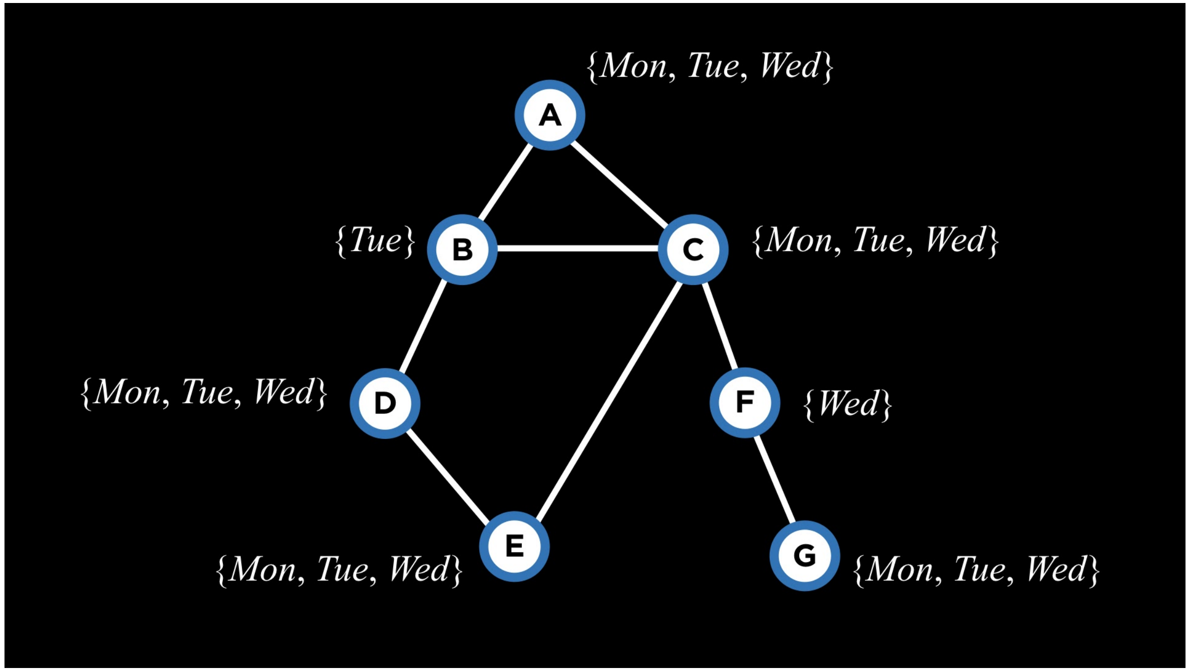
- (●) -3 * C1 - 2 * C2 ✕
- ( ) 500 * C1 + 400 * C2
- ( ) C1 + C2
- ( ) 10 * C1 + 4 * C2
- ( ) 500 * 10 * C1 + 400 * 4 * C2

What are the constraints for this problem? *

1/1

- [x] 3 * C1 + 2 * C2 <= 12, C1 <= 10, C2 <= 4
- [ ] 3 * C1 + 2 * C2 <= 12, C1 + C2 <= 14
- [ ] 3 * C1 <= 10, 2 * C2 <= 4
- [ ] C1 + C2 <= 12, C1 + C2 <= 14

The following question will ask you about the below exam scheduling constraint satisfaction graph, where each node represents a course. Each course is associated with an initial domain of possible exam days (most courses could be on Monday, Tuesday, or Wednesday; a few are already restricted to just a single day). An edge between two nodes means that those two classes must have exams on different days.



After enforcing arc consistency on this entire problem, what are the resulting domains for the variables C, D, and E? * 0/1

- C's domain is {Mon}, D's domain is {Mon, Wed}, E's domain is {Tue, Wed}
- C's domain is {Mon}, D's domain is {Tue}, E's domain is {Wed}
- C's domain is {Mon, Tue}, D's domain is {Wed}, E's domain is {Mon}
- C's domain is {Mon}, D's domain is {Wed}, E's domain is {Tue}
- C's domain is {Mon, Tue, Wed}, D's domain is {Mon, Wed}, E's domain is {Mon, Tue, Wed}
- C's domain is {Mon}, D's domain is {Mon, Wed}, E's domain is {Mon, Tue, Wed}

Comments, if any