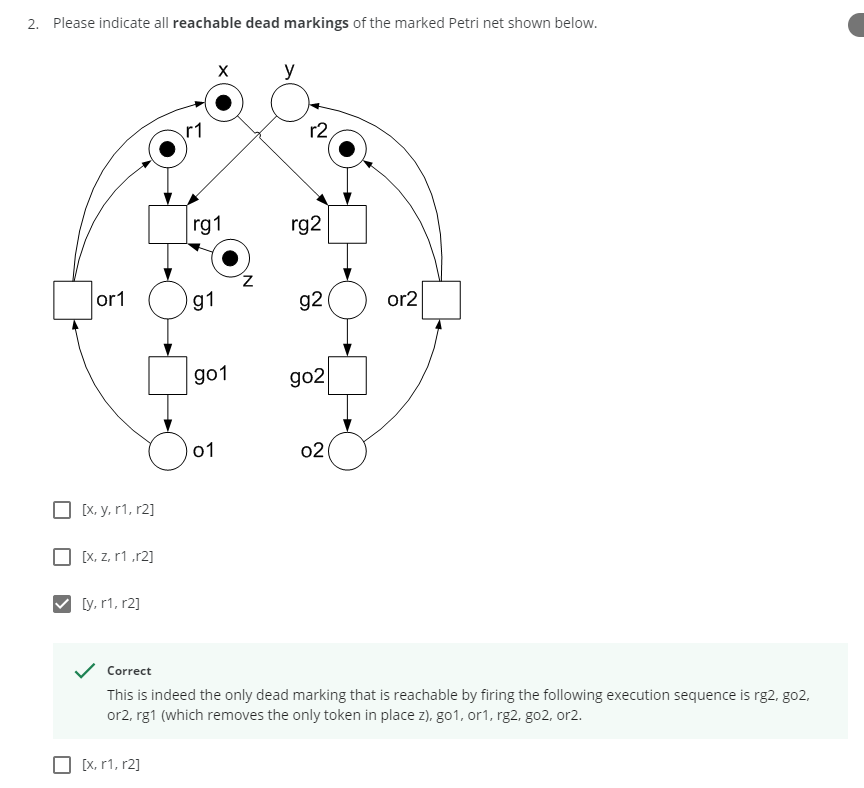


Explanation: If we take case id as doctor then doctors can handle their patients cases accordingly



Explanation:

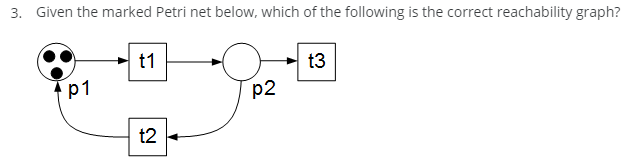
Firing order: rg2, go2, or2 then we will end up having tokens at r2 and y (Initially x and r2 tokens are used for this process)

With the help of tokens at r1, z and y we have firing order: rg1, go1, or1 then we will end up having tokens at r1 and x

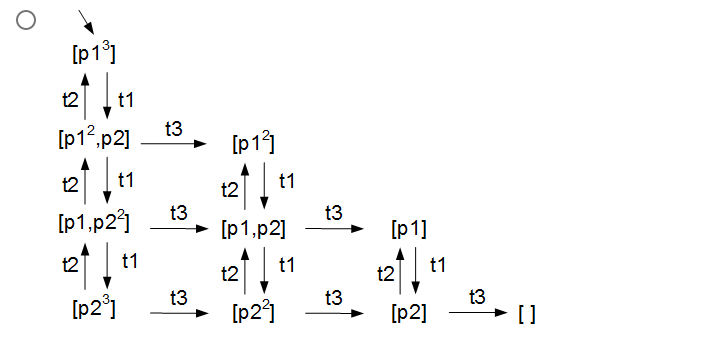
Again using tokens at x and r2, we can fire rg2, go2, or2. So we end having tokens at y and r2

At this time we have tokens at r1, r2 and y. We can’t fire rg1 as we are lacking a token at z and similarly rg2 as we don’t have a token at y.

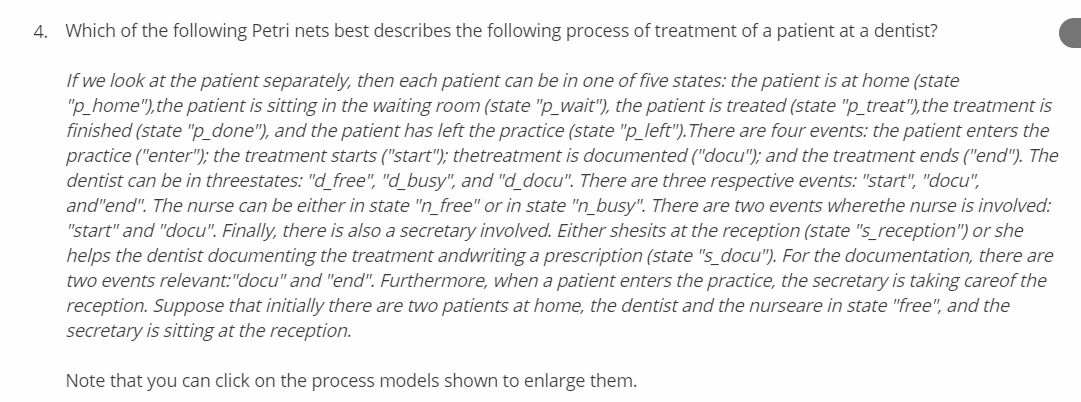
So r1, r2, y are the dead markings



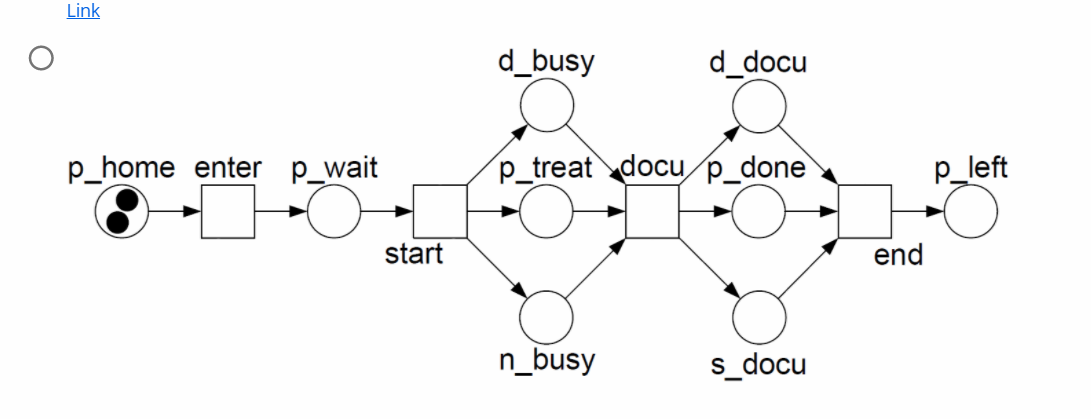
Answer:



Explanation: If we fire t3 token will be gone after the transition

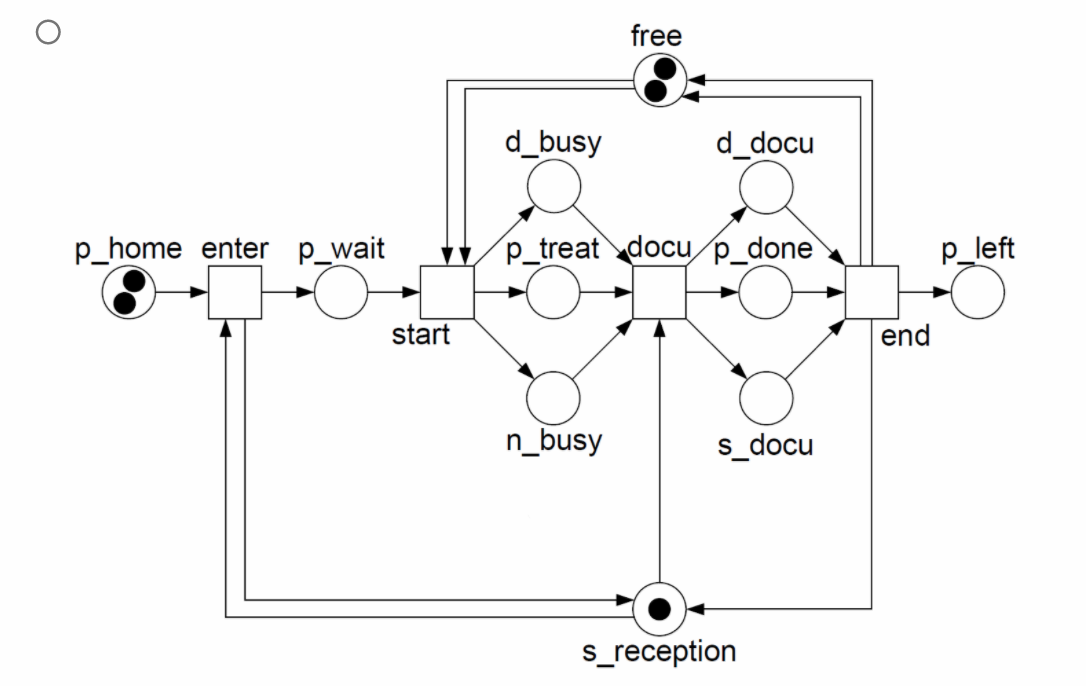


This is not the correct answer



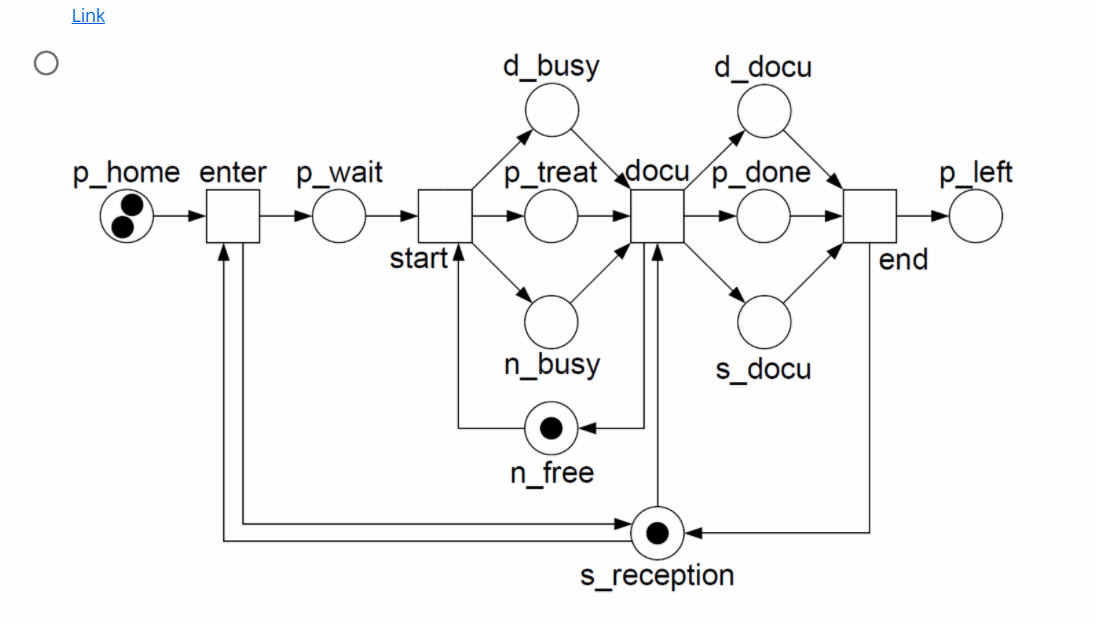
Reason: Secretary is available at reception, so patient can contact secretary if he want which is not shown in the diagram

This is not the correct answer



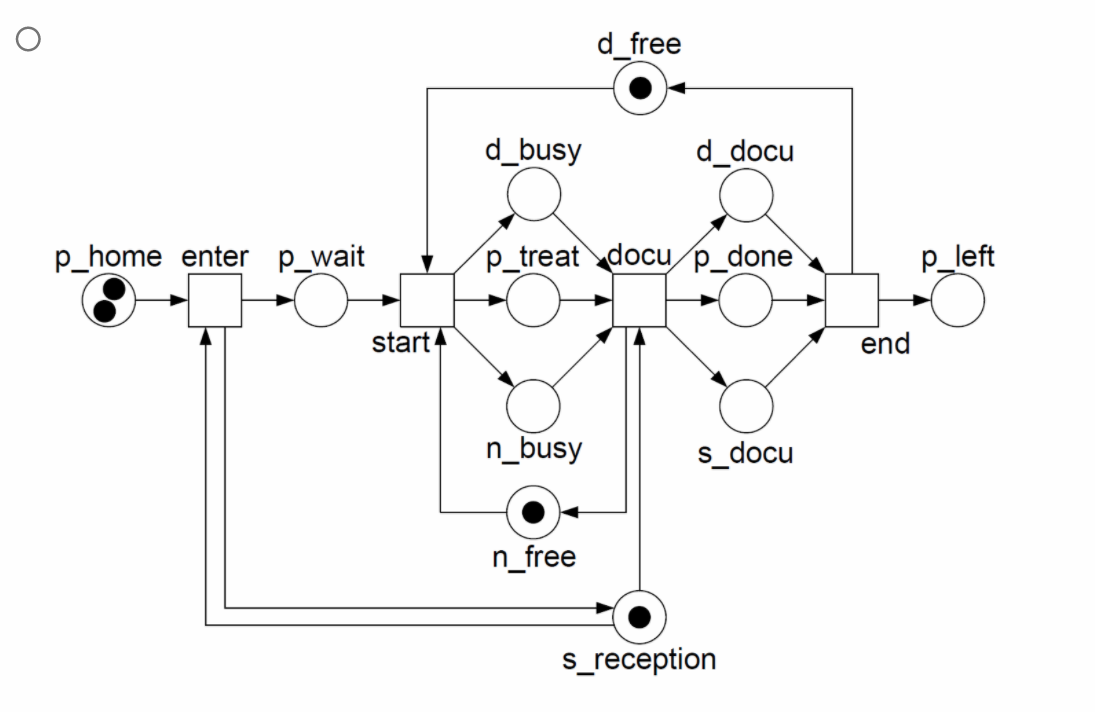
Reason: There is no ‘free’ notation given in the question as we have d\_free and n\_free already

This is not the correct answer



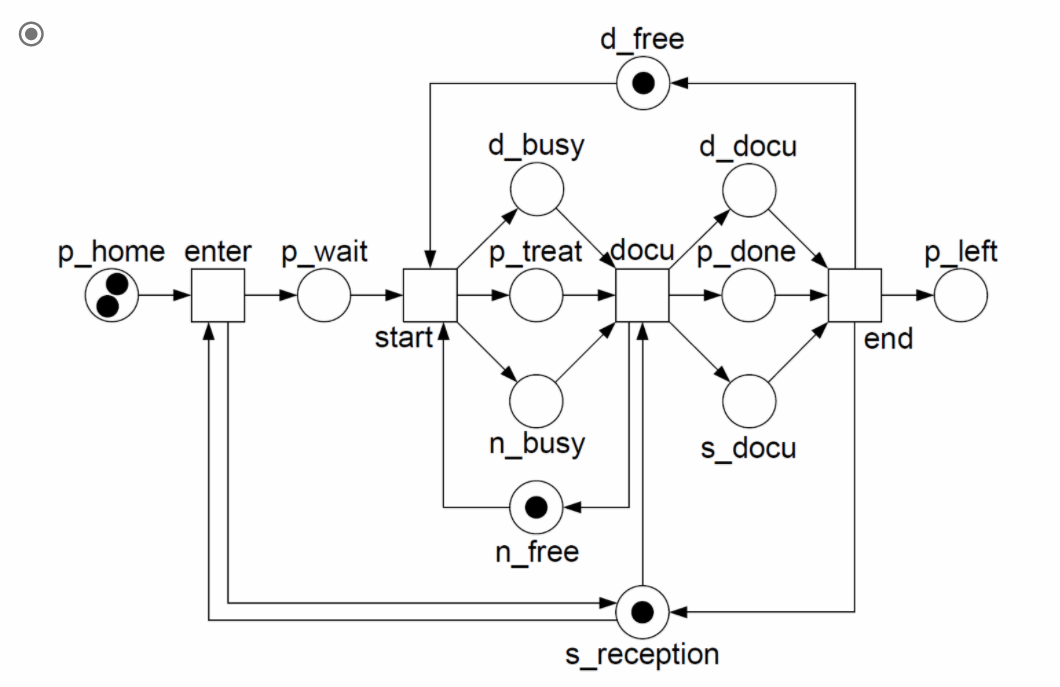
Reason: There is no d\_free notation as dentist should be free at initial state

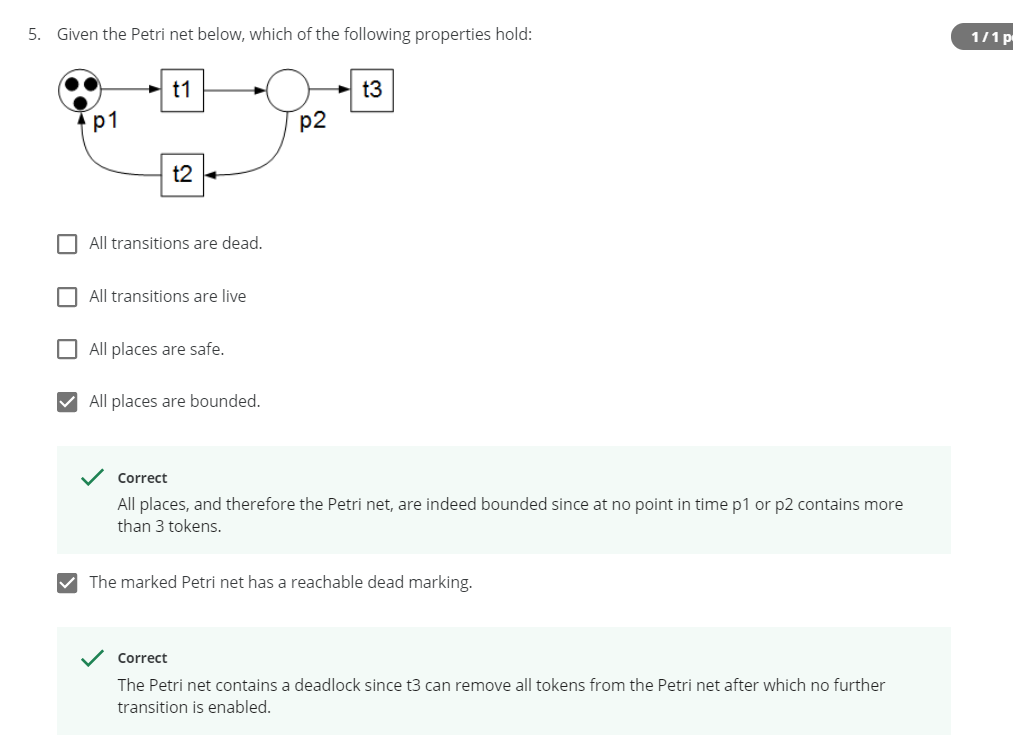
This is not correct answer



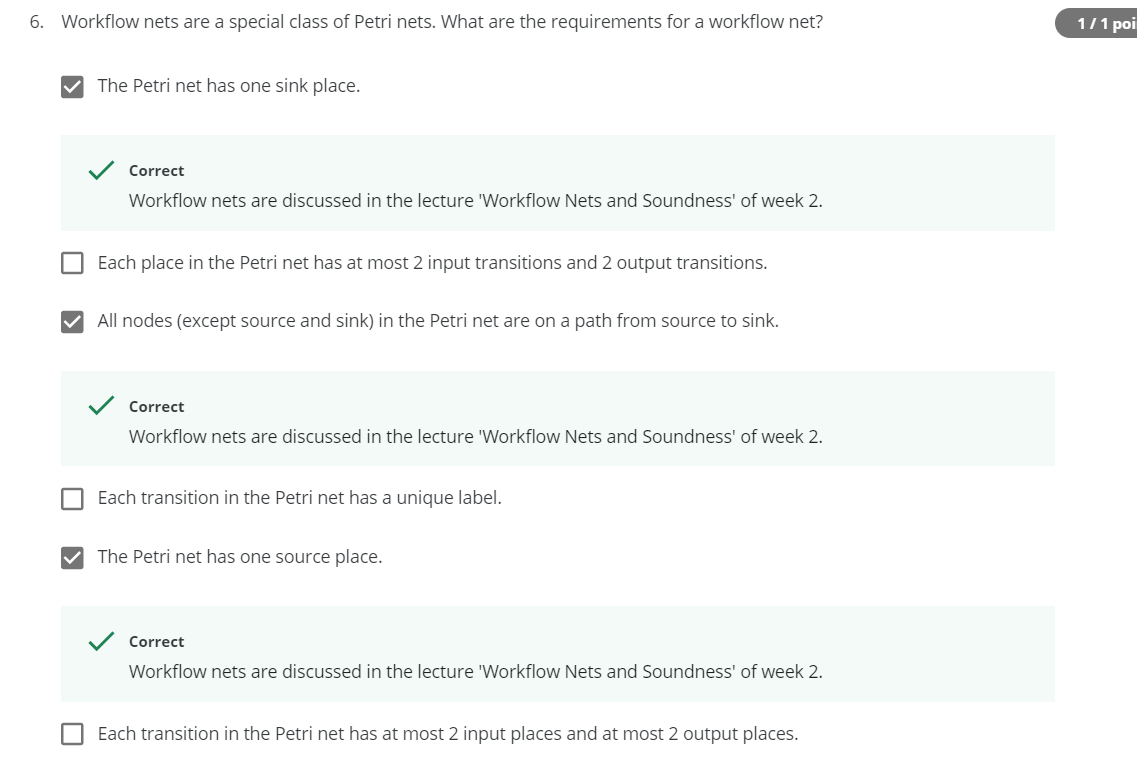
Reason: After end of documentation ‘end’, the patient or secretary can go the reception (arrow is missing from end to s\_reception)

This is the correct answer: It satisfies all conditions

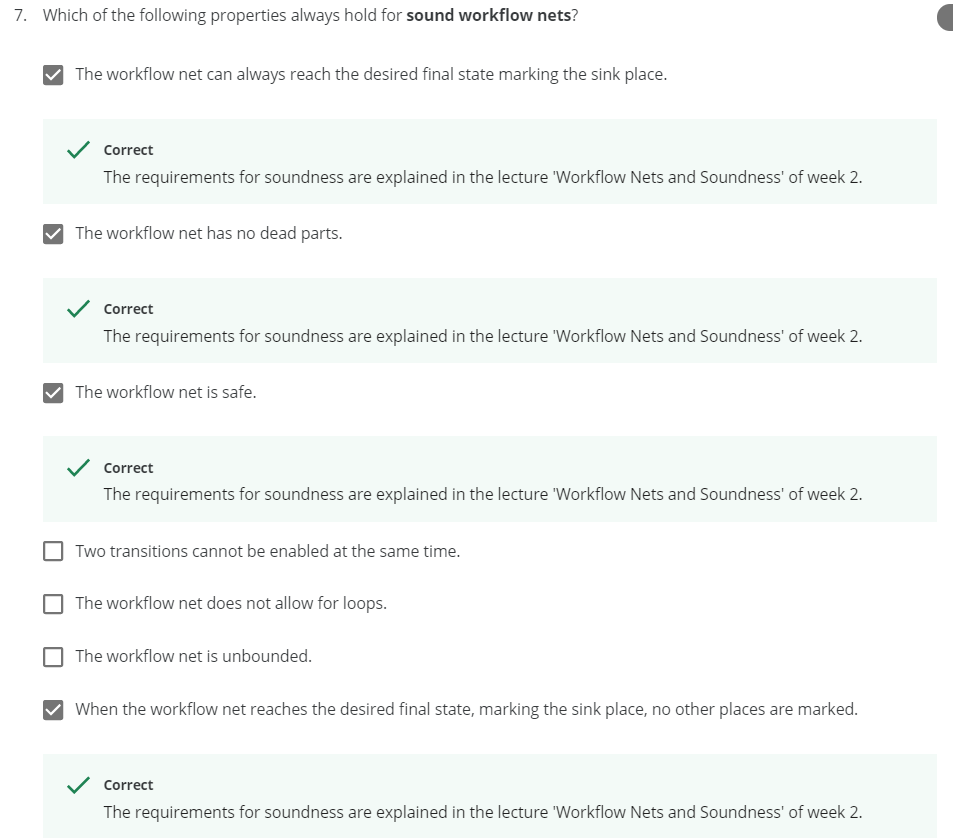




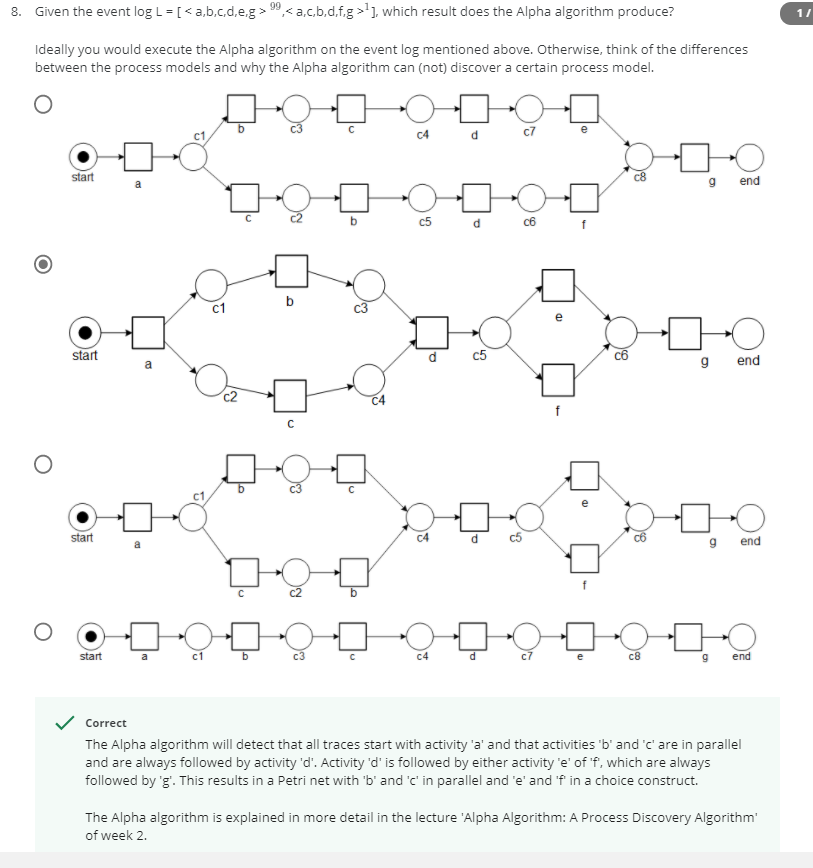
* A place P is k-bounded if there is no reachable marking with more than k tokens in P
* All places are bounded because at any point of time the tokens at place p are finite and some k-bounded
* Safeness means 1-boundness (k = 1)
* If all the places never have a two or more tokens, we call it a safe petri net
* In our case we have 3 tokens initially so it is not safe
* No matter what if we can fire a transition always then it is live
* A place is dead if we can’t fire that transition no matter what
* In our case, once all tokens are removed then we lose the live property
* Some point of time we can reach every transition so all places are not dead



* Workflow nets are a subclass of Petri nets
* Workflow nets have a well-defined start and end and should be free of obvious anomalies (soundness)
* A Workflow net has one source place(start) and one sink place(end) and all other nodes are on a path from source to sink

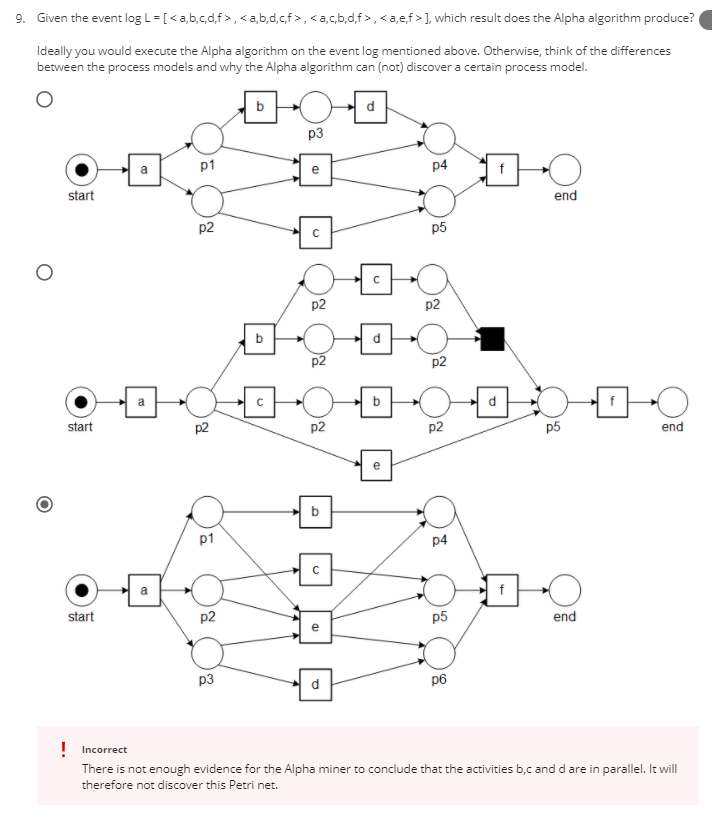


* A Workflow net is sound if it has below four properties:
  + Safeness - 1-bounded and places cannot hold multiple tokens at the same time
  + Proper completion - If a token reached sink place (end) then all other places should be empty
  + Option to complete - Given workflow net must have an end (sink place)
  + Absence of dead parts - For any transition there is a firing sequence enabling it



Fourth option: Wrong; It doesn’t work for <acbdfg>

All other three options provides some model if we trace it we get the actual event log but only second option is correct because it follows the alpha algorithm and simple compared to other two options (Also refer the explanation in the screenshot)



If you trace it, first petri net is correct for the given event log

