## 2002 Q1 Problem 4: CSP/Resource Allocation (30 points) OUzuner

### Part A (15 points)

You have a nightmare. You have become the MIT classroom scheduler, and you must not only assign TAs to classrooms, but also, you must do it using 6.034 ideas.

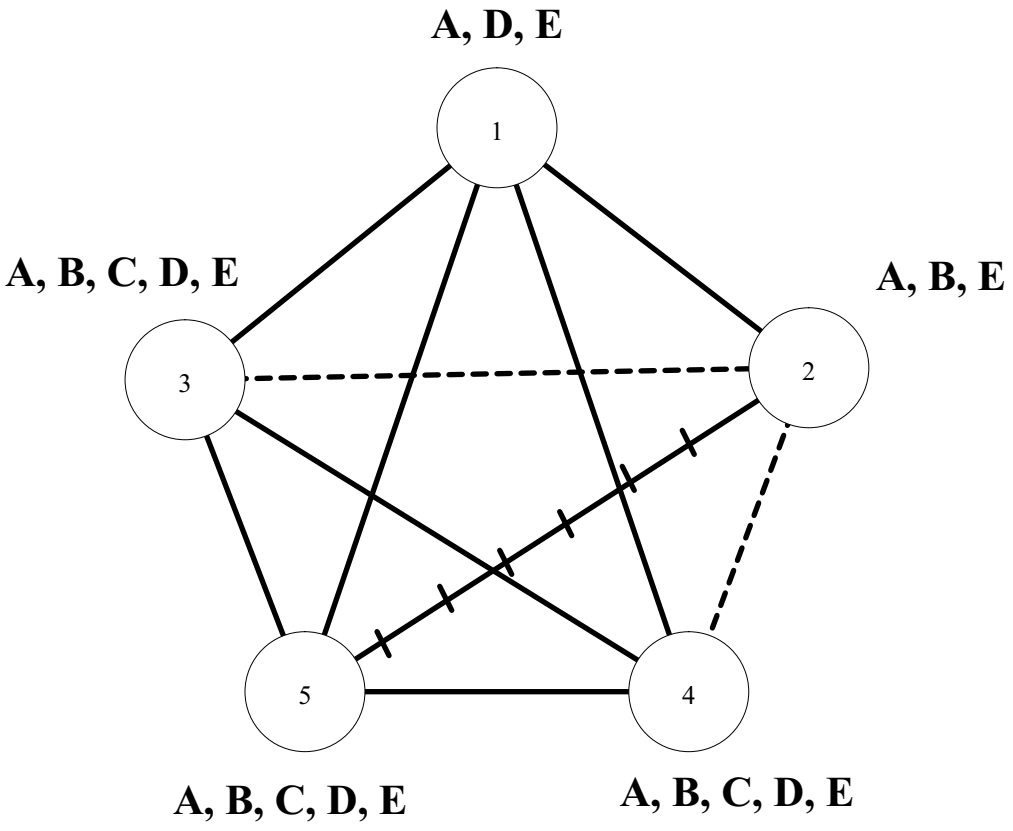
In particular, there are five TAs: A, B, C, D, and E, all to be scheduled at noon.

- Some will not accept assignment to particular rooms.
- Some are allergic to each other and cannot be assigned to classrooms that are close together.
- Others like each other and cannot be assigned to classrooms that are far apart.

And there are five rooms: 1, 2, 3, 4 and 5.

- 2-3 and 2-4 are close together
- 2-5 are far apart
- Others pairs are in between, neither close together nor far apart.

When all the information is assembled, you have the following diagram:



- Each TA's name is listed next to the rooms to which s/he is willing to be assigned.

- The solid lines, —, identify other room pairs that are neither close together nor far apart. Such a pair of rooms cannot have the same person in both rooms. Thus, no solid line can connect the following pairs:

  A-A, B-B, C-C, D-D, E-E

- The dashed lines, - - -, identify room pairs that are close together. Such a pair of rooms cannot have the same person in both rooms, nor can such a pair of rooms have two people who are allergic to each other. That is, no dashed line can connect the following pairs:

  A-A, B-B, C-C, D-D, E-E, A-B, B-D, B-E, C-D

- The hatched line, -|-|-|-, identifies a room pair that is far apart. This pair of rooms cannot have the same person in both rooms, nor can this pair of rooms have two people who like each other. That is, no hatched line can connect the following pairs:

  A-A, B-B, C-C, D-D, E-E, A-E, A-D, B-D

Your job is to find a consistent set of assignments using **the version of constraint checking** outlined below:

- You are to use depth-first search with backup.
- You are to use constraint propagation (also known as arc-consistency) to eliminate values inconsistent with values assigned by the depth-first search at neighboring nodes.
- Your constraint propagation is to ***continue through domains reduced to unique assignments.***
- You are to terminate search and backup whenever a domain becomes empty.
- Assignments are made to rooms in numerical order; assignments are to be tried in alphabetical order.

Show your results by completing the tree **on the next page**. Be sure to include in your tree any assignment that you make, including those that lead to one or more empty domains.

For this section, work both on the constraint graph and the tree.

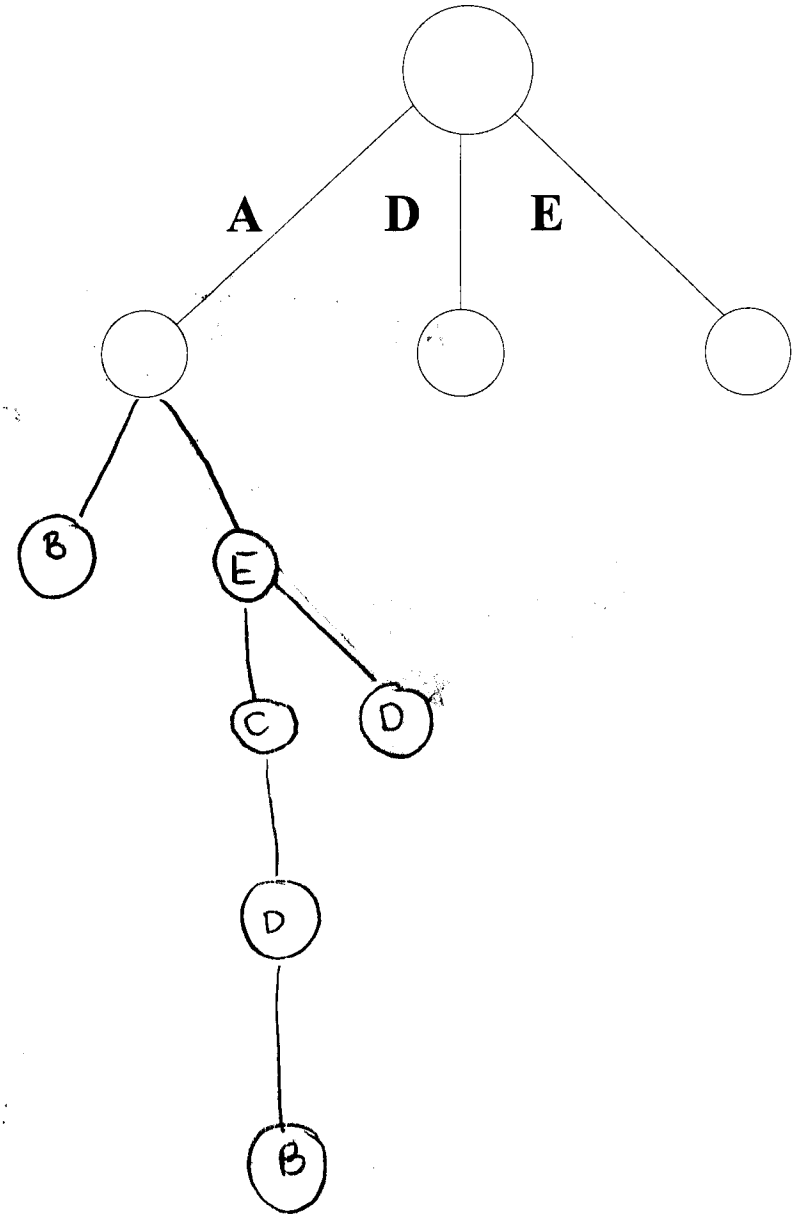
Possible room 1 Assignments

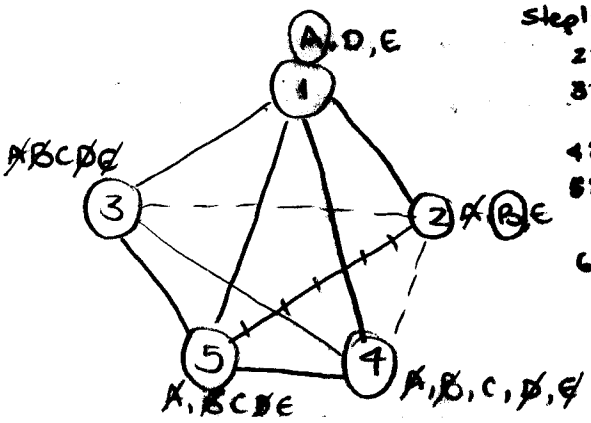
Possible room 2 Assignments

Possible room 3 Assignments

Possible room 4 Assignments

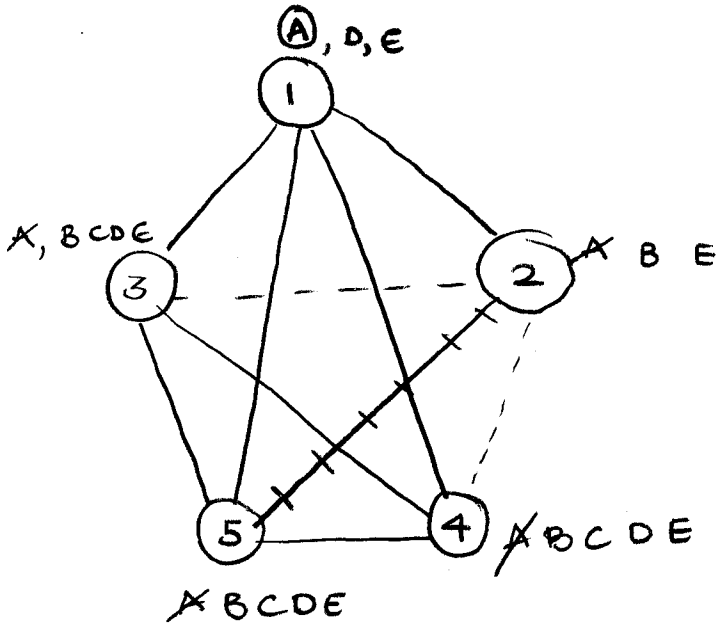
Possible room 5 Assignments



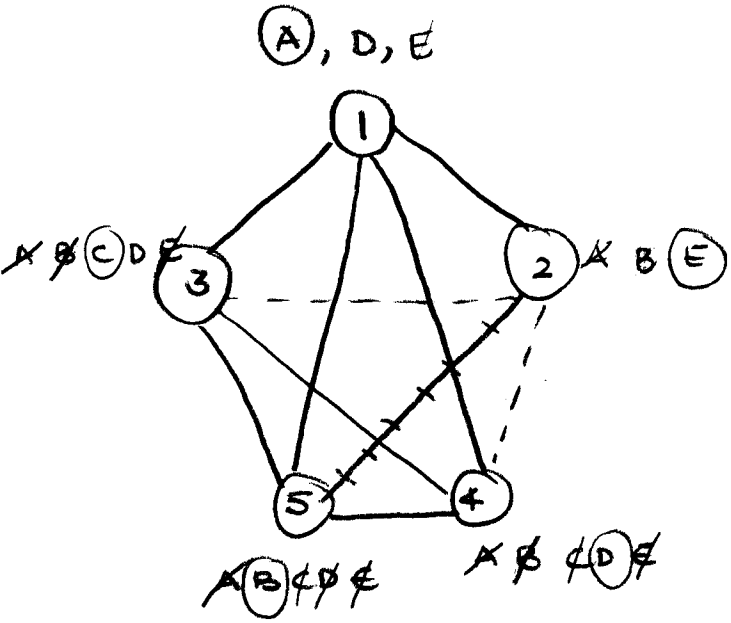




Step 1: Assign A to 1, eliminate A from all other domains
2: Assign B to 2, eliminate B from all other domains
3: Eliminate D and E from 3 and 4 because of B-D & B-E dashed line constraints.
4: Eliminate D from 5 because of B-D hatched line constraint
5: domain 3 and 4 are singletons. Propagate constraint from 3, 4 becomes an empty domain!
6. Backup, assign E to 2, undo the eliminations done due to assignment of B to 2.

New Constraint graph with all eliminations due to assignment of B to 2 undone.




Step 7: Assign E to 2, eliminate E from all other domains (except 1 since we propagate down the tree)




8. eliminate B from 3 and 4 because of dashed lines
9. Assign C to 3 (draw on tree)
10. Eliminate C from all other domains
11. Continue constraint propagation from 4, since 4 is now a singleton. Eliminate D from 5.
12. Assign D to 4.
13. Assign B to 5.

## Part B (15 points)

It occurs to you that instead of assigning TAs to rooms, you could assign rooms to TAs (the so-called dual problem).

- Indicate which rooms are allowed for each TA.

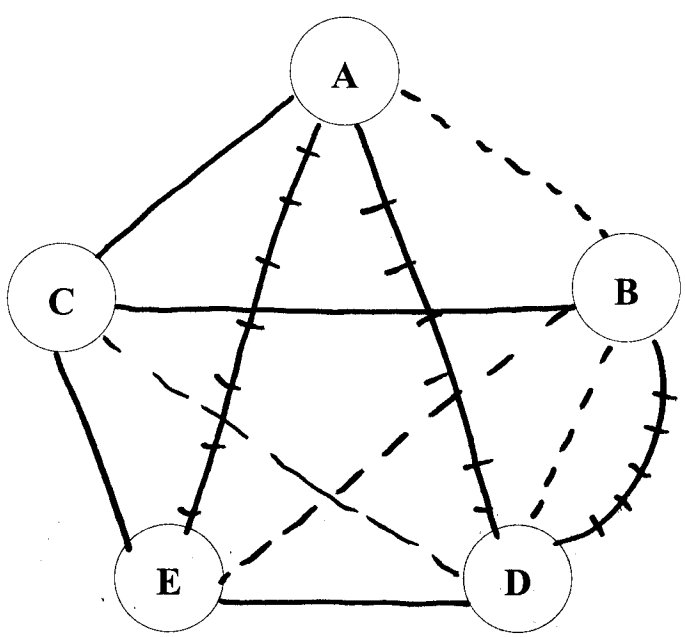
logic: If A is in domain of 1, 1 should be in domain of A.
A is in domain of 1, 2, 3, 4, 5. All 5 should be in domain of A

| | |
|---|---|
| A | 1 2 3 4 5 |
| B | 2 3 4 5 ( appears in domain of 2 3 4 5) |
| C | 3 4 5 |
| D | 1 3 4 5 |
| E | 1 2 3 4 5 |

- Now, decorate the diagram below using dashed lines, - - -, to indicate no one classroom can be assigned to two different TAs and that classrooms that are close together cannot be assigned to people who are allergic to each other.

- Next use hatched lines, -|-|-|-, to indicate no one classroom can be assigned to two different TAs and that classrooms that are far apart cannot be assigned to people who like each other.

- Finally, use solid lines, —, to indicate only that no one classroom can be assigned to two different TAs.




Just read the constraints between TA's given in the question.

① dashed lines between A-B, B-D, B-E, C-D

② hatched between A-E, A-D, B-D

③ straight lines between all other pairs.
i.e. A-C, C-E, C-B, E-D