Osman Emre Bilici
2171353

1)

a)

Variables V = {111,213,223,315,331,351}
Domain D = {A1,A2,B1,B2,B3,C} where A, B, C refers to 9.30,13.30, 16.30 respectively and meaning of the number near the letter is class number. (i.e, A1 = BMB1, 9.30)
Constraints = {

111: {A1,A2,B1,B2,B3,C},

(213,223) : { (A1,B1), (A1,C), (A1,B2), (A1,B3), (A2,B1), (A2,B2), (A2,B3), (A2,C), (B1,C), (B2,C), (B3,C)},
(223,213) : { (A1,B1), (A1,C), (A1,B2), (A1,B3), (A2,B1), (A2,B2), (A2,B3), (A2,C), (B1,C), (B2,C), (B3,C)},

(315,331,351) : { (A1,B1,C), (A2,B1,C), (A1,B2,C), (A2,B2,C) },
(315,351,331) : { (A1,B1,C), (A2,B1,C), (A1,B2,C), (A2,B2,C) },
(331,315,351) : { (A1,B1,C), (A2,B1,C), (A1,B2,C), (A2,B2,C) },
(331,351,315) : { (A1,B1,C), (A2,B1,C), (A1,B2,C), (A2,B2,C) },
(351,331,315) : { (A1,B1,C), (A2,B1,C), (A1,B2,C), (A2,B2,C) },
(351,315,331) : { (A1,B1,C), (A2,B1,C), (A1,B2,C), (A2,B2,C) }
}

b)

Each domain should be used only once. So I did not show that deleting that used nodes from other variables.

	111	**213**	**223**	**315**	**331**	**351**
initial domain	A1,A2,B1, B2,B3,C	A1,A2,B1, B2,B3,C	A1,A2,B1, B2,B3,C	A1,A2,B1, B2,B3,C	A1,A2,B1,B 2,B3,C	A1,A2,B1,B2, B3,C
After 111 = B2	B2	A1,A2,B1, B3,C	A1,A2,B1, B3,C	A1,A2,B1, B3,C	A1,A2,B1,B 3,C	A1,A2,B1,B3, C
After 213 = A1	B2	A1	B1,B3,C	A2,B1,B3, C	A2,B1,B3, C	A2,B1,B3,C
After 223 = C	B2	A1	C	A2,B1,B3	A2,B1,B3	A2,B1,B3
After 315 = A2	B2	A1	C	A2	B1,B3	B1,B3
After 331 = B3	B2	A1	C	A2	B3	No possible values

c)

	111	**213**	**223**	**315**	**331**	**351**
initial domain	A1,A2,B1, B2,B3,C	A1,A2,B1, B2,B3,C	A1,A2,B1, B2,B3,C	A1,A2,B1, B2,B3,C	A1,A2,B1, B2,B3,C	A1,A2,B1, B2,B3,C
After 111 = B2	B2	A1,A2,B1, B3,C	A1,A2,B1, B3,C	A1,A2,B1, B3,C	A1,A2,B1, B3,C	A1,A2,B1, B3,C
After 213 = A1	B2	A1	B1,B3,C	A2,B1,B3, C	A2,B1,B3, C	A2,B1,B3, C
After 223 = C	B2	A1	C	A2,B1,B3	A2,B1,B3	A2,B1,B3
After 315 = A2	B2	A1	C	A2	B1,B3	B1,B3
	B2	A1	C	A2	~~B1~~,~~B3~~	B1,B3
	B2	A1	C	A2	No possible values	B1,B3

After assigned 111 = B2, all other variables are consistent. (They lose value B2, so they are checked.)

After assigned 213 = A1, 223 → 213 and 213 → 223 are consistent after deleting A1 from 223. Others are also consistent.

After assigned 223 = C, 213 → 223 is consistent. But all courses with grade 3, lose value C and they should be checked. For every value of 315, 331,351, there are some allowed values on 315, 331, 351. So they are consistent.

After assigned 315 = A2, 331 → 315 is consistent. 315 → 331 is consistent. 315 → 351 is consistent. 351 → 315 is consistent. 331 → 351 is not consistent. Removing bad values B1 and B3. There are no possible values.

2)

This is my tracking:

Node D : **α** = 5, **β** = ∞
Node B : **β** = 5
Node E : **α** = 6, **β** = 5 Since **α** ≥ **β**, cut off → 10 is never looked.
Node A : **α** = 5
Node C: **α** = 5, **β** = ∞
Node F : **α** = 5, **β** = ∞ **α** = max(2,5)
Node C : **α** = 5, **β** = 2 **β** = min(∞,2) where 2 is best value of F.
α ≥ **β**, cut off → Node G is never looked.

These are the final values:

Node A : **α** = 5 , value = 5
Node B : **β** = 5 , value = 5
Node C : **α** = 5, **β** = 2, value = 2
Node D : **α** = 5, **β** = ∞, value = 5
Node E : **α** = 6, **β** = 5 , value = 6
Node F : **α** = 5, **β** = ∞ , value = 2

3.a)

This table shows the premises count. Whenever the premises of rule satisfied, count is decreased. At the end, premises of K ⇒ L is zero and agenda is empty. So it is proved by forward chaining.

K ⇒ L	1	1	1	1	1	1	1	1	1	1	1	0	0
I ∧ J ⇒ K	2	2	2	2	2	2	2	2	2	1	0	0	0
G ∧ H ⇒ I	2	2	2	2	2	2	2	1	0	0	0	0	0
H ∧ D ⇒ J	2	2	2	2	1	1	1	0	0	0	0	0	0
E ∧ H ⇒ G	2	2	2	2	2	1	1	0	0	0	0	0	0
E ∧ F ⇒ H	2	2	2	2	2	1	0	0	0	0	0	0	0
G ∧ A ⇒ E	2	1	1	1	1	1	1	1	1	1	1	1	1
A ∧ B ⇒ E	2	1	0	0	0	0	0	0	0	0	0	0	0
B ∧ C ⇒ F	2	2	1	0	0	0	0	0	0	0	0	0	0
A	0	0	0	0	0	0	0	0	0	0	0	0	0
B	0	0	0	0	0	0	0	0	0	0	0	0	0
C	0	0	0	0	0	0	0	0	0	0	0	0	0
D	0	0	0	0	0	0	0	0	0	0	0	0	0
Agenda	A,B,C,D	B,C,D	C,D,E	D,E,F	E,F	F	H	G,J	J,I	I	K	L	

b)

I show my work by tracking the selected nodes.

1. K ⇒ L
2. I ∧ J ⇒ K
3. G ∧ H ⇒ I
4. E ∧ H ⇒ G
5. G ∧ A ⇒ E — this is not suitable since one of the premises is not satisfied yet.
6. **A ∧ B ⇒ E** — A and B satisfied. So E is satisfied.
7. E ∧ F ⇒ H
8. **B ∧ C ⇒ F** — B and C satisfied. So F is satisfied.
9. **E ∧ F ⇒ H** — E and F satisfied. So H is satisfied.
10. **E ∧ H ⇒ G** — E and H satisfied. So G is satisfied.
11. **G ∧ H ⇒ I** — G and H satisifed. So I is satisfied.
12. **H ∧ D ⇒ J** — H and D satisfied. So J is satisfied.
13. **I ∧ J ⇒ K** — I and J satisfied. So K is satisfied.
14. **K ⇒ L** — K is satisfied. So L satisfied.