人工智能原理
Assignment 2

Problem 1 (a)

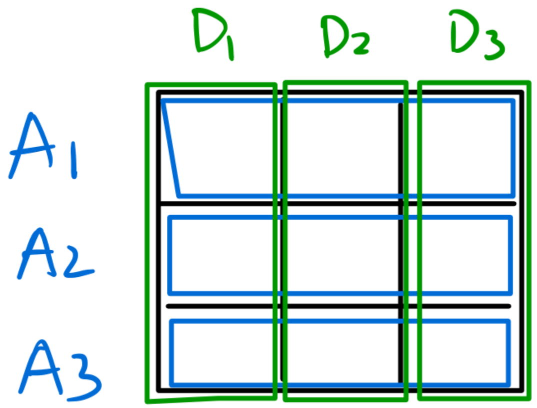
Give precise formulations for each of the following as CSPs: Provide the ***variables***, ***domains*** and ***constraints***.

(a) Crossword puzzle: shown below. We want to find six three-letter words: three words read across (A1, A2, and A3) and three words read down (D1, D2, and D3). Each word must be chosen from the list of forty possible words.

A1,D1	D2	D3
A2		
A3		

Word list:
add, ado, age, ago, aid,
ail, aim, air, and, any,
ape, apt, arc, are, ark,
arm, art, ash, ask, auk,
awe, awl, aye, bad, bag,
ban, bat, bee, boa, ear,
eel, eft, far, fat, fit,
lee, oaf, rat, tar, tie.

(a) Crossword Puzzle



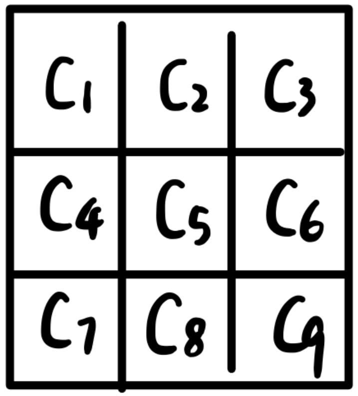
1. CSP Formulation 1

Variables: A_1, A_2, A_3 (A_i is the i^{th} array)

Domain for A_i: $D = \{w | w\ is\ a\ word\ in\ the\ list\}$

Constraints: $A_{11}A_{21}A_{31},\ A_{12}A_{22}A_{32}, A_{13}A_{23}A_{33} \in D$

2. CSP Formulation 2



Variables: $c_1, c_2, \ldots, c_9$

Domain for c_i: $\mathrm{A} = \{\alpha | \alpha\ is\ a\ letter\ in\ the\ list\}$

Constraints: $c_1c_2c_3, c_4c_5c_6, c_7c_8c_9,$
$c_1c_4c_7, c_2c_5c_8, c_3c_6c_9 \in D$

Problem 1 (b)

Independent set: Given a graph and a number k, find an independent set of size k, that is, a set of k vertices, no two of which are adjacent.

- Suppose the graph G has n vertices, and $k \leq n$
- CSP Formulation 1, suppose the graph G has n vertices

Variables: $v_1, v_2, \ldots, v_n$

Domain for v_i: $D = \{1, 0\}\ for\ i = 1, 2, \ldots, n$

($v_i = 1$ iff the i^{th} vertex is in the indep. set)

Constraints: $\sum_{i=1}^{n} v_i = k$ (k vertices in the set)

$\forall i, j \in \{1, 2, \ldots, n\}. \left(v_i = 1 \wedge v_j = 1\right) \rightarrow \neg adj(v_i, v_j)$

(if both the i^th and j^th vertices are in the set, they are not adjacent)

(b) Independent set: Given a graph and a number k, find an independent set of size k, that is, a set of k vertices, no two of which are adjacent.

- CSP Formulation 2

Suppose we name the vertices $v_1, v_2, \dots, v_n$

Variables: $x_1, x_2, \dots, x_k$

Domain for x_i: $\{v_1, v_2, \dots, v_n\}$, for $i = 1, 2, \dots, k$

Constraints: $\forall x_i, x_j \in \{v_1, v_2, \dots, v_n\}. \neg adj(x_i, x_j)$

$AllDiffenrent(x_1, x_2, \dots, x_k)$

Problem 1 (c)

Crypto-arithmetic puzzle: SEND + MORE = MONEY. We want to replace each letter by a different digit so that the equation is correct.

- CSP Formulation 1
- Variables: S, E, N, D, M, O, R, Y
  Domains: $\{0, 1, \ldots, 9\}$
  Constraints:
  $AllDifferent(S, E, N, D, M, O, R, Y)$
  $1000 * (S + M) + 100 * (E + O) + 10 * (N + R) + D + E = M * 10000 + O * 1000 + N * 100 + E * 10 + Y$
- Not good for partial assignment

(c) Crypto-arithmetic puzzle

- CSP Formulation 2

Variables: $S, E, N, D, M, O, R, Y, c_1, c_2, c_3$

Domain for S, E, N, D, M, O, R, Y: $\{0,1,2, \dots 9\}$

Domain for c_1, c_2, c_3 : $\{0,1\}$

Constraints: $AllDifferent(S, E, N, D, M, O, R, Y)$

$$D + E = c_1 * 10 + Y$$

$$N + R + c1 = c_2 * 10 + E$$

$$E + O + c2 = c_3 * 10 + N$$

$$S + M + c3 = M * 10 + O$$

$$\begin{array}{cccccccc} & & S & & E & & N & & D \\ + & & M & c_3 & O & c_2 & R & c_1 & E \\ \hline & M & O & & N & & E & & Y \end{array}$$

Problem 2 (a)

Consider a scheduling problem, where there are five activities to be scheduled in four time slots. Suppose we represent the activities by the variables A, B, C, D, E, where the domain of each variable is $\{1,2,3,4\}$ and the constraints are $A > D, D > E, C \neq A, C > E, C \neq D, B \geq A, B \neq C,$ and $C \neq D + 1$.

(a) Find the first solution by using the **Forward Checking** algorithm with the **MRV heuristics**, breaking ties in alphabetic order. Assign values in the current domain of each variable in **increasing order**. At each node indicate

i. The **variable** being instantiated and the value being assigned to it.

ii. The **CurDom** for each variable.

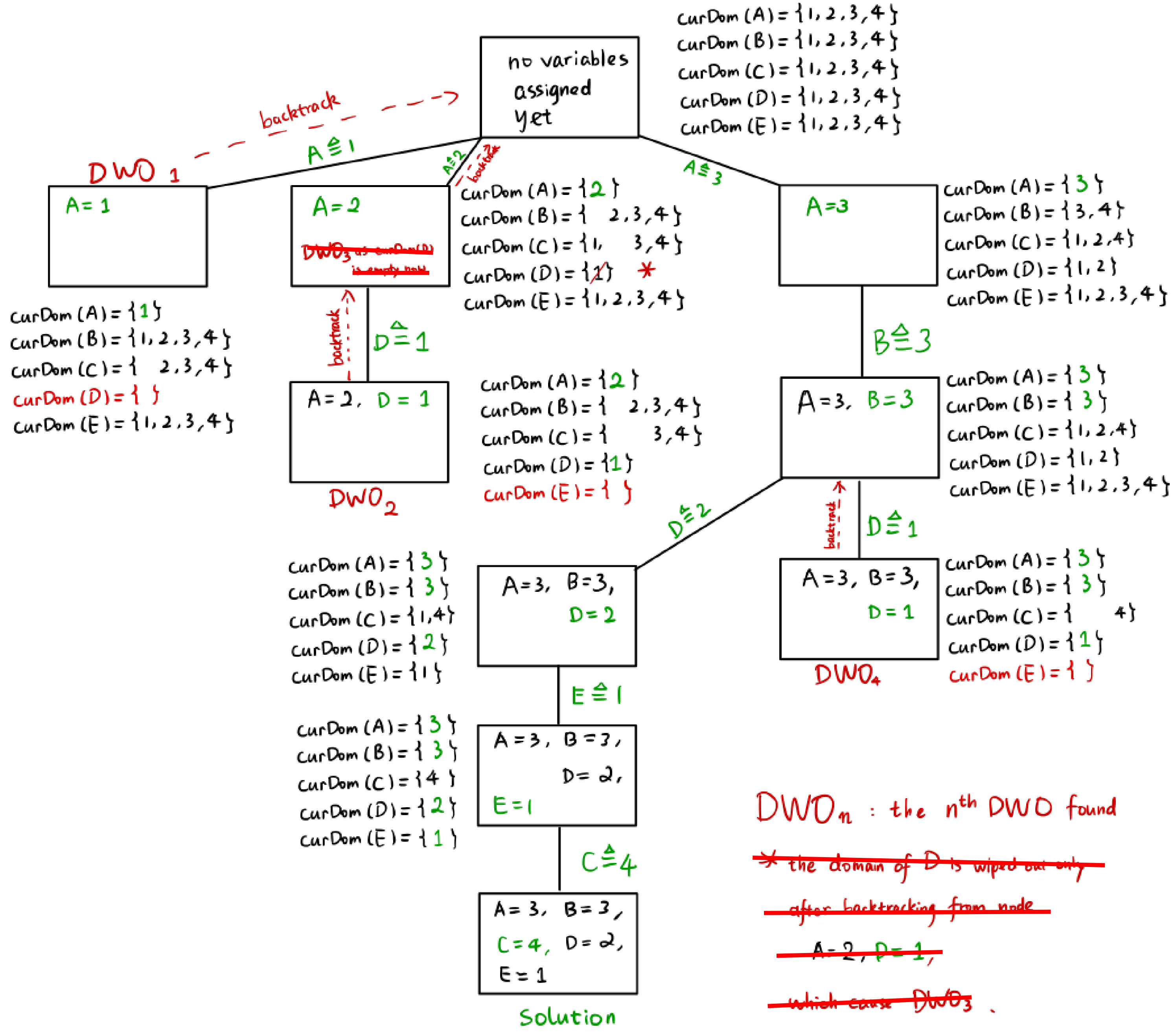
iii. Mark any node with an empty CurDom with **DWO**.

Problem 2 (a)

For part (a), pay attention to

- Forward Checking with MRV heuristics
- If two variables have equal-sized domains, break tie with α-order
- Choose values in an increasing order: 1 first, then 2, …
- At each node mark the domains for the unassigned variables, specify the current domain, mark DWO (Domain Wiped Out) if empty domain exists.

(a) Forward Checking, MRV heuristics, tie-breaking: alphabetic order, increasing value



Problem 2 (b)

Consider a scheduling problem, where there are five activities to be scheduled in four time slots. Suppose we represent the activities by the variables A, B, C, D, E, where the domain of each variable is $\{1,2,3,4\}$ and the constraints are $A > D, D > E, C \neq A, C > E, C \neq D, B \geq A, B \neq C,$ and $C \neq D + 1$.

(b) Enforce **GAC** on the constraints, and give the resultant variable domains. You should <u>show which values of a domain are removed</u> at each step, and <u>which arc is responsible for removing the value</u>. Then use the GAC algorithm to find the **first solution**.

Problem 2 (b)

For part (b), pay attention to

- GAC enforcement works both way, e.g., for $A > D$, check both from A to D and from D to A
- Enforce GAC before even the first assignment!
- Exhaustively enforce GAC, don’t leave out any constraints
  - For example, suppose the constraint $A > D$ is already checked, if in checking $D > E$ the domain of D is changed, we need to go back and check for $A > D$ again
- When removing values, mark with the responsible arc

(b) GAC, mark removed values with responsible arc

no variables assigned yet

$A \hat{=} 3$

A = 3

$B \hat{=} 3$

A = 3, B = 3

$C \hat{=} 4$

A = 3, B = 3
C = 4

$D \hat{=} 2$

A = 3, B = 3
C = 4, D = 2

$E \hat{=} 1$

A = 3, B = 3
C = 4, D = 2
E = 1

Solution

1^{st} GAC

$curDom(A) = \{\cancel{1}, \cancel{2}, 3, 4\}$
$curDom(B) = \{\cancel{1}, \cancel{2}, 3, 4\}$
$curDom(C) = \{\cancel{1}, 2, 3, 4\}$
$curDom(D) = \{\cancel{1}, 2, 3, \cancel{4}\}$
$curDom(E) = \{1, 2, \cancel{3}, \cancel{4}\}$

removed values	arc	removed values	arc
A = 1	$A > D$	A = 2	$A > D$
B = 1	$B \geq A$	B = 2	$B \geq A$
C = 1	$C > E$		
D = 1	$D > E$	D = 4	$A > D$
E = 3	$D > E$	E = 4	$D > E$

2^{nd} GAC

$curDom(A) = \{3\}$
$curDom(B) = \{3, \cancel{4}\}$
$curDom(C) = \{\cancel{2}, \cancel{3}, 4\}$
$curDom(D) = \{2, \cancel{3}\}$
$curDom(E) = \{1, \cancel{2}\}$

removed values	arc	removed values	arc
B = 4	$B \neq C$		
C = 3	$C \neq A$	C = 2	$C \neq D$
D = 3	$A > D$		
E = 2	$D > E$		

3^{rd} GAC

$curDom(A) = \{3\}$
$curDom(B) = \{3\}$
$curDom(C) = \{4\}$
$curDom(D) = \{2\}$
$curDom(E) = \{1\}$

4^{th} GAC

$curDom(A) = \{3\}$
$curDom(B) = \{3\}$
$curDom(C) = \{4\}$
$curDom(D) = \{2\}$
$curDom(E) = \{1\}$

5^{th} GAC

$curDom(A) = \{3\}$
$curDom(B) = \{3\}$
$curDom(C) = \{4\}$
$curDom(D) = \{2\}$
$curDom(E) = \{1\}$

Problem 3

Determine whether the following sentence is valid using resolution:

$$(\exists x \forall y P(x,y) \vee \exists x \forall y Q(x,y)) \rightarrow \exists x \forall y (P(x,y) \vee Q(x,y))$$

- A sentence S is valid iff $\vDash S$ iff $\neg S \rightarrow ()$ (derivation of resolution)
- Let $S = (\exists x \forall y P(x,y) \vee \exists x \forall y Q(x,y)) \rightarrow \exists x \forall y (P(x,y) \vee Q(x,y))$
- We need to prove $\neg S \rightarrow ()$ by using rules of resolution
- But first we need to convert it to clausal form

Conversion to Causal Form

1. Eliminate Implications.
2. Move Negations inwards (and simplify ¬¬).
3. Standardize Variables.
4. Skolemize.
5. Convert to Prenex Form.
6. Distribute disjunctions over conjunctions.
7. Flatten nested conjunctions and disjunctions.
8. Convert to Clauses

1. Eliminate Implications.

$$\neg S = \neg\{(\exists x \forall y P(x,y) \lor \exists x \forall y Q(x,y)) \rightarrow \exists x \forall y (P(x,y) \lor Q(x,y))\}$$
$$\equiv \neg\{\neg(\exists x \forall y P(x,y) \lor \exists x \forall y Q(x,y)) \lor \exists x \forall y (P(x,y) \lor Q(x,y))\}$$

2. Move Negations inwards

$$\neg\neg(\exists x \forall y P(x,y) \lor \exists x \forall y Q(x,y)) \land \neg\exists x \forall y (P(x,y) \lor Q(x,y))$$
$$(\exists x \forall y P(x,y) \lor \exists x \forall y Q(x,y)) \land \forall x \exists y \neg(P(x,y) \lor Q(x,y))$$
$$(\exists x \forall y P(x,y) \lor \exists x \forall y Q(x,y)) \land \forall x \exists y (\neg P(x,y) \land \neg Q(x,y))$$

3. Standardize Variables

$$(\exists s \forall t P(s,t) \lor \exists u \forall v Q(u,v)) \land \forall x \exists y (\neg P(x,y) \land \neg Q(x,y))$$

$$\left(\exists s \forall t P(s,t) \lor \exists u \forall v Q(u,v)\right) \land \forall x \exists y (\neg P(x,y) \land \neg Q(x,y))$$

4. Skolemize (Eliminate existential quantifiers)
   - Introduce constant symbols a and b, and function symbol f
   - Use a to replace s, b to replace u,$f(x)$ to replace y

$$\left(\forall t P(a,t) \lor \forall v Q(b,v)\right) \land \forall x (\neg P\left(x, f(x)\right) \land \neg Q(x, f(x)))$$

5. Convert to Prenex Form

$$\forall t \forall v \forall x. \left(P(a,t) \lor Q(b,v)\right) \land (\neg P\left(x, f(x)\right) \land \neg Q(x, f(x)))$$

$$\forall t \forall v \forall x. \big(P(a,t) \vee Q(b,v)\big) \wedge (\neg P\big(x, f(x)\big) \wedge \neg Q(x, f(x)))$$

- Resulting Clauses (note that a, b are constants)

1. $[P(a,t), Q(b,v)]$
2. $\neg P\big(x, f(x)\big)$
3. $\neg Q\big(x, f(x)\big)$

- Derivation
- R[1, 2] $\{x = a, t = f(a)\}$

4. $Q(b,v)$

- R[3, 4] $\{x = b, v = f(b)\}$

5. $()$

$\{x = a, t = f(a)\}$

1. $[P(a,t), Q(b,v)]$
2. $\neg P(a, f(a))$

$\{x = b, v = f(b)\}$

3. $\neg Q\big(b, f(b)\big)$
4. $Q(b, f(b))$

Logical Interpretations

- In FOL, the sentence $R(a, b)$ involves

1. A binary predicate symbol R
2. Two constant symbols A and B

- $R(a, b)$ is syntactically well-formed. But what does it **mean**?

- Consider in a world where

1. There are two people: $a = John$ and $b = Mary$
2. R is the relation $Likes$: $John$ likes $Mary$, $Mary$ likes $Mary$

- In this world, $R(a, b)$ describes $John\ likes\ Mary$

Logical Interpretations

- An Interpretation provides the formula $R(a,b)$ with a meaning $Likes(John, Mary)$.
- Formally, an $interpretation\ M$ in FOL is a pair $< D, I >$
  1. D: any nonempty set of objects (domain)
  2. I: a mapping from nonlogical symbols (predicate and function symbols) to functions and relations over D (interpretation mapping)
- A sentence can evaluate to $true$ or $false$ in an interpretation.
- For formulae with free variables, an assignment is also needed.

- For the formula $R(a, b)$ (in this case, a sentence)
- A possible interpretation $M \; = < D, I >$
  - $D \; = \; \{John, \; Mary\}$
  - $I(a) = \; John, \; I(b) = \; Mary$
  - $I(R) \; = \; Likes \; = \; \{(John, Mary), (Mary, Mary)\}$
  - In M,$R(a, b)$ is $true$
- Another possible interpretation $M' =< D, I' >$
  - $D \; = \; \{John, \; Mary\}$
  - $I'(a) = \; John, \; I'(b) = \; Mary$
  - $I'(R) \; = \; Likes' \; = \; \{(John, John), (Mary, Mary)\}$
  - In M',$R(a, b)$ is $false$
- A formula (or a set of formulae) could have many different interpretations.

- For the propositional subset of FOL, we can ignore the domain D completely, and think of an interpretation M as simply being a mapping, I, from the propositional symbols to either 0 or 1. (In Mathematical Logic, we use 0 and 1 to denote the truth value) [Brachman and Levesque, 2004]
    - Proposition $p \equiv$ `John likes Mary` is $true$ in the interpretation where $I(p) = 1$

Problem 4 (a)

Victor has been murdered, and Arthur, Bertram, and Carleton are the only suspects (meaning exactly one of them is the murderer). Arthur says that Bertram was the victim's friend, but that Carleton hated the victim. Bertram says that he was out of town the day of the murder, and besides, he didn't even know the guy. Carleton says that he saw Arthur and Bertram with the victim just before the murder. You may assume that everyone – except possibly for the murderer – is telling the truth.

(a) Use Resolution to find the murderer. In other words, formalize the facts as a set of clauses, prove that there is a murderer, and extract his identity from the derivation.

- First formalize the facts
- Let a, b, c be 3 constants to represent 3 suspects respect.,

$M(x)$ = x is the murderer, $F(x)$ = x is a friend of the victim's, $Al(x)$ = x has an alibi.

- From the problem we know:
- $M(a) \vee M(b) \vee M(c)$
- $M(a) \rightarrow \neg M(b) \wedge \neg M(c)$
- $M(b) \rightarrow \neg M(a) \wedge \neg M(c)$
- $M(c) \rightarrow \neg M(a) \wedge \neg M(b)$

- Formalize the facts:
- Single murderer
- $M(a) \vee M(b) \vee M(c)$
- $M(a) \rightarrow \neg M(b) \wedge \neg M(c)$
- $M(b) \rightarrow \neg M(a) \wedge \neg M(c)$
- $M(c) \rightarrow \neg M(a) \wedge \neg M(b)$
- Statements
- $\neg M(a) \rightarrow F(b) \wedge \neg F(c)$
- $\neg M(b) \rightarrow Al(b) \wedge \neg F(b)$
- $\neg M(c) \rightarrow \neg Al(a) \wedge \neg Al(b)$
- $\forall x. Al(x) \rightarrow \neg M(x)$
- Query: there is a murderer
- $\exists x. M(x)$

- Convert into clausal forms
- Eliminate the implication

$$M(a) \to \neg M(b) \wedge \neg M(c)$$
$$\equiv \neg M(a) \vee \left(\neg M(b) \wedge \neg M(c)\right)$$
$$\equiv (\neg M(a) \vee \neg M(b)) \wedge (\neg M(a) \vee \neg M(c))$$

- Eliminate the quantifiers
  - From $\forall x. Al(x) \to \neg M(x)$ to $\neg Al(x) \vee \neg M(x)$
  - Negate the query and convert into clausal form
  - $\neg \exists x. M(x) \equiv \forall x. \neg M(x)$, then drop the universal quantifier:

$$\neg M(x)$$

- Finally the KB and the negated query in clauses
- 1. $M(a) \vee M(b) \vee M(c)$
- 2. $\neg M(a) \vee \neg M(b)$
- 3. $\neg M(a) \vee \neg M(c)$
- 4. $\neg M(b) \vee \neg M(c)$
- 5. $M(a) \vee F(b)$
- 6. $M(a) \vee \neg F(c)$
- 7. $M(b) \vee Al(b)$
- 8. $M(b) \vee \neg F(b)$
- 9. $M(c) \vee \neg Al(a)$
- 10. $M(c) \vee \neg Al(b)$
- 11. $\neg Al(x) \vee \neg M(x)$
- 12. $\neg M(x)$

- Derivation of Resolution
- R[7, 10]

  13. $M(b) \vee M(c)$
- R[5, 8]

  14. $M(a) \vee M(b)$
- R[3, 13]

  15. $\neg M(a) \vee M(b)$
- R[14, 15]

  16. $M(b)$
- R[12, 16]$\{\boldsymbol{x} = b\}$
- 17. ()

- 7. $M(b) \vee Al(b)$
- 10. $M(c) \vee \neg Al(b)$
- 5. $M(a) \vee F(b)$
- 8. $M(b) \vee \neg F(b)$
- 3. $\neg M(a) \vee \neg M(c)$
- 13. $M(b) \vee M(c)$
- 14. $M(a) \vee M(b)$
- 15. $\neg M(a) \vee M(b)$
- 12. $\neg M(x)$
- 16. $M(b)$

Problem 4 (b)

(b) Suppose we discover that we were wrong – we cannot assume that there was only a single murderer (there may have been a conspiracy). Show that in this case the facts do not support anyone's guilt. <u>In other words, for each suspect, present a logical interpretation that supports all the facts but where that suspect is innocent and the other two are guilty.</u>

- For Arthur, find an interpretation in which the other two are guilty, similarly for Bertram and Carleton.

- Since there may be a conspiracy, we need to update the KB.

- Suspects (now conspiracies are possible)
- $M(a) \lor M(b) \lor M(c)$
- $\cancel{M(a) \rightarrow \neg M(b) \land \neg M(c)}$
- $\cancel{M(b) \rightarrow \neg M(a) \land \neg M(c)}$
- $\cancel{M(c) \rightarrow \neg M(a) \land \neg M(b)}$
- Statements
- $\neg M(a) \rightarrow F(b) \land \neg F(c)$
- $\neg M(b) \rightarrow Al(b) \land \neg F(b)$
- $\neg M(c) \rightarrow \neg Al(a) \land \neg Al(b)$
- $\forall x. Al(x) \rightarrow \neg M(x)$

- Consider the case: only Arthur is innocent
- A logical interpretation $M =< D, I >$
  - Domain: $D = \{Arthur, Bertram, Carleton\}$
  - Interpretation mapping:
    1. Constants: $I(a) = Arthur, I(b) = Bertram, I(c) = Carleton$
    2. Predicates:
       - $I(M) = \{Bertram, Carleton\}$
       - $I(F) = \{Bertram\}$
       - $I(Al) = \{Arthur\}$

The Updated KB

- $M(a) \lor M(b) \lor M(c)$
- $\neg M(a) \rightarrow F(b) \land \neg F(c)$
- $\neg M(b) \rightarrow Al(b) \land \neg F(b)$
- $\neg M(c) \rightarrow \neg Al(a) \land \neg Al(b)$
- $\forall x. Al(x) \rightarrow \neg M(x)$

- $M \vDash KB$, and Arthur is innocent
- The cases for Bertram and Carleton are similar.

Problem 4 (a)

- Formalize the facts
- Let Ma be the proposition that Arthur is the murderer, Mb and Mc for Bertram and Carleton
- Let Fb be the proposition that Bertram is a friend to victim, Fc for Carleton
- Let Aa be the proposition that Arthur has an alibi, Ba and Ca for Bertram and Carleton

- Formalize the KB
- Single murderer
- $Ma \vee Mb \vee Mc$
- $Ma \rightarrow \neg Mb \wedge \neg Mc$
- $Mb \rightarrow \neg Ma \wedge \neg Mc$
- $Mc \rightarrow \neg Ma \wedge \neg Mb$
- Statements
- $\neg Ma \rightarrow Fb \wedge \neg Fc$
- $\neg Mb \rightarrow Ab \wedge \neg Fb$
- $\neg Mc \rightarrow \neg Aa \wedge \neg Ab$

- Commonsense
- $Aa \rightarrow \neg Ma$
- $Ab \rightarrow \neg Mb$

- Convert the KB into clauses

1. $Ma \vee Mb \vee Mc$
2. $\neg Ma \vee \neg Mb$
3. $\neg Ma \vee \neg Mc$
4. $\neg Mb \vee \neg Mc$
5. $Ma \vee Fb$
6. $Ma \vee \neg Fc$
7. $Mb \vee Ab$
8. $Mb \vee \neg Fb$
9. $Mc \vee \neg Aa$
10. $Mc \vee \neg Ab$

- 11. $\neg Aa \vee \neg Ma$
- 12. $\neg Ab \vee \neg Mb$

- Resolution derivation steps
- R[7, 10]

13. $Mb \vee Mc$

- R[5, 8]

14. $Ma \vee Mb$

- R[3, 13]

15. $\neg Ma \vee Mb$

- R[14, 15]

16. Mb

- 7. $Mb \vee Ab$
- 10. $Mc \vee \neg Ab$

- 5. $Ma \vee Fb$
- 8. $Mb \vee \neg Fb$

- 3. $\neg Ma \vee \neg Mc$
- 13. $Mb \vee Mc$

- 14. $Ma \vee Mb$
- 15. $\neg Ma \vee Mb$

Problem 4 (b)

- Consider the case where only Arthur is innocent
- To present an interpretation that satisfies this condition, provide the truth values for each propositions:
- $Ma = false,\ Mb = Mc = true$
- $Fb = true, Fc = false$
- $Aa = false, Ab = false$

- Updated KB
1. $Ma \vee Mb \vee Mc$
2. $Ma \vee Fb$
3. $Ma \vee \neg Fc$
4. $Mb \vee Ab$
5. $Mb \vee \neg Fb$
6. $Mc \vee \neg Aa$
7. $Mc \vee \neg Ab$
8. $\neg Aa \vee \neg Ma$
9. $\neg Ab \vee \neg Mb$

- The cases for Bertram and Carleton are similar