AI Assignment

AI 1 - Assignment 06
[Constraint Propagation]

Problem 6.1 ⇒ Scheduling CS Classes as a CSP

01. Since, there will be one var. per class,

Time	Class	Var	Domain
8:30 - 9:30	Class 1	C_1	A
8:00 - 9:00	Class 2	C_2	A, C
9:00 - 10:00	Class 3	C_3	A, B, C
9:30 - 10:30	Class 4	C_4	A, B, C
9:00 - 10:00	Class 5	C_5	A, C

Var ⇒ each classes

Domain ⇒ professors for that class

Constraints ⇒ ${}^5C_2 = \frac{5!}{2!(5-2)!} = \frac{5!}{2!\,3!}$

$= \frac{5 \times 4 \times 3!}{2 \times 3!} = 10$

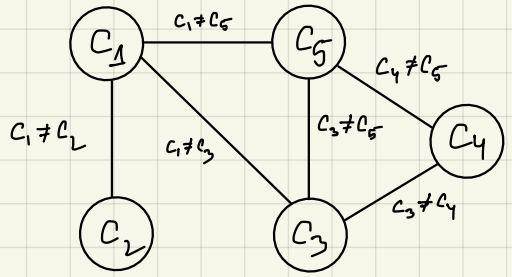
a b c d e

ab bc cd de
ac bd ce
ad be
ae

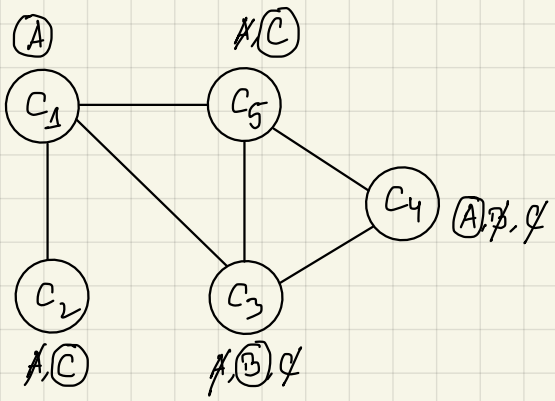
Possible Constraints:

$C_1 \neq C_2$ ✓	$C_2 \neq C_3$ ✗	$C_3 \neq C_4$ ✓	$C_4 \neq C_5$ ✓
$C_1 \neq C_3$ ✓	$C_2 \neq C_4$ ✗	$C_3 \neq C_5$ ✓	
$C_1 \neq C_4$ ✗	$C_2 \neq C_5$ ✗		
$C_1 \neq C_5$ ✓			

02.



03.



Soln:

$C_1 = A$

$C_2 = C$

$C_3 = B$

$C_4 = A$

$C_5 = C$

a total inconsistent assignment:

$C_1 = A$

$C_2 = C$

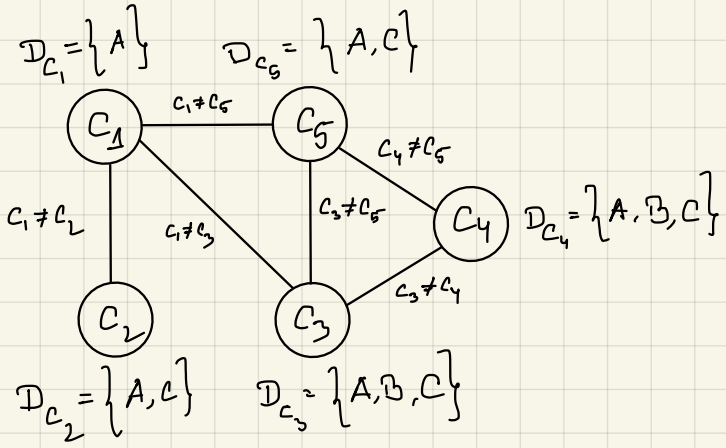
$C_3 = B$

$C_4 = A$

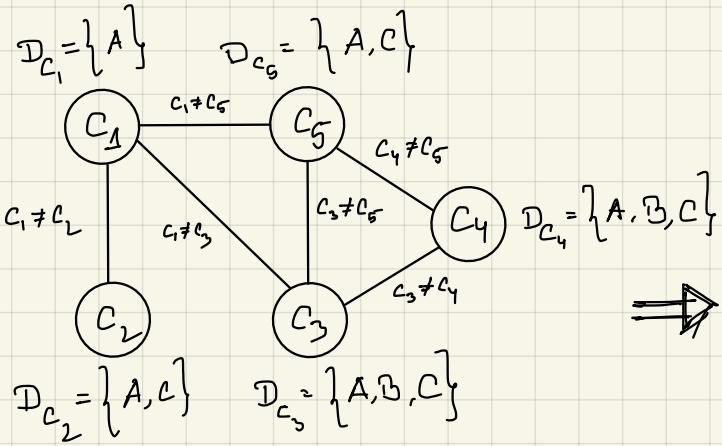
$C_5 = A$ ✗

Problem 6.2 ⇒ Scheduling CS Classes with Constraint Propagation

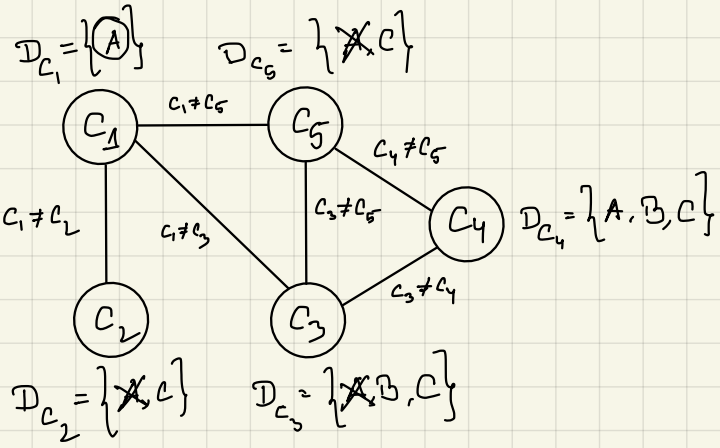
CSP graph:



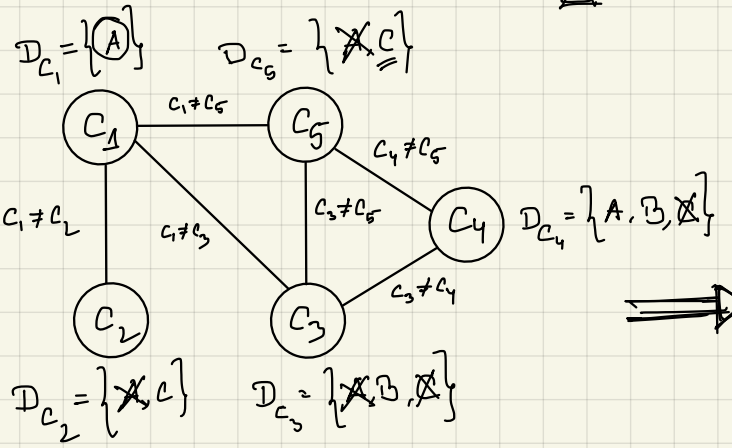
* running arc consistency ⇒



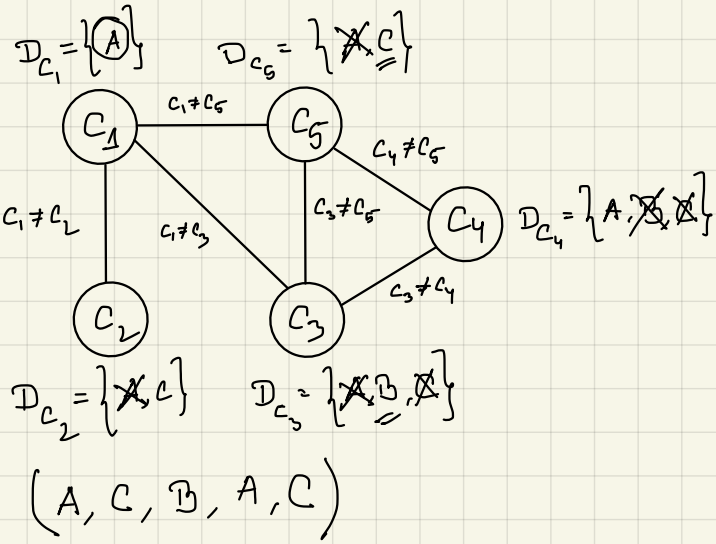
⇒



⇐



⇒



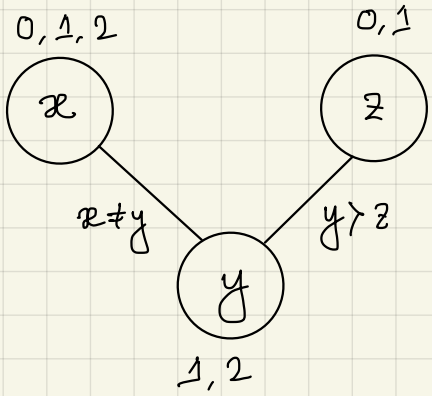
(A, C, B, A, C)

Problem 6.3 ⇒ CSP Formalization

Binary CSP $\Pi := (V, D, C)$

V	D
x	$D_x = \{0, 1, 2\}$
y	$D_y = \{1, 2\}$
z	$D_z = \{0, 1\}$

$C: x \neq y,\ y > z$



* Forward checking: Updating domain
One-step lookahead

heuristics

Choose var. ⟹ MCV ⟹ most constrained var. → less option
[That has fewest consistent values]

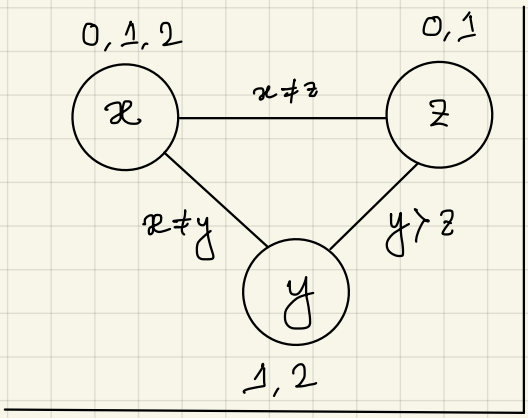
Choose val ⟹ LCV ⟹ least constrained value
[order values of selected X_i (var.) by decreasing number of consistent values of neighboring var.]

AC-3

* Arc consistency: Updating domain (Not just neighbors, rather the whole constraints graph)

$X_i \xrightarrow[\text{Consistent}]{\text{arc}} X_j$: remove values from D_i

01. $(i, w) \Longrightarrow i$ is arc-consistent w.r.t w



$(x, y) \Longrightarrow$

x	y
0	1
1	2
2	

$x \neq y$ arc-consistent

$(y, x) \Longrightarrow$

y	x
1	0
2	1
	2

$y \neq x$ arc-consistent

$(y, z) \Longrightarrow$

y	z
1	0
2	1

$y > z$ arc-consistent

$(z, y) \Longrightarrow$

z	y
0	1
1	2

$z < y$ arc-consistent

$(x, z) \Longrightarrow$

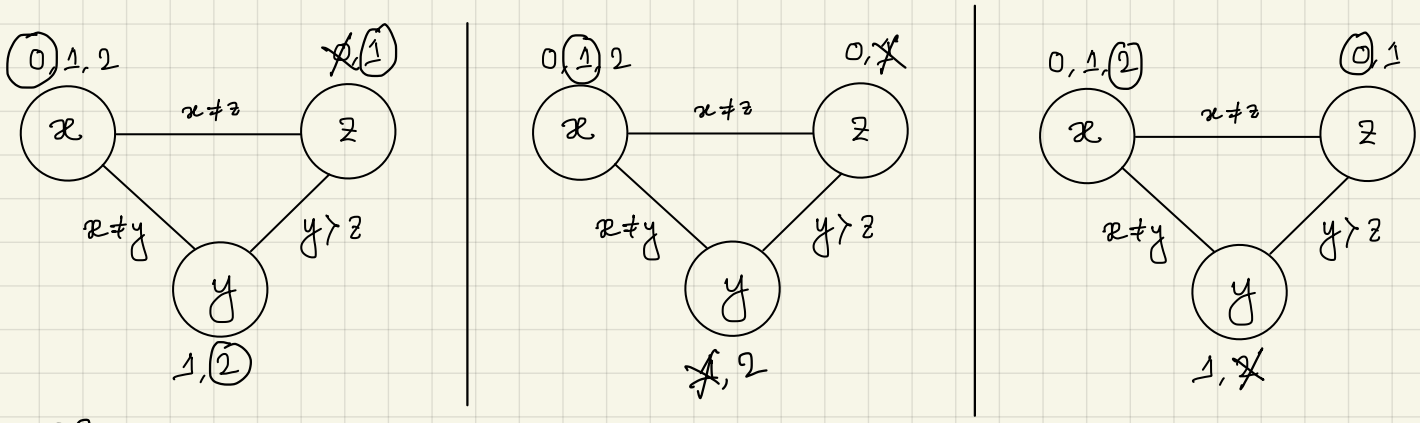
x	z
0	0
1	1
2	

$x \neq z$ arc-consistent

$(z, x) \Longrightarrow$

z	x
0	0
1	1
	2

$z \neq x$ arc-consistent

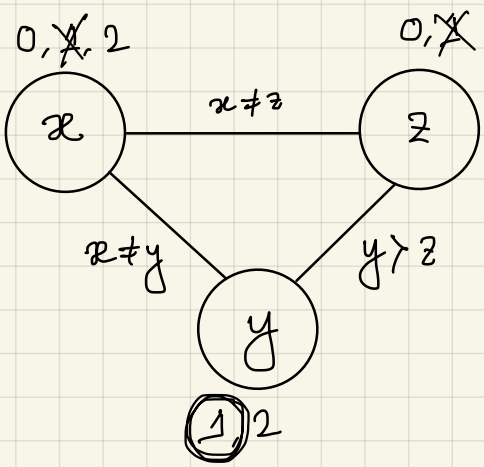


02.

x	y	z	x≠y	x≠z	y>z
0	2	1	✓	✓	✓
1	2	0	✓	✓	✓
2	1	0	✓	✓	✓

$\Rightarrow$ (0, 2, 1)
(1, 2, 0)
(2, 1, 0)

03.



~~Assign~~, $y = 1$, then

$D_x = \{0, 2\}$

$D_y = \{1\}$

$D_z = \{0\}$

AI1 - Assignment 07 (Propositional Logic)

Syntax	Semantics
Valid Express.	Meaning of Express.

Propositional symbol (atomic formulas): A, B, C

Logical connectives: $\neg, \wedge, \vee, \Rightarrow, \Leftrightarrow$

$\neg f,\ f \wedge g,\ f \vee g,\ f \Rightarrow g,\ f \Leftrightarrow g$

* Model: model ω in propositional logic $\Rightarrow$ an assignment of True val. to propositional symbols

A, B, C: $2^3 = 8$ possible ω (T, F)

$\vdash \{(0,0,0), (0,0,1), \dots, (1,1,1)\}$

* Interpretation funct^n: $I(f, \omega)$
  - returning True (1) $\Rightarrow$ if ω satisfies f
  - returning False (0) $\Rightarrow$ if ω does not satisfy f

$I(f, \omega) \in \{0, 1\}$

f	g	$\neg f$	$f \wedge g$	$f \vee g$	$f \Rightarrow g$ (TFF)	$f \Leftrightarrow g$
0	0	1	0	0	1	1
0	1	1	0	1	1	0
1	0	0	0	1	0	0
1	1	0	1	1	1	1

$f \Rightarrow g$: Conditional proposition

* Models: $M(f)$ — set of models ω for which $I(f, \omega) = 1$

Knowledge Base: set of formulas

$$M(KB) = \bigcap_{f \in KB} M(f)$$

* SAT $\Rightarrow$ checking satisfiability in PL

Model checking

propositional symbol — variable
formula — constraints
model — assignment

} CSP

Assignment 11

Problem 11.1

according to problem description,

- concept : student, course, woman
- roles : has_register, taught_by, is_student

ALC conversion :

01. $student \sqcap \exists has\_register.course$
02. $(student \sqcap \forall has\_register.course) \sqsubseteq \forall taught\_by.woman$
03. $\forall has\_register.\top \sqsubseteq \forall is\_student.\top$

ALC formula into FOL :

01. $\forall x \left(student(x) \wedge \exists y \left(has\_register(x,y) \wedge course(y) \right) \right)$

02. $\forall x \left(student(x) \wedge \forall y \left(has\_register(x,y) \rightarrow course(y) \right) \right)$

$\rightarrow \left(\forall z \left(taught\_by(y,z) \rightarrow woman(z) \right) \right)$

03. $\forall x \forall y \; has\_register(x,y) \rightarrow is\_student(x)$

Problem 11.2 (ALC Semantics)

01. ALC concepts $\forall R.(C \sqcap D)$ & $(\forall R.C) \sqcap (\forall R.D)$

$$[[\forall R.(C \sqcap D)]] = \{ x \in D \mid \forall y.\ \text{if } \langle x,y \rangle \in [[R]] \text{ then } y \in [[C \sqcap D]] \}$$

$$= \{ x \in D \mid \forall y.\ \text{if } \langle x,y \rangle \in [[R]] \text{ then } (y \in [[C]] \cap y \in [[D]]) \}$$

$$= \{ x \in D \mid \forall y.\ \text{if } \langle x,y \rangle \in [[R]] \text{ then } y \in [[C]] \} \cap \{ x \in D \mid \forall y.\ \text{if } \langle x,y \rangle \in [[R]] \text{ then } y \in [[D]] \}$$

$$= [[\forall R.C]] \sqcap [[\forall R.D]]$$

02. FOL conversion:

(i) $\forall R.(C \sqcap D) = \forall y\ R(x,y) \rightarrow (C(y) \wedge D(y))$

(ii) $\forall R.C \sqcap \forall R.D = (\forall y\ R(x,y) \rightarrow C(y)) \wedge (\forall z\ R(x,y) \rightarrow D(y))$

$$\forall y\ R(x,y) \rightarrow (C(y) \wedge D(y)) \leftrightarrow (\forall y\ R(x,y) \rightarrow C(y)) \wedge (\forall z\ R(x,y) \rightarrow D(y))$$

Since, we need to show that two formulas are equivalent, then Natural Deduction calculus will be best suited for this case.

Problem 11.3 (ALC TBox)

- primitive concepts : woman, man
- roles : has_child, has_parent, has_sibling, has_spouse

Concepts are define using ALC TBox as follows,

$\text{person} = \text{woman} \sqcup \text{man}$

$\text{parent} = \text{person} \sqcap \exists \text{has\_child}.\text{person}$

$\text{mother} = \text{woman} \sqcap \exists \text{has\_child}.\text{person}$

$\text{father} = \text{man} \sqcap \exists \text{has\_child}.\text{person}$

$\text{grandmother} = \text{woman} \sqcap \exists \text{has\_child}.(\text{mother} \sqcup \text{father})$

$\text{aunt} = \text{woman} \sqcap \exists \text{has\_sibling}.(\text{mother} \sqcup \text{father})$

$\text{uncle} = \text{man} \sqcap \exists \text{has\_sibling}.(\text{mother} \sqcup \text{father})$

$\text{sister} = \text{woman} \sqcap \exists \text{has\_sibling}.\text{person}$

$\text{brother} = \text{man} \sqcap \exists \text{has\_sibling}.\text{person}$

$\text{onlychild} = \text{person} \sqcap \neg \exists \text{has\_sibling}.\text{person}$

$\text{cousin} = \text{person} \sqcap \exists \text{has\_parent}.\exists \text{has\_sibling}.(\text{mother} \sqcup \text{father})$

$\text{nephew} = \text{man} \sqcap \exists \text{has\_parent}.(\text{brother} \sqcup \text{sister})$

$\text{niece} = \text{woman} \sqcap \exists \text{has\_parent}.(\text{brother} \sqcup \text{sister})$

$\text{fatherinlaw} = \text{man} \sqcap \exists \text{has\_child}.\exists \text{has\_spouse}.\text{person}$

$\text{motherinlaw} = \text{woman} \sqcap \exists \text{has\_child}.\exists \text{has\_spouse}.\text{person}$