Artificial Intelligence

SCALAB
Grupo de Inteligencia Artificial

Universidad Carlos III de Madrid

2017-2018

Production Systems – Exercises

Exercise 1

In a production system we have introduced the following rules:

- R1: IF a(X) and b(Y) THEN c(Y)
- R2: IF a(X) and c(X) THEN d(X)

The database contains the following facts:

a(manuel), b(manuel), b(john), c(alberto)

1. Which INSTANCES of what rule are activated in the first cycle of execution?
2. If the system operates under a first rule (FIFO) or last rule (LIFO) conflict resolution strategy (CS), considering the order of rules and facts, show the sequence of rules carried out and the data on the Working Memory (WM) for each cycle

Recap: Production Systems

- Architecture
  - Production Systems have 3 parts:

$$Data = \begin{cases} -Rule\ Set(or\ productions): \ knowledge\ in\ form\ of\ a\ set\ of\ production \\ rules. \\ -Fact\ Set(or\ working\ memory): \ representation\ of\ the\ current\ state \\ or\ the\ problem's\ context. \end{cases}$$

$$Program = \begin{cases} -Rule\ Interpreter: \ acts\ on\ the\ two\ previous\ components.\ It\ decides \\ which\ rule\ to\ apply\ in\ each\ case\ depending\ on\ the\ WM. \end{cases}$$

- Working of operation cycles
  - In each cycle there are 3 Phases
    - Matching: What rules are applicable? (activation, conflict set, agenda)
    - Selection or conflict resolution: What rule triggers? (Depending on resolution strategy (FIFO,LIFO,etc) and refraction(rules cannot be used again))
    - Execution: Corresponding action is performed (there is a change in WM)

Solution Exercise1 – FIFO

- Rules (Rule Set)
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(X) THEN d(X)

$WM_0 = \{$ a(manuel), b(manuel), b(john), c(alberto) $\}$

Solution Exercise1 – FIFO

- Rules (Rule Set)
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(X) THEN d(X)

$WM_0 =$ { a(manuel), b(manuel), b(john), c(alberto) }
$CS_0 =$ { R1(X=manuel, Y=manuel), R1(X=manuel, Y=john) }

Note that although x and y are different, we can apply R1 in two ways: y=manuel and y=john. We could not do this in the case of R2, as there we have the limitation x,x.

Solution Exercise1 – FIFO

- Rules (Rule Set)
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(X) THEN d(X)

WM_0 = { a(manuel), b(manuel), b(john), c(alberto) }
CS_0 = { R1(X=manuel, Y=manuel), R1(X=manuel, Y=john) }

Solution Exercise1 – FIFO

- Rules (Rule Set)
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(X) THEN d(X)

WM_0 = { a(manuel), b(manuel), b(john), c(alberto) }
CS_0 = { R1(X=manuel, Y=manuel), R1(X=manuel, Y=john) }

WM_1 = { a(manuel), b(manuel), b(john), c(alberto), c(manuel) }

Solution Exercise1 – FIFO

- Rules (Rule Set)
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(X) THEN d(X)

WM_0 = { a(manuel), b(manuel), b(john), c(alberto) }
CS_0 = { R1(X=manuel, Y=manuel), R1(X=manuel, Y=john) }

WM_1 = { a(manuel), b(manuel), b(john), c(alberto), c(manuel) }
CS_1 = { R1(X=manuel, Y=manuel), R1(X=manuel, Y=john), R2(X=manuel)}

Solution Exercise1 – FIFO

- Rules (Rule Set)
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(X) THEN d(X)

WM_0 = { a(manuel), b(manuel), b(john), c(alberto) }
CS_0 = { R1(X=manuel, Y=manuel), R1(X=manuel, Y=john) }

WM_1 = { a(manuel), b(manuel), b(john), c(alberto), c(manuel) }
CS_1 = { R1(X=manuel, Y=manuel), R1(X=manuel, Y=john), R2(X=manuel)}

Solution Exercise1 – FIFO

- Rules (Rule Set)
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(X) THEN d(X)

WM_0 = { a(manuel), b(manuel), b(john), c(alberto) }
CS_0 = { R1(X=manuel, Y=manuel), R1(X=manuel, Y=john) }

WM_1 = { a(manuel), b(manuel), b(john), c(alberto), c(manuel) }
CS_1 = { R1(X=manuel, Y=manuel), R1(X=manuel, Y=john), R2(X=manuel)}

WM_2 = { a(manuel), b(manuel), b(john), c(alberto), c(manuel), c(john) }

Solution Exercise1 – FIFO

- Rules (Rule Set)
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(X) THEN d(X)

WM_0 = { a(manuel), b(manuel), b(john), c(alberto) }
CS_0 = { R1(X=manuel, Y=manuel), R1(X=manuel, Y=john) }

WM_1 = { a(manuel), b(manuel), b(john), c(alberto), c(manuel) }
CS_1 = { R1(X=manuel, Y=manuel), R1(X=manuel, Y=john), R2(X=manuel)}

WM_2 = { a(manuel), b(manuel), b(john), c(alberto), c(manuel), c(john) }
CS_2 = { R1(X=manuel, Y=manuel), R1(X=manuel,Y=john), R2(X=manuel)}

Solution Exercise1 – FIFO

- Rules (Rule Set)
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(X) THEN d(X)

WM_0 = { a(manuel), b(manuel), b(john), c(alberto) }
CS_0 = { R1(X=manuel, Y=manuel), R1(X=manuel, Y=john) }

WM_1 = { a(manuel), b(manuel), b(john), c(alberto), c(manuel) }
CS_1 = { R1(X=manuel, Y=manuel), R1(X=manuel, Y=john), R2(X=manuel)}

WM_2 = { a(manuel), b(manuel), b(john), c(alberto), c(manuel), c(john) }
CS_2 = { R1(X=manuel, Y=manuel), R1(X=manuel,Y=john), R2(X=manuel)}

Solution Exercise1 – FIFO

- Rules (Rule Set)
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(X) THEN d(X)

WM_0 = { a(manuel), b(manuel), b(john), c(alberto) }
CS_0 = { R1(X=manuel, Y=manuel), R1(X=manuel, Y=john) }

WM_1 = { a(manuel), b(manuel), b(john), c(alberto), c(manuel) }
CS_1 = { R1(X=manuel, Y=manuel), R1(X=manuel, Y=john), R2(X=manuel)}

WM_2 = { a(manuel), b(manuel), b(john), c(alberto), c(manuel), c(john) }
CS_2 = { R1(X=manuel, Y=manuel), R1(X=manuel,Y=john), R2(X=manuel)}

WM_3 = { a(manuel), b(manuel), b(john), c(alberto),c(manuel), c(john), d(manuel) }

Solution Exercise1 – FIFO

- Rules (Rule Set)
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(X) THEN d(X)

$WM_0 = \{$ a(manuel), b(manuel), b(john), c(alberto) $\}$
$CS_0 = \{$ R1(X=manuel, Y=manuel), R1(X=manuel, Y=john) $\}$

$WM_1 = \{$ a(manuel), b(manuel), b(john), c(alberto), c(manuel) $\}$
$CS_1 = \{$ R1(X=manuel, Y=manuel), R1(X=manuel, Y=john), R2(X=manuel)$\}$

$WM_2 = \{$ a(manuel), b(manuel), b(john), c(alberto), c(manuel), c(john) $\}$
$CS_2 = \{$ R1(X=manuel, Y=manuel), R1(X=manuel,Y=john), R2(X=manuel)$\}$

$WM_3 = \{$ a(manuel), b(manuel), b(john), c(alberto),c(manuel), c(john), d(manuel) $\}$
$CS_3 = \{$ R1(X=manuel, Y=manuel), R1(X=manuel,Y=john), R2(X=manuel)$\} \rightarrow CS_3 = \emptyset$

Solution Exercise1 – FIFO

- Rules (Rule Set)
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(X) THEN d(X)

WM_0 = { a(manuel), b(manuel), b(john), c(alberto) }
CS_0 = { R1(X=manuel, Y=manuel), R1(X=manuel, Y=john) }

WM_1 = { a(manuel), b(manuel), b(john), c(alberto), c(manuel) }
CS_1 = { R1(X=manuel, Y=manuel), R1(X=manuel, Y=john), R2(X=manuel)}

WM_2 = { a(manuel), b(manuel), b(john), c(alberto), c(manuel), c(john) }
CS_2 = { R1(X=manuel, Y=manuel), R1(X=manuel,Y=john), R2(X=manuel)}

WM_3 = { a(manuel), b(manuel), b(john), c(alberto),c(manuel), c(john), d(manuel) }
CS_3 = { R1(X=manuel, Y=manuel), R1(X=manuel,Y=john), R2(X=manuel)} $\rightarrow CS_3 = \emptyset$

Solution Exercise1 – LIFO

- Rules
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(X) THEN d(X)

$WM_0 = \{$ a(manuel), b(manuel), b(john), c(alberto) $\}$

Solution Exercise1 – LIFO

- Rules
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(X) THEN d(X)

$WM_0 =$ { a(manuel), b(manuel), b(john), c(alberto) }
$CS_0 =$ { R1(X=manuel, Y=manuel), R1(X=manuel, Y=john) }

Solution Exercise1 – LIFO

- Rules
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(X) THEN d(X)

WM_0 = { a(manuel), b(manuel), b(john), c(alberto) }
CS_0 = { R1(X=manuel, Y=manuel), R1(X=manuel, Y=john) }

Solution Exercise1 – LIFO

- Rules
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(X) THEN d(X)

WM_0 = { a(manuel), b(manuel), b(john), c(alberto) }
CS_0 = { R1(X=manuel, Y=manuel), R1(X=manuel, Y=john) }

WM_1 = { a(manuel), b(manuel), b(john), c(alberto), c(john) }

Solution Exercise1 – LIFO

- Rules
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(X) THEN d(X)

WM_0 = { a(manuel), b(manuel), b(john), c(alberto) }
CS_0 = { R1(X=manuel, Y=manuel), R1(X=manuel, Y=john) }

WM_1 = { a(manuel), b(manuel), b(john), c(alberto), c(john) }
CS_1 = { R1(X=manuel, Y=manuel), R1(X=manuel, Y=john)}

Solution Exercise1 – LIFO

- Rules
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(X) THEN d(X)

WM_0 = { a(manuel), b(manuel), b(john), c(alberto) }
CS_0 = { R1(X=manuel, Y=manuel), R1(X=manuel, Y=john) }

WM_1 = { a(manuel), b(manuel), b(john), c(alberto), c(john) }
CS_1 = { R1(X=manuel, Y=manuel), R1(X=manuel, Y=john)}

Solution Exercise1 – LIFO

- Rules
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(X) THEN d(X)

WM_0 = { a(manuel), b(manuel), b(john), c(alberto) }
CS_0 = { R1(X=manuel, Y=manuel), R1(X=manuel, Y=john) }

WM_1 = { a(manuel), b(manuel), b(john), c(alberto), c(john) }
CS_1 = { R1(X=manuel, Y=manuel), R1(X=manuel, Y=john)}

WM_2 = { a(manuel), b(manuel), b(john), c(alberto), c(john), c(manuel) }

Solution Exercise1 – LIFO

- Rules
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(X) THEN d(X)

WM_0 = { a(manuel), b(manuel), b(john), c(alberto) }
CS_0 = { R1(X=manuel, Y=manuel), R1(X=manuel, Y=john) }

WM_1 = { a(manuel), b(manuel), b(john), c(alberto), c(john) }
CS_1 = { R1(X=manuel, Y=manuel), R1(X=manuel, Y=john)}

WM_2 = { a(manuel), b(manuel), b(john), c(alberto), c(john), c(manuel) }
CS_2 = { R1(X=manuel, Y=manuel), R1(X=manuel,Y=john), R2(X=manuel)}

Solution Exercise1 – LIFO

- Rules
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(X) THEN d(X)

WM_0 = { a(manuel), b(manuel), b(john), c(alberto) }
CS_0 = { R1(X=manuel, Y=manuel), R1(X=manuel, Y=john) }

WM_1 = { a(manuel), b(manuel), b(john), c(alberto), c(john) }
CS_1 = { R1(X=manuel, Y=manuel), R1(X=manuel, Y=john)}

WM_2 = { a(manuel), b(manuel), b(john), c(alberto), c(john), c(manuel) }
CS_2 = { R1(X=manuel, Y=manuel), R1(X=manuel,Y=john), R2(X=manuel)}

Solution Exercise1 – LIFO

- Rules
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(X) THEN d(X)

WM_0 = { a(manuel), b(manuel), b(john), c(alberto) }
CS_0 = { R1(X=manuel, Y=manuel), R1(X=manuel, Y=john) }

WM_1 = { a(manuel), b(manuel), b(john), c(alberto), c(john) }
CS_1 = { R1(X=manuel, Y=manuel), R1(X=manuel, Y=john)}

WM_2 = { a(manuel), b(manuel), b(john), c(alberto), c(john), c(manuel) }
CS_2 = { R1(X=manuel, Y=manuel), R1(X=manuel,Y=john), R2(X=manuel)}

WM_3 = { a(manuel), b(manuel), b(john), c(alberto),c(manuel), c(john), d(manuel) }

Solution Exercise1 – LIFO

- Rules
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(X) THEN d(X)

WM_0 = { a(manuel), b(manuel), b(john), c(alberto) }
CS_0 = { R1(X=manuel, Y=manuel), R1(X=manuel, Y=john) }

WM_1 = { a(manuel), b(manuel), b(john), c(alberto), c(john) }
CS_1 = { R1(X=manuel, Y=manuel), R1(X=manuel, Y=john)}

WM_2 = { a(manuel), b(manuel), b(john), c(alberto), c(john), c(manuel) }
CS_2 = { R1(X=manuel, Y=manuel), R1(X=manuel,Y=john), R2(X=manuel)}

WM_3 = { a(manuel), b(manuel), b(john), c(alberto),c(manuel), c(john), d(manuel) }
CS_3 = { R1(X=manuel, Y=manuel), R1(X=manuel,Y=john), R2(X=manuel)} $\rightarrow CS_3 = \emptyset$

Solution Exercise1 – LIFO

- Rules
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(X) THEN d(X)

WM_0 = { a(manuel), b(manuel), b(john), c(alberto) }
CS_0 = { R1(X=manuel, Y=manuel), R1(X=manuel, Y=john) }

WM_1 = { a(manuel), b(manuel), b(john), c(alberto), c(john) }
CS_1 = { R1(X=manuel, Y=manuel), R1(X=manuel, Y=john)}

WM_2 = { a(manuel), b(manuel), b(john), c(alberto), c(john), c(manuel) }
CS_2 = { R1(X=manuel, Y=manuel), R1(X=manuel,Y=john), R2(X=manuel)}

WM_3 = { a(manuel), b(manuel), b(john), c(alberto),c(manuel), c(john), d(manuel) }
CS_3 = { R1(X=manuel, Y=manuel), R1(X=manuel,Y=john), R2(X=manuel)} $\rightarrow CS_3 = \emptyset$

Exercise 2

In a production system we have introduced the following rules:

- R1: IF a(X) and b(Y) THEN c(Y)
- R2: IF a(X) and c(Y) THEN d(Y)
- R3: IF a(X) and c(X) THEN e(X)

The database contains the following facts:

a(manuel), b(john), c(manuel)

If the system operates under a first rule (FIFO) or last rule (LIFO) conflict resolution strategy (CS), considering the order of rules and facts, show the sequence of rules carried out and the data on the Working Memory (WM) for each cycle

Solution Exercise2 – FIFO

- Rules
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(Y) THEN d(Y)
  - R3: IF a(X) and c(X) THEN e(X)

WM_0 = { a(manuel), b(jonh), c(manuel) }

Solution Exercise2 – FIFO

- Rules
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(Y) THEN d(Y)
  - R3: IF a(X) and c(X) THEN e(X)

WM_0 = { a(manuel), b(jonh), c(manuel) }
CS_0 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel) }

Solution Exercise2 – FIFO

- Rules
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(Y) THEN d(Y)
  - R3: IF a(X) and c(X) THEN e(X)

WM_0 = { a(manuel), b(jonh), c(manuel) }
CS_0 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel) }

Solution Exercise2 – FIFO

- Rules
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(Y) THEN d(Y)
  - R3: IF a(X) and c(X) THEN e(X)

WM_0 = { a(manuel), b(jonh), c(manuel) }
CS_0 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel) }

WM_1 = { a(manuel), b(john), c(manuel), c(jonh) }

Solution Exercise2 – FIFO

- Rules
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(Y) THEN d(Y)
  - R3: IF a(X) and c(X) THEN e(X)

WM_0 = { a(manuel), b(jonh), c(manuel) }
CS_0 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel) }

WM_1 = { a(manuel), b(john), c(manuel), c(jonh) }
CS_1 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel), R2(X=manuel, Y=jonh)}

Solution Exercise2 – FIFO

- Rules
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(Y) THEN d(Y)
  - R3: IF a(X) and c(X) THEN e(X)

WM_0 = { a(manuel), b(jonh), c(manuel) }
CS_0 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel) }

WM_1 = { a(manuel), b(john), c(manuel), c(jonh) }
CS_1 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel), R2(X=manuel, Y=jonh)}

Solution Exercise2 – FIFO

- Rules
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(Y) THEN d(Y)
  - R3: IF a(X) and c(X) THEN e(X)

WM_0 = { a(manuel), b(jonh), c(manuel) }
CS_0 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel) }

WM_1 = { a(manuel), b(john), c(manuel), c(jonh) }
CS_1 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel), R2(X=manuel, Y=jonh)}

WM_2 = { a(manuel), b(john), c(manuel), c(jonh), d(manuel) }

Solution Exercise2 – FIFO

- Rules
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(Y) THEN d(Y)
  - R3: IF a(X) and c(X) THEN e(X)

WM_0 = { a(manuel), b(jonh), c(manuel) }
CS_0 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel) }

WM_1 = { a(manuel), b(john), c(manuel), c(jonh) }
CS_1 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel), R2(X=manuel, Y=jonh)}

WM_2 = { a(manuel), b(john), c(manuel), c(jonh), d(manuel) }
CS_2 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel), R2(X=manuel, Y=jonh)}

Solution Exercise2 – FIFO

- Rules
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(Y) THEN d(Y)
  - R3: IF a(X) and c(X) THEN e(X)

WM_0 = { a(manuel), b(jonh), c(manuel) }
CS_0 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel) }

WM_1 = { a(manuel), b(john), c(manuel), c(jonh) }
CS_1 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel), R2(X=manuel, Y=jonh)}

WM_2 = { a(manuel), b(john), c(manuel), c(jonh), d(manuel) }
CS_2 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel), R2(X=manuel, Y=jonh)}

Solution Exercise2 – FIFO

- Rules
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(Y) THEN d(Y)
  - R3: IF a(X) and c(X) THEN e(X)

WM_0 = { a(manuel), b(jonh), c(manuel) }
CS_0 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel) }

WM_1 = { a(manuel), b(john), c(manuel), c(jonh) }
CS_1 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel), R2(X=manuel, Y=jonh)}

WM_2 = { a(manuel), b(john), c(manuel), c(jonh), d(manuel) }
CS_2 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel), R2(X=manuel, Y=jonh)}

WM_3 = { a(manuel), b(john), c(manuel), c(john), d(manuel), e(manuel) }

Solution Exercise2 – FIFO

- Rules
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(Y) THEN d(Y)
  - R3: IF a(X) and c(X) THEN e(X)

WM_0 = { a(manuel), b(jonh), c(manuel) }
CS_0 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel) }

WM_1 = { a(manuel), b(john), c(manuel), c(jonh) }
CS_1 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel), R2(X=manuel, Y=jonh)}

WM_2 = { a(manuel), b(john), c(manuel), c(jonh), d(manuel) }
CS_2 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel), R2(X=manuel, Y=jonh)}

WM_3 = { a(manuel), b(john), c(manuel), c(jonh), d(manuel), e(manuel) }
CS_3 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel), R2(X=manuel, Y=jonh)}

Solution Exercise2 – FIFO

- Rules
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(Y) THEN d(Y)
  - R3: IF a(X) and c(X) THEN e(X)

WM_0 = { a(manuel), b(jonh), c(manuel) }
CS_0 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel) }

WM_1 = { a(manuel), b(john), c(manuel), c(jonh) }
CS_1 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel), R2(X=manuel, Y=jonh)}

WM_2 = { a(manuel), b(john), c(manuel), c(jonh), d(manuel) }
CS_2 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel), R2(X=manuel, Y=jonh)}

WM_3 = { a(manuel), b(john), c(manuel), c(john), d(manuel), e(manuel) }
CS_3 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel), R2(X=manuel, Y=jonh)}

Solution Exercise2 – FIFO

- Rules
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(Y) THEN d(Y)
  - R3: IF a(X) and c(X) THEN e(X)

WM_0 = { a(manuel), b(jonh), c(manuel) }
CS_0 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel) }

WM_1 = { a(manuel), b(john), c(manuel), c(jonh) }
CS_1 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel), R2(X=manuel, Y=jonh)}

WM_2 = { a(manuel), b(john), c(manuel), c(jonh), d(manuel) }
CS_2 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel), R2(X=manuel, Y=jonh)}

WM_3 = { a(manuel), b(john), c(manuel), c(john), d(manuel), e(manuel) }
CS_3 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel), R2(X=manuel, Y=jonh)}

WM_4 = { a(manuel), b(john), c(manuel), c(john), d(manuel), e(manuel), d(john) }

Solution Exercise2 – FIFO

- Rules
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(Y) THEN d(Y)
  - R3: IF a(X) and c(X) THEN e(X)

WM_0 = { a(manuel), b(jonh), c(manuel) }
CS_0 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel) }

WM_1 = { a(manuel), b(john), c(manuel), c(jonh) }
CS_1 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel), R2(X=manuel, Y=jonh)}

WM_2 = { a(manuel), b(john), c(manuel), c(jonh), d(manuel) }
CS_2 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel), R2(X=manuel, Y=jonh)}

WM_3 = { a(manuel), b(john), c(manuel), c(john), d(manuel), e(manuel) }
CS_3 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel), R2(X=manuel, Y=jonh)}

WM_4 = { a(manuel), b(john), c(manuel), c(john), d(manuel), e(manuel), d(john) }
CS_4 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel), R2(X=manuel, Y=jonh)} $\rightarrow CS_4 = \emptyset$

Solution Exercise2 – LIFO

- Rules
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(Y) THEN d(Y)
  - R3: IF a(X) and c(X) THEN e(X)

WM_0 = { a(manuel), b(jonh), c(manuel) }

Solution Exercise2 – LIFO

- Rules
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(Y) THEN d(Y)
  - R3: IF a(X) and c(X) THEN e(X)

WM_0 = { a(manuel), b(jonh), c(manuel) }
CS_0 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel) }

Solution Exercise2 – LIFO

- Rules
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(Y) THEN d(Y)
  - R3: IF a(X) and c(X) THEN e(X)

WM_0 = { a(manuel), b(jonh), c(manuel) }
CS_0 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel) }

Solution Exercise2 – LIFO

- Rules
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(Y) THEN d(Y)
  - R3: IF a(X) and c(X) THEN e(X)

WM_0 = { a(manuel), b(jonh), c(manuel) }
CS_0 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel) }

WM_1 = { a(manuel), b(john), c(manuel), e(manuel) }

Solution Exercise2 – LIFO

- Rules
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(Y) THEN d(Y)
  - R3: IF a(X) and c(X) THEN e(X)

WM_0 = { a(manuel), b(jonh), c(manuel) }
CS_0 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel) }

WM_1 = { a(manuel), b(john), c(manuel), e(manuel) }
CS_1 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel)}

Solution Exercise2 – LIFO

- Rules
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(Y) THEN d(Y)
  - R3: IF a(X) and c(X) THEN e(X)

WM_0 = { a(manuel), b(jonh), c(manuel) }
CS_0 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel) }

WM_1 = { a(manuel), b(john), c(manuel), e(manuel) }
CS_1 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel)}

Solution Exercise2 – LIFO

- Rules
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(Y) THEN d(Y)
  - R3: IF a(X) and c(X) THEN e(X)

WM_0 = { a(manuel), b(jonh), c(manuel) }
CS_0 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel) }

WM_1 = { a(manuel), b(john), c(manuel), e(manuel) }
CS_1 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel)}

WM_2 = { a(manuel), b(john), c(manuel), e(manuel), d(manuel) }

Solution Exercise2 – LIFO

- Rules
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(Y) THEN d(Y)
  - R3: IF a(X) and c(X) THEN e(X)

WM_0 = { a(manuel), b(jonh), c(manuel) }
CS_0 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel) }

WM_1 = { a(manuel), b(john), c(manuel), e(manuel) }
CS_1 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel)}

WM_2 = { a(manuel), b(john), c(manuel), e(manuel), d(manuel) }
CS_2 = { R1(X=manuel, Y=jonh), R2(X=manuel,Y=manuel), R3(X=manuel)}

Solution Exercise2 – LIFO

- Rules
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(Y) THEN d(Y)
  - R3: IF a(X) and c(X) THEN e(X)

WM_0 = { a(manuel), b(jonh), c(manuel) }
CS_0 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel) }

WM_1 = { a(manuel), b(john), c(manuel), e(manuel) }
CS_1 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel)}

WM_2 = { a(manuel), b(john), c(manuel), e(manuel), d(manuel) }
CS_2 = { R1(X=manuel, Y=jonh), R2(X=manuel,Y=manuel), R3(X=manuel)}

Solution Exercise2 – LIFO

- Rules
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(Y) THEN d(Y)
  - R3: IF a(X) and c(X) THEN e(X)

WM_0 = { a(manuel), b(jonh), c(manuel) }
CS_0 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel) }

WM_1 = { a(manuel), b(john), c(manuel), e(manuel) }
CS_1 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel)}

WM_2 = { a(manuel), b(john), c(manuel), e(manuel), d(manuel) }
CS_2 = { R1(X=manuel, Y=jonh), R2(X=manuel,Y=manuel), R3(X=manuel)}

WM_3 = { a(manuel), b(john), c(manuel), e(manuel), d(manuel), c(john) }

Solution Exercise2 – LIFO

- Rules
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(Y) THEN d(Y)
  - R3: IF a(X) and c(X) THEN e(X)

WM_0 = { a(manuel), b(jonh), c(manuel) }
CS_0 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel) }

WM_1 = { a(manuel), b(john), c(manuel), e(manuel) }
CS_1 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel)}

WM_2 = { a(manuel), b(john), c(manuel), e(manuel), d(manuel) }
CS_2 = { R1(X=manuel, Y=jonh), R2(X=manuel,Y=manuel), R3(X=manuel)}

WM_3 = { a(manuel), b(john), c(manuel), e(manuel), d(manuel), c(john) }
CS_3 = { R1(X=manuel, Y=jonh), R2(X=manuel,Y=manuel), R3(X=manuel), R2(X=manuel, Y=john)}

Solution Exercise2 – LIFO

- Rules
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(Y) THEN d(Y)
  - R3: IF a(X) and c(X) THEN e(X)

WM_0 = { a(manuel), b(jonh), c(manuel) }
CS_0 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel) }

WM_1 = { a(manuel), b(john), c(manuel), e(manuel) }
CS_1 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel)}

WM_2 = { a(manuel), b(john), c(manuel), e(manuel), d(manuel) }
CS_2 = { R1(X=manuel, Y=jonh), R2(X=manuel,Y=manuel), R3(X=manuel)}

WM_3 = { a(manuel), b(john), c(manuel), e(manuel), d(manuel), c(john) }
CS_3 = { R1(X=manuel, Y=jonh), R2(X=manuel,Y=manuel), R3(X=manuel), R2(X=manuel, Y=john)}

Solution Exercise2 – LIFO

- Rules
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(Y) THEN d(Y)
  - R3: IF a(X) and c(X) THEN e(X)

WM_0 = { a(manuel), b(jonh), c(manuel) }
CS_0 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel) }

WM_1 = { a(manuel), b(john), c(manuel), e(manuel) }
CS_1 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel)}

WM_2 = { a(manuel), b(john), c(manuel), e(manuel), d(manuel) }
CS_2 = { R1(X=manuel, Y=jonh), R2(X=manuel,Y=manuel), R3(X=manuel)}

WM_3 = { a(manuel), b(john), c(manuel), e(manuel), d(manuel), c(john) }
CS_3 = { R1(X=manuel, Y=jonh), R2(X=manuel,Y=manuel), R3(X=manuel), R2(X=manuel, Y=john)}

WM_4 = { a(manuel), b(john), c(manuel), e(manuel), d(manuel), c(john), d(john) }

Solution Exercise2 – LIFO

- Rules
  - R1: IF a(X) and b(Y) THEN c(Y)
  - R2: IF a(X) and c(Y) THEN d(Y)
  - R3: IF a(X) and c(X) THEN e(X)

WM_0 = { a(manuel), b(jonh), c(manuel) }
CS_0 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel) }

WM_1 = { a(manuel), b(john), c(manuel), e(manuel) }
CS_1 = { R1(X=manuel, Y=jonh), R2(X=manuel, Y=manuel), R3(X=manuel)}

WM_2 = { a(manuel), b(john), c(manuel), e(manuel), d(manuel) }
CS_2 = { R1(X=manuel, Y=jonh), R2(X=manuel,Y=manuel), R3(X=manuel)}

WM_3 = { a(manuel), b(john), c(manuel), e(manuel), d(manuel), c(john) }
CS_3 = { R1(X=manuel, Y=jonh), R2(X=manuel,Y=manuel), R3(X=manuel), R2(X=manuel, Y=john)}

WM_4 = { a(manuel), b(john), c(manuel), e(manuel), d(manuel), c(john), d(john) }
CS_4 = { R1(X=manuel, Y=jonh), R2(X=manuel,Y=manuel), R3(X=manuel), R2(X=manuel, Y=john)} $\rightarrow CS_4 = \emptyset$
Note we reach the same solution that with CS=FIFO but the ordering was different.

Exercise 3

In a production system we have the following rules:

- R1: $A \land B \to C$
- R2: $A \to D$
- R3: $C \land D \to E$
- R4: $B \land E \land F \to G$
- R5: $A \land E \to H$
- R6: $D \land E \land H \to I$

The WM contains: A,B,F

How can H be deduced using the following methods:

1. Forward chaining
2. Backward chaining

Recap: Forward and Backward chaining

We study a progression: from a set of initial data to a solution/answer/conclusion.

There are two alternatives:

- Very few initial data and/or a lot of possible conclusions $=>$ Then it is reasonable to progress from initial data to a solution.
  - Reasoning directed by the data (premises) $=>$ see left part of the rules.
  - Chaining of rules forwards: progressive chaining.
- A lot of initial data, but only a few are relevant.
  - Reasoning directed by the goals (results) $=>$ see right part of the rules.
  - Chaining of rules backwards: regressive chaining.

Solution Exercise3 – Forward chaining

- Rules:
  - R1: A $\wedge$ B $\rightarrow$ C
  - R2: A $\rightarrow$ D
  - R3: C $\wedge$ D $\rightarrow$ E
  - R4: B $\wedge$ E $\wedge$ F $\rightarrow$ G
  - R5: A $\wedge$ E $\rightarrow$ H
  - R6: D $\wedge$ E $\wedge$ H $\rightarrow$ I

$WM_0 = \{$ A, B, F $\}$

Solution Exercise3 – Forward chaining

- Rules:
  - R1: A $\wedge$ B $\rightarrow$ C
  - R2: A $\rightarrow$ D
  - R3: C $\wedge$ D $\rightarrow$ E
  - R4: B $\wedge$ E $\wedge$ F $\rightarrow$ G
  - R5: A $\wedge$ E $\rightarrow$ H
  - R6: D $\wedge$ E $\wedge$ H $\rightarrow$ I

$WM_0 = \{$ A, B, F $\}$ $CS_0 = \{$ R1, R2 $\}$

Solution Exercise3 – Forward chaining

- Rules:
  - R1: $A \wedge B \rightarrow C$
  - R2: $A \rightarrow D$
  - R3: $C \wedge D \rightarrow E$
  - R4: $B \wedge E \wedge F \rightarrow G$
  - R5: $A \wedge E \rightarrow H$
  - R6: $D \wedge E \wedge H \rightarrow I$

$WM_0 = \{ A, B, F \}$ $\qquad$ $CS_0 = \{ R1, R2 \}$

$WM_1 = \{ A, B, F, C \}$

Solution Exercise3 – Forward chaining

- Rules:
  - R1: $A \land B \rightarrow C$
  - R2: $A \rightarrow D$
  - R3: $C \land D \rightarrow E$
  - R4: $B \land E \land F \rightarrow G$
  - R5: $A \land E \rightarrow H$
  - R6: $D \land E \land H \rightarrow I$

$WM_0 = \{ A, B, F \}$ $\quad CS_0 = \{ R1, R2 \}$

$WM_1 = \{ A, B, F, C \}$ $\quad CS_1 = \{ R2 \}$

Solution Exercise3 – Forward chaining

- Rules:
  - R1: $A \wedge B \rightarrow C$
  - R2: $A \rightarrow D$
  - R3: $C \wedge D \rightarrow E$
  - R4: $B \wedge E \wedge F \rightarrow G$
  - R5: $A \wedge E \rightarrow H$
  - R6: $D \wedge E \wedge H \rightarrow I$

$WM_0 = \{ \text{A, B, F} \}$ $\quad CS_0 = \{ \text{R1, R2} \}$

$WM_1 = \{ \text{A, B, F, C} \}$ $\quad CS_1 = \{ \text{R2} \}$

$WM_2 = \{ \text{A, B, F, C, D} \}$

Solution Exercise3 – Forward chaining

- Rules:
  - R1: $A \land B \rightarrow C$
  - R2: $A \rightarrow D$
  - R3: $C \land D \rightarrow E$
  - R4: $B \land E \land F \rightarrow G$
  - R5: $A \land E \rightarrow H$
  - R6: $D \land E \land H \rightarrow I$

$WM_0 = \{ A, B, F \}$	$CS_0 = \{ R1, R2 \}$
$WM_1 = \{ A, B, F, C \}$	$CS_1 = \{ R2 \}$
$WM_2 = \{ A, B, F, C, D \}$	$CS_2 = \{ R3 \}$

Solution Exercise3 – Forward chaining

- Rules:
  - R1: $A \wedge B \rightarrow C$
  - R2: $A \rightarrow D$
  - R3: $C \wedge D \rightarrow E$
  - R4: $B \wedge E \wedge F \rightarrow G$
  - R5: $A \wedge E \rightarrow H$
  - R6: $D \wedge E \wedge H \rightarrow I$

$WM_0 = \{ A, B, F \}$ $\quad CS_0 = \{ R1, R2 \}$
$WM_1 = \{ A, B, F, C \}$ $\quad CS_1 = \{ R2 \}$
$WM_2 = \{ A, B, F, C, D \}$ $\quad CS_2 = \{ R3 \}$
$WM_3 = \{ A, B, F, C, D, E \}$

Solution Exercise3 – Forward chaining

- Rules:
  - R1: $A \wedge B \rightarrow C$
  - R2: $A \rightarrow D$
  - R3: $C \wedge D \rightarrow E$
  - R4: $B \wedge E \wedge F \rightarrow G$
  - R5: $A \wedge E \rightarrow H$
  - R6: $D \wedge E \wedge H \rightarrow I$

$WM_0 = \{ A, B, F \}$	$CS_0 = \{ R1, R2 \}$
$WM_1 = \{ A, B, F, C \}$	$CS_1 = \{ R2 \}$
$WM_2 = \{ A, B, F, C, D \}$	$CS_2 = \{ R3 \}$
$WM_3 = \{ A, B, F, C, D, E \}$	$CS_3 = \{ R4, R5 \}$

Solution Exercise3 – Forward chaining

- Rules:
  - R1: A $\wedge$ B $\rightarrow$ C
  - R2: A $\rightarrow$ D
  - R3: C $\wedge$ D $\rightarrow$ E
  - R4: B $\wedge$ E $\wedge$ F $\rightarrow$ G
  - R5: A $\wedge$ E $\rightarrow$ H
  - R6: D $\wedge$ E $\wedge$ H $\rightarrow$ I

WM_0 = { A, B, F } CS_0 = { R1, R2 }
WM_1 = { A, B, F, C } CS_1 = { R2 }
WM_2 = { A, B, F, C, D } CS_2 = { R3 }
WM_3 = { A, B, F, C, D, E } CS_3 = { R4, R5 }
WM_4 = { A, B, F, C, D, E, G }

Solution Exercise3 – Forward chaining

- Rules:
  - R1: A $\wedge$ B $\rightarrow$ C
  - R2: A $\rightarrow$ D
  - R3: C $\wedge$ D $\rightarrow$ E
  - R4: B $\wedge$ E $\wedge$ F $\rightarrow$ G
  - R5: A $\wedge$ E $\rightarrow$ H
  - R6: D $\wedge$ E $\wedge$ H $\rightarrow$ I

$$
\begin{array}{ll}
WM_0 = \{ \text{A, B, F} \} & CS_0 = \{ \text{R1, R2} \} \\
WM_1 = \{ \text{A, B, F, C} \} & CS_1 = \{ \text{R2} \} \\
WM_2 = \{ \text{A, B, F, C, D} \} & CS_2 = \{ \text{R3} \} \\
WM_3 = \{ \text{A, B, F, C, D, E} \} & CS_3 = \{ \text{R4, R5} \} \\
WM_4 = \{ \text{A, B, F, C, D, E, G} \} & CS_4 = \{ \text{R5} \}
\end{array}
$$

Solution Exercise3 – Forward chaining

- Rules:
  - R1: A $\wedge$ B $\rightarrow$ C
  - R2: A $\rightarrow$ D
  - R3: C $\wedge$ D $\rightarrow$ E
  - R4: B $\wedge$ E $\wedge$ F $\rightarrow$ G
  - R5: A $\wedge$ E $\rightarrow$ H
  - R6: D $\wedge$ E $\wedge$ H $\rightarrow$ I

WM_0 = { A, B, F } — CS_0 = { R1, R2 }
WM_1 = { A, B, F, C } — CS_1 = { R2 }
WM_2 = { A, B, F, C, D } — CS_2 = { R3 }
WM_3 = { A, B, F, C, D, E } — CS_3 = { R4, R5 }
WM_4 = { A, B, F, C, D, E, G } — CS_4 = { R5 }
WM_5 = { A, B, F, C, D, E, G,H }

Solution Exercise3 – Forward chaining

- Rules:
  - R1: $A \wedge B \rightarrow C$
  - R2: $A \rightarrow D$
  - R3: $C \wedge D \rightarrow E$
  - R4: $B \wedge E \wedge F \rightarrow G$
  - R5: $A \wedge E \rightarrow H$
  - R6: $D \wedge E \wedge H \rightarrow I$

$WM_0 = \{$ A, B, F $\}$	$CS_0 = \{$ R1, R2 $\}$
$WM_1 = \{$ A, B, F, C $\}$	$CS_1 = \{$ R2 $\}$
$WM_2 = \{$ A, B, F, C, D $\}$	$CS_2 = \{$ R3 $\}$
$WM_3 = \{$ A, B, F, C, D, E $\}$	$CS_3 = \{$ R4, R5 $\}$
$WM_4 = \{$ A, B, F, C, D, E, G $\}$	$CS_4 = \{$ R5 $\}$
$WM_5 = \{$ A, B, F, C, D, E, G,H $\}$	

Solution Exercise3 – Backward chaining

- Rules:
  - R1: $A \wedge B \rightarrow C$
  - R2: $A \rightarrow D$
  - R3: $C \wedge D \rightarrow E$
  - R4: $B \wedge E \wedge F \rightarrow G$
  - R5: $A \wedge E \rightarrow H$
  - R6: $D \wedge E \wedge H \rightarrow I$

$WM_0 = \{$ A, B, F $\}$ Subgoals = $\{H\}$

Solution Exercise3 – Backward chaining

- Rules:
  - R1: A $\wedge$ B $\rightarrow$ C
  - R2: A $\rightarrow$ D
  - R3: C $\wedge$ D $\rightarrow$ E
  - R4: B $\wedge$ E $\wedge$ F $\rightarrow$ G
  - R5: A $\wedge$ E $\rightarrow$ H
  - R6: D $\wedge$ E $\wedge$ H $\rightarrow$ I

$WM_0 = \{$ A, B, F $\}$ Subgoals = $\{H\}$ $CS_0 = \{$ R5 $\}$

Solution Exercise3 – Backward chaining

- Rules:
  - R1: A $\wedge$ B $\rightarrow$ C
  - R2: A $\rightarrow$ D
  - R3: C $\wedge$ D $\rightarrow$ E
  - R4: B $\wedge$ E $\wedge$ F $\rightarrow$ G
  - R5: A $\wedge$ E $\rightarrow$ H
  - R6: D $\wedge$ E $\wedge$ H $\rightarrow$ I

$WM_0 = \{$ A, B, F $\}$	Subgoals = $\{H\}$	$CS_0 = \{$ R5 $\}$
$WM_1 = \{$ A, B, F $\}$	Subgoals = $\{E, (H)\}$	

Solution Exercise3 – Backward chaining

- Rules:
  - R1: A $\wedge$ B $\rightarrow$ C
  - R2: A $\rightarrow$ D
  - R3: C $\wedge$ D $\rightarrow$ E
  - R4: B $\wedge$ E $\wedge$ F $\rightarrow$ G
  - R5: A $\wedge$ E $\rightarrow$ H
  - R6: D $\wedge$ E $\wedge$ H $\rightarrow$ I

$WM_0 = \{$ A, B, F $\}$	Subgoals = $\{H\}$	$CS_0 = \{$ R5 $\}$
$WM_1 = \{$ A, B, F $\}$	Subgoals = $\{E, (H)\}$	$CS_1 = \{$ R3 $\}$

Solution Exercise3 – Backward chaining

- Rules:
  - R1: A $\wedge$ B $\rightarrow$ C
  - R2: A $\rightarrow$ D
  - R3: C $\wedge$ D $\rightarrow$ E
  - R4: B $\wedge$ E $\wedge$ F $\rightarrow$ G
  - R5: A $\wedge$ E $\rightarrow$ H
  - R6: D $\wedge$ E $\wedge$ H $\rightarrow$ I

$WM_0 = \{$ A, B, F $\}$	Subgoals = $\{H\}$	$CS_0 = \{$ R5 $\}$
$WM_1 = \{$ A, B, F $\}$	Subgoals = $\{E, (H)\}$	$CS_1 = \{$ R3 $\}$
$WM_2 = \{$ A, B, F $\}$	Subgoals = $\{D, C, (E), (H)\}$	

Solution Exercise3 – Backward chaining

- Rules:
  - R1: A $\wedge$ B $\rightarrow$ C
  - R2: A $\rightarrow$ D
  - R3: C $\wedge$ D $\rightarrow$ E
  - R4: B $\wedge$ E $\wedge$ F $\rightarrow$ G
  - R5: A $\wedge$ E $\rightarrow$ H
  - R6: D $\wedge$ E $\wedge$ H $\rightarrow$ I

WM_0 = { A, B, F }	Subgoals = $\{H\}$	CS_0 = { R5 }
WM_1 = { A, B, F }	Subgoals = $\{E, (H)\}$	CS_1 = { R3 }
WM_2 = { A, B, F }	Subgoals = $\{D, C, (E), (H)\}$	CS_2 = { R2 }

Solution Exercise3 – Backward chaining

- Rules:
  - R1: A $\wedge$ B $\rightarrow$ C
  - R2: A $\rightarrow$ D
  - R3: C $\wedge$ D $\rightarrow$ E
  - R4: B $\wedge$ E $\wedge$ F $\rightarrow$ G
  - R5: A $\wedge$ E $\rightarrow$ H
  - R6: D $\wedge$ E $\wedge$ H $\rightarrow$ I

WM_0 = { A, B, F }	Subgoals = $\{H\}$	CS_0 = { R5 }
WM_1 = { A, B, F }	Subgoals = $\{E, (H)\}$	CS_1 = { R3 }
WM_2 = { A, B, F }	Subgoals = $\{D, C, (E), (H)\}$	CS_2 = { R2 }
WM_3 = { A, B, F }	Subgoals = $\{(D), C, (E), (H)\}$	

Solution Exercise3 – Backward chaining

- Rules:
  - R1: A $\wedge$ B $\rightarrow$ C
  - R2: A $\rightarrow$ D
  - R3: C $\wedge$ D $\rightarrow$ E
  - R4: B $\wedge$ E $\wedge$ F $\rightarrow$ G
  - R5: A $\wedge$ E $\rightarrow$ H
  - R6: D $\wedge$ E $\wedge$ H $\rightarrow$ I

WM_0 = { A, B, F }	Subgoals = $\{H\}$	CS_0 = { R5 }
WM_1 = { A, B, F }	Subgoals = $\{E, (H)\}$	CS_1 = { R3 }
WM_2 = { A, B, F }	Subgoals = $\{D, C, (E), (H)\}$	CS_2 = { R2 }
WM_3 = { A, B, F }	Subgoals = $\{(D), C, (E), (H)\}$	CS_3 = { R1 }

Solution Exercise3 – Backward chaining

- Rules:
  - R1: A $\wedge$ B $\rightarrow$ C
  - R2: A $\rightarrow$ D
  - R3: C $\wedge$ D $\rightarrow$ E
  - R4: B $\wedge$ E $\wedge$ F $\rightarrow$ G
  - R5: A $\wedge$ E $\rightarrow$ H
  - R6: D $\wedge$ E $\wedge$ H $\rightarrow$ I

$WM_0 = \{$ A, B, F $\}$	Subgoals = $\{H\}$	$CS_0 = \{$ R5 $\}$
$WM_1 = \{$ A, B, F $\}$	Subgoals = $\{E, (H)\}$	$CS_1 = \{$ R3 $\}$
$WM_2 = \{$ A, B, F $\}$	Subgoals = $\{D, C, (E), (H)\}$	$CS_2 = \{$ R2 $\}$
$WM_3 = \{$ A, B, F $\}$	Subgoals = $\{(D), C, (E), (H)\}$	$CS_3 = \{$ R1 $\}$
$WM_4 = \{$ A, B, F $\}$	Subgoals = $\{(D), (C), (E), (H)\}$	

Solution Exercise3 – Backward chaining

- Rules:
  - R1: A $\wedge$ B $\rightarrow$ C
  - R2: A $\rightarrow$ D
  - R3: C $\wedge$ D $\rightarrow$ E
  - R4: B $\wedge$ E $\wedge$ F $\rightarrow$ G
  - R5: A $\wedge$ E $\rightarrow$ H
  - R6: D $\wedge$ E $\wedge$ H $\rightarrow$ I

$WM_0 = \{$ A, B, F $\}$	Subgoals = $\{H\}$	$CS_0 = \{$ R5 $\}$
$WM_1 = \{$ A, B, F $\}$	Subgoals = $\{E, (H)\}$	$CS_1 = \{$ R3 $\}$
$WM_2 = \{$ A, B, F $\}$	Subgoals = $\{D, C, (E), (H)\}$	$CS_2 = \{$ R2 $\}$
$WM_3 = \{$ A, B, F $\}$	Subgoals = $\{(D), C, (E), (H)\}$	$CS_3 = \{$ R1 $\}$
$WM_4 = \{$ A, B, F $\}$	Subgoals = $\{(D), (C), (E), (H)\}$	
$WM_5 = \{$ A, B, F, D, C, E, H $\}$		

Exercise 4a. Library

We have a library system with the following rules: when a person asks for a book the library will lend the book to that person if the book is available. If the book has been borrowed by somebody else, then the person reserves the book and waits for it to become available. When a person borrows a book, that person keeps the book until somebody else makes a reservation on it, and when that happens the book is returned. Each book can only have one reservation at any time.

Recap: Space-state representation problem

Model of a physical system as a set of variables and state constants related by first-order rules. Example: If the age of a patient is less than 10 years, s/he has red spots and fever then s/he has chicken pox.

- Variables: p1, 10, fever, red-spots, chicken-pox
- Constants: patient, symptom, disease, age.
- Rule: patient(p1),age(p1,10), symptom(p1,fever), symptom(p1,red-spots) -> disease(p1,chicken-pox)

Solution Exercise 4a. Representation

- asks(P,B): person P asks for book B
- available(B): book B is available
- borrowed(P,B): person P borrows the book B
- reserves_wait(P,B): person P reserves book B and waits for it
- reservation(B): book B is reserved

Solution Exercise 4a. Rules

R1(borrow): asks(P,B), available(B) $\rightarrow$ borrowed(P,B), $\neg$available(B), $\neg$asks(P,B)

R2(reserve): asks(P1,B), borrowed(P2,B), $\neg$reservation(B) $\rightarrow$ reserves_wait(P1,B), reservation(B), $\neg$asks(P1,B)

R3(return): reserves_wait(P1,B), borrowed(P2,B) $\rightarrow$ $\neg$borrowed(P2,B), available(B), asks(P1,B), $\neg$reservation(B)

Exercise 4b – Library

- The WM contains the following facts
  1. asks(student1, book1)
  2. asks(student2, book1)
  3. asks(student3, book1)
  4. available(book1)
- Execute the system under a FIFO conflict resolution strategy for 5 cycles. Show the WM, the CS and the executed rule for each cycle.

Solution Exercise4b

R1(borrow): asks(P,B), available(B) $\rightarrow$ borrowed(P,B), $\neg$available(B), $\neg$asks(P,B)
R2(reserve): asks(P1,B), borrowed(P2,B), $\neg$reservation(B) $\rightarrow$ reserves_wait(P1,B), reservation(B), $\neg$asks(P1,B)
R3(return): reserves_wait(P1,B), borrowed(P2,B) $\rightarrow$ $\neg$borrowed(P2,B), available(B), asks(P1,B), $\neg$reservation(B)

WM_0 = { asks(s1,b1), asks(s2,b1), asks(s3,b1), available(b1) }

Solution Exercise4b

R1(borrow): asks(P,B), available(B) $\rightarrow$ borrowed(P,B), $\neg$available(B), $\neg$asks(P,B)
R2(reserve): asks(P1,B), borrowed(P2,B), $\neg$reservation(B) $\rightarrow$ reserves_wait(P1,B), reservation(B), $\neg$asks(P1,B)
R3(return): reserves_wait(P1,B), borrowed(P2,B) $\rightarrow$ $\neg$borrowed(P2,B), available(B), asks(P1,B), $\neg$reservation(B)

WM_0 = { asks(s1,b1), asks(s2,b1), asks(s3,b1), available(b1) }
CS_0 = { R1(P=s1,B=b1), R1(P=s2,B=b1), R1(P=s3,B=b1) }

Solution Exercise4b

R1(borrow): asks(P,B), available(B) $\rightarrow$ borrowed(P,B), $\neg$available(B), $\neg$asks(P,B)
R2(reserve): asks(P1,B), borrowed(P2,B), $\neg$reservation(B) $\rightarrow$ reserves_wait(P1,B), reservation(B), $\neg$asks(P1,B)
R3(return): reserves_wait(P1,B), borrowed(P2,B) $\rightarrow$ $\neg$borrowed(P2,B), available(B), asks(P1,B), $\neg$reservation(B)

WM_0 = { asks(s1,b1), asks(s2,b1), asks(s3,b1), available(b1) }
CS_0 = { R1(P=s1,B=b1), R1(P=s2,B=b1), R1(P=s3,B=b1) } selected rule: first appearing in the CS

Solution Exercise4b

R1(borrow): asks(P,B), available(B) $\rightarrow$ borrowed(P,B), $\neg$available(B), $\neg$asks(P,B)
R2(reserve): asks(P1,B), borrowed(P2,B), $\neg$reservation(B) $\rightarrow$ reserves_wait(P1,B), reservation(B), $\neg$asks(P1,B)
R3(return): reserves_wait(P1,B), borrowed(P2,B) $\rightarrow$ $\neg$borrowed(P2,B), available(B), asks(P1,B), $\neg$reservation(B)

WM_0 = { asks(s1,b1), asks(s2,b1), asks(s3,b1), available(b1) }
CS_0 = { R1(P=s1,B=b1), R1(P=s2,B=b1), R1(P=s3,B=b1) } selected rule: first appearing in the CS

WM_1 = WM_0 $\cup$ { borrowed(s1,b1)} $-$ {asks(s1,b1), available(b1) }

Solution Exercise4b

R1(borrow):	asks(P,B), available(B) $\rightarrow$ borrowed(P,B), $\neg$available(B), $\neg$asks(P,B)
R2(reserve):	asks(P1,B), borrowed(P2,B), $\neg$reservation(B) $\rightarrow$ reserves_wait(P1,B), reservation(B), $\neg$asks(P1,B)
R3(return):	reserves_wait(P1,B), borrowed(P2,B) $\rightarrow$ $\neg$borrowed(P2,B), available(B), asks(P1,B), $\neg$reservation(B)

WM_0 = { asks(s1,b1), asks(s2,b1), asks(s3,b1), available(b1) }
CS_0 = { R1(P=s1,B=b1), R1(P=s2,B=b1), R1(P=s3,B=b1) } selected rule: first appearing in the CS

WM_1 = WM_0 $\cup$ { borrowed(s1,b1)} $-$ {asks(s1,b1), available(b1) }
CS_1 = { R2(P1=s2,P2=s1,B=b1), R2(P1=s3,P2=s1,B=b1) }

Solution Exercise4b

R1(borrow): asks(P,B), available(B) $\rightarrow$ borrowed(P,B), $\neg$available(B), $\neg$asks(P,B)
R2(reserve): asks(P1,B), borrowed(P2,B), $\neg$reservation(B) $\rightarrow$ reserves_wait(P1,B), reservation(B), $\neg$asks(P1,B)
R3(return): reserves_wait(P1,B), borrowed(P2,B) $\rightarrow$ $\neg$borrowed(P2,B), available(B), asks(P1,B), $\neg$reservation(B)

WM_0 = { asks(s1,b1), asks(s2,b1), asks(s3,b1), available(b1) }
CS_0 = { R1(P=s1,B=b1), R1(P=s2,B=b1), R1(P=s3,B=b1) } selected rule: first appearing in the CS

$WM_1 = WM_0 \cup$ { borrowed(s1,b1)} $-$ {asks(s1,b1), available(b1) }
CS_1 = { R2(P1=s2,P2=s1,B=b1), R2(P1=s3,P2=s1,B=b1) }

$WM_2 = WM_1 \cup$ { reserves_wait(s2,b1), reservation(b1)} $-$ {asks(s2,b1)}

Solution Exercise4b

R1(borrow): asks(P,B), available(B) $\rightarrow$ borrowed(P,B), $\neg$available(B), $\neg$asks(P,B)
R2(reserve): asks(P1,B), borrowed(P2,B), $\neg$reservation(B) $\rightarrow$ reserves_wait(P1,B), reservation(B), $\neg$asks(P1,B)
R3(return): reserves_wait(P1,B), borrowed(P2,B) $\rightarrow$ $\neg$borrowed(P2,B), available(B), asks(P1,B), $\neg$reservation(B)

WM_0 = { asks(s1,b1), asks(s2,b1), asks(s3,b1), available(b1) }
CS_0 = { R1(P=s1,B=b1), R1(P=s2,B=b1), R1(P=s3,B=b1) } selected rule: first appearing in the CS

$WM_1 = WM_0 \cup$ { borrowed(s1,b1)} $-$ {asks(s1,b1), available(b1) }
CS_1 = { R2(P1=s2,P2=s1,B=b1), R2(P1=s3,P2=s1,B=b1) }

$WM_2 = WM_1 \cup$ { reserves_wait(s2,b1), reservation(b1)} $-$ {asks(s2,b1)}
CS_2 = { R3(P1=s2,P2=s1,B=b1) }

Solution Exercise4b

R1(borrow): asks(P,B), available(B) $\rightarrow$ borrowed(P,B), $\neg$available(B), $\neg$asks(P,B)
R2(reserve): asks(P1,B), borrowed(P2,B), $\neg$reservation(B) $\rightarrow$ reserves_wait(P1,B), reservation(B), $\neg$asks(P1,B)
R3(return): reserves_wait(P1,B), borrowed(P2,B) $\rightarrow$ $\neg$borrowed(P2,B), available(B), asks(P1,B), $\neg$reservation(B)

WM_0 = { asks(s1,b1), asks(s2,b1), asks(s3,b1), available(b1) }
CS_0 = { R1(P=s1,B=b1), R1(P=s2,B=b1), R1(P=s3,B=b1) } selected rule: first appearing in the CS

$WM_1 = WM_0 \cup$ { borrowed(s1,b1)} $-$ {asks(s1,b1), available(b1) }
CS_1 = { R2(P1=s2,P2=s1,B=b1), R2(P1=s3,P2=s1,B=b1) }

$WM_2 = WM_1 \cup$ { reserves_wait(s2,b1), reservation(b1)} $-$ {asks(s2,b1)}
CS_2 = { R3(P1=s2,P2=s1,B=b1) }

$WM_3 = WM_2 \cup$ {asks(s2,b1), available(b1)} $-$ {borrowed(s1,b1), reservation(b1)}

Solution Exercise4b

R1(borrow):	asks(P,B), available(B) $\rightarrow$ borrowed(P,B), $\neg$available(B), $\neg$asks(P,B)
R2(reserve):	asks(P1,B), borrowed(P2,B), $\neg$reservation(B) $\rightarrow$ reserves_wait(P1,B), reservation(B), $\neg$asks(P1,B)
R3(return):	reserves_wait(P1,B), borrowed(P2,B) $\rightarrow$ $\neg$borrowed(P2,B), available(B), asks(P1,B), $\neg$reservation(B)

WM_0 = { asks(s1,b1), asks(s2,b1), asks(s3,b1), available(b1) }
CS_0 = { R1(P=s1,B=b1), R1(P=s2,B=b1), R1(P=s3,B=b1) } selected rule: first appearing in the CS

$WM_1 = WM_0 \cup$ { borrowed(s1,b1)} $-$ {asks(s1,b1), available(b1) }
CS_1 = { R2(P1=s2,P2=s1,B=b1), R2(P1=s3,P2=s1,B=b1) }

$WM_2 = WM_1 \cup$ { reserves_wait(s2,b1), reservation(b1)} $-$ {asks(s2,b1)}
CS_2 = { R3(P1=s2,P2=s1,B=b1) }

$WM_3 = WM_2 \cup$ {asks(s2,b1), available(b1)} $-$ {borrowed(s1,b1), reservation(b1)}
CS_3 = { R1(P=s3,B=b1), R1(P=s2,B=b1) }

Our system does not force to borrow books to the person with reservations!

Solution Exercise4b

R1(borrow): asks(P,B), available(B) $\rightarrow$ borrowed(P,B), $\neg$available(B), $\neg$asks(P,B)
R2(reserve): asks(P1,B), borrowed(P2,B), $\neg$reservation(B) $\rightarrow$ reserves_wait(P1,B), reservation(B), $\neg$asks(P1,B)
R3(return): reserves_wait(P1,B), borrowed(P2,B) $\rightarrow$ $\neg$borrowed(P2,B), available(B), asks(P1,B), $\neg$reservation(B)

WM_0 = { asks(s1,b1), asks(s2,b1), asks(s3,b1), available(b1) }
CS_0 = { R1(P=s1,B=b1), R1(P=s2,B=b1), R1(P=s3,B=b1) } selected rule: first appearing in the CS

$WM_1 = WM_0 \cup$ { borrowed(s1,b1)} $-$ {asks(s1,b1), available(b1) }
CS_1 = { R2(P1=s2,P2=s1,B=b1), R2(P1=s3,P2=s1,B=b1) }

$WM_2 = WM_1 \cup$ { reserves_wait(s2,b1), reservation(b1)} $-$ {asks(s2,b1)}
CS_2 = { R3(P1=s2,P2=s1,B=b1) }

$WM_3 = WM_2 \cup$ {asks(s2,b1), available(b1)} $-$ {borrowed(s1,b1), reservation(b1)}
CS_3 = { R1(P=s3,B=b1), R1(P=s2,B=b1) }

Our system does not force to borrow books to the person with reservations!

$WM_4 = WM_3 \cup$ {borrowed(s3,b1)} $-$ {asks(s3,b1), available(b1) }

Solution Exercise4b

R1(borrow): asks(P,B), available(B) $\rightarrow$ borrowed(P,B), $\neg$available(B), $\neg$asks(P,B)
R2(reserve): asks(P1,B), borrowed(P2,B), $\neg$reservation(B) $\rightarrow$ reserves_wait(P1,B), reservation(B), $\neg$asks(P1,B)
R3(return): reserves_wait(P1,B), borrowed(P2,B) $\rightarrow$ $\neg$borrowed(P2,B), available(B), asks(P1,B), $\neg$reservation(B)

WM_0 = { asks(s1,b1), asks(s2,b1), asks(s3,b1), available(b1) }
CS_0 = { R1(P=s1,B=b1), R1(P=s2,B=b1), R1(P=s3,B=b1) } selected rule: first appearing in the CS

WM_1 = WM_0 $\cup$ { borrowed(s1,b1)} $-$ {asks(s1,b1), available(b1) }
CS_1 = { R2(P1=s2,P2=s1,B=b1), R2(P1=s3,P2=s1,B=b1) }

WM_2 = WM_1 $\cup$ { reserves_wait(s2,b1), reservation(b1)} $-$ {asks(s2,b1)}
CS_2 = { R3(P1=s2,P2=s1,B=b1) }

WM_3 = WM_2 $\cup$ {asks(s2,b1), available(b1)} $-$ {borrowed(s1,b1), reservation(b1)}
CS_3 = { R1(P=s3,B=b1), R1(P=s2,B=b1) }

Our system does not force to borrow books to the person with reservations!

WM_4 = WM_3 $\cup$ {borrowed(s3,b1)} $-$ {asks(s3,b1), available(b1) }
CS_4 = { R2(P1=s2,P2=s3,B=b1) }

Solution Exercise4b

R1(borrow): asks(P,B), available(B) $\rightarrow$ borrowed(P,B), $\neg$available(B), $\neg$asks(P,B)
R2(reserve): asks(P1,B), borrowed(P2,B), $\neg$reservation(B) $\rightarrow$ reserves_wait(P1,B), reservation(B), $\neg$asks(P1,B)
R3(return): reserves_wait(P1,B), borrowed(P2,B) $\rightarrow$ $\neg$borrowed(P2,B), available(B), asks(P1,B), $\neg$reservation(B)

WM_0 = { asks(s1,b1), asks(s2,b1), asks(s3,b1), available(b1) }
CS_0 = { R1(P=s1,B=b1), R1(P=s2,B=b1), R1(P=s3,B=b1) } selected rule: first appearing in the CS

$WM_1 = WM_0 \cup$ { borrowed(s1,b1)} $-$ {asks(s1,b1), available(b1) }
CS_1 = { R2(P1=s2,P2=s1,B=b1), R2(P1=s3,P2=s1,B=b1) }

$WM_2 = WM_1 \cup$ { reserves_wait(s2,b1), reservation(b1)} $-$ {asks(s2,b1)}
CS_2 = { R3(P1=s2,P2=s1,B=b1) }

$WM_3 = WM_2 \cup$ {asks(s2,b1), available(b1)} $-$ {borrowed(s1,b1), reservation(b1)}
CS_3 = { R1(P=s3,B=b1), R1(P=s2,B=b1) }

Our system does not force to borrow books to the person with reservations!

$WM_4 = WM_3 \cup$ {borrowed(s3,b1)} $-$ {asks(s3,b1), available(b1) }
CS_4 = { R2(P1=s2,P2=s3,B=b1) }

$WM_5 = WM_4 \cup$ { reserves_wait(s2,b1), reservation(b1)} $-$ {asks(s2,b1)}

Exercise 5. Board game

- Our game is played on a 4x4 board in which the opponents (White and Black) start with two tokens each. White starts at the bottom left and top right, and Black in the other two corners.
- A square on the board is *k-adjacent* to another if exactly k movements (horizontal, vertical, diagonal or a combination of them) are required to go from first to second.
- In each turn the player can make one of the following actions
  - Copy a token to a 1-adjacent square
  - Move a token to a 2-adjacent square
- The game ends when no player can move. The winner is the player with more tokens in the board
- How can we formalize the world of this game in a production system?
- Hint: draw the game

Recap: Space-state representation problem

1. Identify initial state
2. Identify operators (successor function): describes possible actions from one state to another (state transformation)
3. Identify reachable states from initial state. Defined by 1 and 2.
4. Identify final states.
5. Identify cost: sum of the number of actions (operators) performed.
6. Solution from initial to final states.
7. There can be priorities in rules.

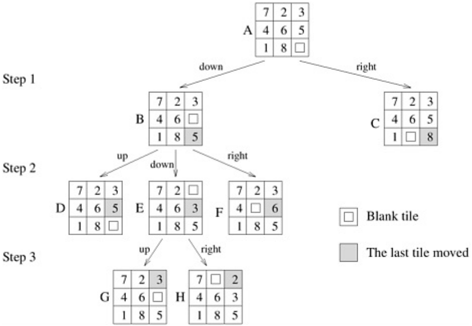
Recap: Space-state representation problem 2

Abstract the problem as much as possible

- Describe just the strictly necessary
- State: describe de location of each token in the board
- Operators should be as general as possible. We want to reduce the number of rules. Example of the 8-puzzle:
  - We could have 9!x4 operators to go to each possible state. That is all possible permutations of numbers in board 4x4. X
  - We could have 8x4 operators that move each number (there are 8 numbers) and 4 movements. X
  - We could have just 4 operators that move the empty space: up, down, left, right. ✓

Recap: Space-state representation problem 3

8 puzzle example: operators



Solution Exercise5. Representation

- Cells: constants $c11$, $c12$, ..., $c44$

Solution Exercise5. Representation

- Cells: constants *c*11, *c*12, ..., *c*44
- **state(CELL, COLOR)**: cell *CELL* has a token of color *COLOR*
  We choose not representing empty cells!
  state(c11, black), state(c44, black), state(c41,white), state(c14, white)

Solution Exercise5. Representation

- Cells: constants *c*11, *c*12, ..., *c*44
- **state(CELL, COLOR)**: cell *CELL* has a token of color *COLOR*
  We choose not representing empty cells!
  state(c11, black), state(c44, black), state(c41,white), state(c14, white)
- **opponent(COLOR1,COLOR2)**: color COLOR2 is opponent of color COLOR1
  opponent(white,black), opponent(black,white)

Solution Exercise5. Representation

- Cells: constants *c*11, *c*12, ..., *c*44
- **state(CELL, COLOR)**: cell *CELL* has a token of color *COLOR*
  We choose not representing empty cells!
  state(c11, black), state(c44, black), state(c41,white), state(c14, white)
- **opponent(COLOR1,COLOR2)**: color COLOR2 is opponent of color COLOR1
  opponent(white,black), opponent(black,white)
- **adj(k,CELL1,CELL2)**: CELL2 is k-adjacent to CELL1
  adj(1,c11,c21), adj(1,c21,c11), ..., adj(2,c11,c13), adj(2,c13,c11), ...

Solution Exercise5. Representation

- Cells: constants *c*11, *c*12, ..., *c*44
- **state(CELL, COLOR)**: cell *CELL* has a token of color *COLOR*
  We choose not representing empty cells!
  state(c11, black), state(c44, black), state(c41,white), state(c14, white)
- **opponent(COLOR1,COLOR2)**: color COLOR2 is opponent of color COLOR1
  opponent(white,black), opponent(black,white)
- **adj(k,CELL1,CELL2)**: CELL2 is k-adjacent to CELL1
  adj(1,c11,c21), adj(1,c21,c11), ..., adj(2,c11,c13), adj(2,c13,c11), ...
- **turn(COLOR)**: next turn is for COLOR
  turn(white)

Solution Exercise5. Representation

- Cells: constants *c*11, *c*12, ..., *c*44
- **state(CELL, COLOR)**: cell *CELL* has a token of color *COLOR*
  We choose not representing empty cells!
  state(c11, black), state(c44, black), state(c41,white), state(c14, white)
- **opponent(COLOR1,COLOR2)**: color COLOR2 is opponent of color COLOR1
  opponent(white,black), opponent(black,white)
- **adj(k,CELL1,CELL2)**: CELL2 is k-adjacent to CELL1
  adj(1,c11,c21), adj(1,c21,c11), ..., adj(2,c11,c13), adj(2,c13,c11), ...
- **turn(COLOR)**: next turn is for COLOR
  turn(white)
- **tokens(COLOR,N)**: COLOR has *N* tokens
  tokens(white,2), tokens(black,2)

Solution Exercise5. Representation

- Cells: constants *c*11, *c*12, ..., *c*44
- **state(CELL, COLOR)**: cell *CELL* has a token of color *COLOR*
  We choose not representing empty cells!
  state(c11, black), state(c44, black), state(c41,white), state(c14, white)
- **opponent(COLOR1,COLOR2)**: color COLOR2 is opponent of color COLOR1
  opponent(white,black), opponent(black,white)
- **adj(k,CELL1,CELL2)**: CELL2 is k-adjacent to CELL1
  adj(1,c11,c21), adj(1,c21,c11), ..., adj(2,c11,c13), adj(2,c13,c11), ...
- **turn(COLOR)**: next turn is for COLOR
  turn(white)
- **tokens(COLOR,N)**: COLOR has *N* tokens
  tokens(white,2), tokens(black,2)
- **winner(COLOR)**: COLOR wins
- **dead_heat**: game ends with dead_heat

Solution Exercise 5. Rules

R1(copy):	turn(Color1), state(Cell1, Color1), adj(1,Cell1,Cell2), ¬ state(Cell2,Color), tokens(Color1,N), opponent(Color1,Color2)	$\rightarrow$	state(Cell2,Color1), ¬ turn(Color1), turn(Color2), ¬ tokens(Color1,N), tokens(Color1, N+1)
R2(move):	turn(Color1), state(Cell1, Color1), adj(2,Cell1,Cell2), ¬ state(Cell2,Color), opponent(Color1,Color2)	$\rightarrow$	state(Cell2,Color1), ¬ state(Cell1, Color1), ¬ turn(Color1), turn(Color2)
R3(end1):	tokens(Color1,N), tokens(Color2,M), $N > M$	$\rightarrow$	winner(Color1)
R3(end2):	tokens(Color1,N), tokens(Color2,M), $N = M$	$\rightarrow$	dead_heat

Rules R1 and R2 have a higher priority than R3. Then, R3 only fires when R1 and R2 have no instances